Государственное образовательное учреждение высшего профессионального образования

«Московский государственный технический университет имени Н. Э. Баумана»

(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Р А С Ч Ё Т Н О - П О Я С Н И Т Е Л Ь Н А Я З А П И С К А

к дипломной работе:

Метод обнаружения выбросов временных рядов

Студент Капустин А.И. И.О. Фамилия
(Подпись, дата)

Руководитель курсового проекта Оленев А.А. И.О. Фамилия
(Подпись, дата)

Москва 2016

Содержание

Глоссарий

Выборка/выборка данных — конечный набор прецедентов (объектов, случаев, событий, испытуемых, образцов, и т.п.), некоторым способом выбранных из множества всех возможных прецедентов, называемого генеральной совокупностью[1].

— Метка(ярлык) - порция данных, идентифицирующая набор данных, описывающая его определенные свойства и обычно хранимая в том же пространстве памяти, что и набор данных[2].

классификатор

Теория распознавания образа — раздел информатики и смежных дисциплин, развивающий основы и методы классификации и идентификации предметов, явлений, процессов, сигналов, ситуаций и т.п. объектов, которые характеризуются конечным набором некоторых свойств и признаков. ddos-атака? Датасет - набор данных[3]

Введение

Задача поиска аномалий является одной из классических задач машинного обучения. В настоящее время задачу поиска аномалий активно решают во многих областях жизнедеятельности:

а) Защита информации и безопастность

б) Социальная сфера и медицина

в) Банковская и финансовая отрасль

г) Распознавание и обработка текста, изображений, речи

д) Другие сферы деятельности(например, мониторинг неисправностей механизмов)

Задачей поиска выбросов, как частный случай задачи поиска аномалий так же занимаются во всех вышеперичисленных отраслях.

Количество данных в мире удваивается примерно каждые два года. Поэтому актуальной задачей является разработка новых методов и усовершенствования старых методов поиска выбросов.

В данной работе предлагается новый метод, позволяющий найти аномалии в выборках данных.

1 Аналитический раздел

1.1 Цель и задачи работы

Целью данной работы является создание программного комплекста для обнаружения выбросов временных рядов в собираемых данных. Для достижения данной цели необходимо решить следующие задачи:

— пронализировать предметную область и существующие методы обнаружения выбросов

— разработать метод обнаружения выбросов

— создать ПО, собирающее данные для анализа

— создать ПО, реализующего разработанный метод обнаружения выбросов

— провести вычислительный эксперимент с использованием разработанного метода

1.2 Что такое аномалия

В анализе данных есть два основных направления, которые занимаются поиском аномалий - это детектирование новизны и обнаружение выбросов. "Объект новизны" - это так же объект, который отличается по своим свойствам от объектов выборки. Однако, в отличие от выброса, его ещё нет в самой выборке и задача анализа сводится к его обнаружению при появление. Например, если анализировать замеры уровня шума и отбрасывать слишком высокие или слишком низкие значения, то это называется борьбой с выбросами. А если создаётся алгоритм, который для каждого нового замера оценивает, насколько он похож на прошлые, и выбрасывает аномальные, то это назвается "борьбой с новизной" . [4]. Выбросы являются следствием:

а) ошибок в данных
б) неверно классифицированных объектов
в) присутсвием объектов других выборок
г) намеренным искажением данных

На рисунке 1.1 находится три вида точек: зеленые, желтые, красные. Множество зеленых точек представляют собой "нормальные"данные. Множество желтых точек означает выбросы в "слабом смысле". Они незначительно отклоняются от основных нормальных данных. Красные же точки являются аномальными - выбросами "в сильном смысле" . Они значительно отклоняются от нормальных данных. В данной работе будет изучаться вопрос находждения "сильных выбросов" и критериев отличия сильного выброса от основных данных. В дальнейшем под словом "выброс" будет подразумеваться "сильный выброс" , а под аномалией - выброс(выброс - част-

ный случай аномалии). Понятие аномалии интерпетируют по-разному в зависимости от характера данных. Обычно аномалией назыют некоторое отклонение от нормы. В дальнейшем будет дано несколько более формальных опредений аномалий, специфичных для метода их определений.

1.3 Обнаружение аномалий

В машинном обучении обнаружение "ненормальных"экземпляров в наборах данных всегда представляло большой интерес. Вероятно, первое определение было дано Граббсом[5] в 1969 году: "Относительное наблюдение или выброс - это элемент выборки, который, заметно отличается от других членов выборки, в которых он встречается ". Это определение является актуальным и сегодня, но мотивация для обнаружения аномалий изменилась. Тогда основная причина поиска аномалий заключалась в том, чтобы удалить выбросы из набора данных для обучения, поскольку используемые алгоритмы, были весьма чувствительны к выбросам в данных. Эта процедура также называется очищением данных. После разработки классификаторов устойчивых к наличию аномалий в обучающем наборе данных, интерес к их поиску угас. Однако, в начале 21 века в связи с развитием интернета и значительным увеличением объема собираемых данных для анализа, исследователи стали больше интересоваться аномалиями, поскольку они оказывались часто связаны с особенно интересными событиями. В этом контексте определение Граббса также было расширено, так что сегодня аномалии имеют две важные характеристики:

а) Аномалия отличается от нормы по своим особенностям

б) Аномалия редко встречается в наборе данных по сравнению с "нормальными" данными

1.3.1 Классификация методов обнаружений аномалий

Классическая система классификации предполагает предварительное обучение на обучающем наборе данных и последующую классификацию на основе этого набора. Данные делятся на "обучающую выборку" - данные, при помощи которых алогритм обучает классификар и, "тестовую выборку" - данные, при анилизе которых, классификатор остается неизменным.Тестовая выборка нужна для того чтобы проверить корректность обучения классификатора.

Однако, в случае с поиском аномалий, возможны варианты, отличающиеся от классического. Подходящий метод классификации выбирается на основе наличия разметки данных. Выделяются три основых класса методов:

а) Обучение с учителем. Для обучения необходимо наличие полностью размеченных данные для обучения и для тестов. Классификатор обучается один раз и

применяться впоследствии.В связи с тем, что для многих наборов данных заранее неизвестно что является аномалией, а что нет, применение этого метода ограничено.

б) Обучение с частичным привлечением учителя. Для обучения необходимо наличие тестового и учебного набора данных. Однако, в отличие от обучения с привелечением учителя, разметка данных не требуется. Все данные, представленные в выборках, считаются нормальными. На основе этих данных строится некая модель. Все данные, отклоняющиеся от этой модели, считаются аномальными. Эта идея также известна как "одноклассовая"классификация [6].

в) Обучение без учителя. Самый гибкий способ, который не требует разметки набора данных. Идея заключается в том, что алгоритм обнаружения аномалий оценивает данные исключительно на основе внутренних свойств набора данных что является нормальным, а что является выбросом. В этой работе основное внимание будет этому именно этому способу. Так же этот способ называют "неконтролируемый способ обнаружения аномалий".

1.4 Результат метода обнаружения аномалий

В результате работы алгоритма обнаружения аномалий с элементом данных связаывается метка или оценка достоверности(показатель аномальности). Метка- показатель, который принимает нулевое значения, в случае если она связана с нормальными данными и единицу в противном случае. Оценка показывает вероятность того, что элемент является аномалией. Для разных алгоритмов используется разные шкалы оценок, поэтому приведение конкретных примеров оценок будет некорректным. В алгоритмах метода обучения с учителем зачастую используются метки как выходные данные, в алгоритмах с частичным привлечением учителя и без учителя обнаружения аномалий чаще встречаются оценки.

1.5 Виды аномалий

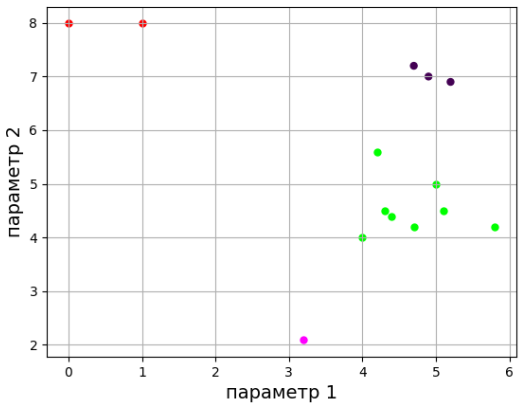

Рисунок 1.1 — Простой двумерный пример

Основная идея алгоритмов обнаружения аномалий заключается в обнаружении экземпляров данных в наборе данных, которые отклоняются от нормы. Однако на практике существует множество случаев, когда это основное предположение является неоднозначным. На рис 1.1 показаны некоторые из этих случаев с использованием простого двумерного набора данных. Две аномалии могут быть легко идентифицированы визуально: красные точки сильно отличаются отличаются значениям параметров от областей плотной группировки точек. Если смотреть на весь набор данных в целом, то фиолетовую точку можно отнести к тому же классу, что и зеленые точки. Однако, если сфокусироваься только на кластере зеленых точек и сравнивать его с фиолетовой точкой, пренебрегая всеми другими точками, то её можно рассматривать как аномалию. Поэтому фиолетовая точка называется локальной аномалией, так как она аномальна по сравнению с ее близкой окрестностью. В зависимости от цели анализа, локальные аномалии могут представлять интерес или нет. Другой вопрос заключается в том,что следует ли рассматривать точки черного кластера как три аномалии или как (небольшой) кластер. Такие небольшие кластеры явления называются микрокластерами. Показатели аномальности у точек этого кластера выше, чем у точек зеленого кластера, но меньше, чем у красных точек. Этот простой пример показывает, что задача нахождения аномалий аномалии не всегда тривиальна, а вычисление показателя аномалности иногда полезнее, чем проставление двоичной метки.

Задача обнаружения одиночных аномальных экземпляров крупном наборе данных называется обнаружением точечных аномалии[7]. Сегодня почти все неконтролируемые алгоритмы обнаружения относятся к этому типу. Если же аномалии составляют заметный процент, от набора данных, то задачу поиска аномалий называют задачей обнаружения коллектнивных аномалий. Пусть аномалии представляют собой некое множество, тогда необязательно каждый элемент этого множества должен быть аномальным. Возможен вариант когда только определенная их комбинация определяет аномалию. Третий вид - контекстуальные аномалии. Элемент выборке в отрыве от своего контекста может казаться нормальным. Однако, если рассмотреть контекст этого элемента, то очевидным станет его аномальная природа. Распространенным контекстом является время. В качестве примера предположим, что измеряется температура в диапазоне от $-30°C$ до $+40°C$ в течение года. Таким образом, температура $25°C$ кажется довольно нормальной, но когда учитывается контекстное время (например, месяц), такая высокая температура $25°C$ в течение зимы будет рассматриваться как аномалия.

Алгоритмы обнаружения точечных аномалий так же можно использовать для обнаружения контекстуальных и коллективных аномалий. Для этого нужно включить контекст в алгоритм как параметр алгоритма. В вышеприведенным примере включение месяца как дополнительного параметра поможет обнаружить анома-

лию. Однако в более сложных сценариях может потребоваться один или несколько новых парметров, чтобы преобразовать задачу определения контекстной аномалии в задачу обнаружения точечной аномалии. Для того, чтобы преобразовать задачу поиска коллективной аномалии в задачу поиска одиночной, нужно произвести измненения изначального набора данных. Для этого можно использовать корреляцию, агрегация и группировка. Преобразование может быть нетривиальным.[8] . Преобразование требует глубоких знаний о наборе исходных данных и часто приводит к существенным искажениям при переводе данных в новый формат. Такое семантическое преобразование называется генерированием представления данных(*англ. data view generation*).

Таким образом можно сделать вывод, что многие задачи обнаружения аномалий требуют предварительной обработки данных перед передачей их на вход алгоритму. В противном случае можно получить формально верные, но фактические бесполезные результаты.

1.5.1 Нормализация данных

После получения предварительно обработанного датасета для поиска точечной аномалии, то последним шагом перед передачей в алгоритм, является нормализация данных. Нормализация данных предназначена для устранения зависимости от выбора единицы измерения и заключается в преобразовании диапазонов значений всех атрибутов к стандартным интервалам([0,1] или [-1,1])[9]. Нормализация данных направлена на придание всем атрибутам одинакового "веса".

1.5.1.1 Основные методы нормализация данных

а) Min-max нормализация заключается в применении к диапазону значений атрибута x линейного преобразования, которое отображает [min(x),max(x)] в [A,B].

$$x'_i = \tau(x_i) = \frac{x_i - min(x)}{max(x) - min(x)} * (B - A) + A \tag{1.1}$$

$$x \in [min(x), max(x)] \Rightarrow \tau() \Rightarrow [A,B] \tag{1.2}$$

Min-max нормализация сохраняет все зависимости и порядок оригинальных значений атрибута. Недостатком этгого метода является то, что выбросы могут сжать основную массу значений к очень маленькому интервалу

б) Z-нормализация основывается на приведении распределения исходного атрибута x к центрированному распределению со стандартным отклоненим, равным 1 [9]
.

$$x'_i = \tau(x_i) = \frac{x_i - \overline{x}}{\sigma_x} \tag{1.3}$$

$$M[x'] = 1 \tag{1.4}$$

$$D[\overline{x}'] = 0 \tag{1.5}$$

Метод полезен когда в данных содержатся выбросы.

в) Масштабирование заключается в изменении длины вектора значений атрибута путем умножения на константу [9] .

$$x'_i = \tau(x_i) = \lambda * x_i \tag{1.6}$$

Длина вектора x уменьшается при $|\lambda| < 1$ и увеличивается при $|\lambda| > 1$

1.6 Неконтролируемые алгоритмы обнаружения аномалий

1.6.1 Вероятностный-генеративный подход

Основная идея генеративного подхода заключается в использование вероятностного смесевого моделирования данных. Предлагается подобрать такую вероятностую модель, из которой было получены нормированные данные. Такие модели обычно называются генеративными моделями, где для каждой точки(элемента данных) можем посчитать генеративную вероятность(или вероятность правдоподобия).Т.е. задача сводится к нахождению плотности распределения p(x). Аномалиями при этом считаются точки(элементы набора данных), имеющию низкое правдоподобие. В качестве показателя аномальности выступает функция p. Для построения генеративной модели нужно решить следующую задачу:

$$\prod_{x \in X_{norm}} p(x,\theta) \rightarrow max_\theta \tag{1.7}$$

где X_{norm} - нормальные данные представленного набора данных $p(x,\theta)|\theta \in \omega$ -семейство плотностей вероятностей, параметризованные θ;

Этот метод редко используется на практике, так как тяжело проверить полученную генеративную модель на адекватность, сложно убедится в правильном выборе семейства смесевых распределений. Это связано с тем, что низкое значение функции правдоподобия может означать как и аномальное значение, так и неудачно подобранную модель. Этот метод применяется с опорой на априорную информацию, в случае когда можно проверить полученную модель на адекватность.

1.6.2 Линейный подход

Основной идей линейного подхода является построение некой модели, характеризующей нормальные данные. Точки, которые значительно отклоняются от этой модели, считаются аномалиями.

Предполагается, что нормальные данные находятся в подпространсве пространства атрибутов данных(размер подпространства атрибутов данных равен

размерности данных). В свою очередь, задача линейного метода - найти низкоразмерное подпространства, такие что, выборка данных этого подпространства значительно отличается от остальных точек пространства данных.

Одним из возможных вариантов решения является использование линейной регрессии. Выбирается одна из наблюдаемых переменных набора данных и относительно неё решается задача линейной регрессии оставшихся атрибутов. Итоговым ответом будет является усредненное значения показателя аномалии по всем атрибутам.

Алгоритмы, основанные на линейном подходе, требуют наличия линейной зависимости атрибутов данных.

1.6.3 Метрические методы

Мерические методы пытаются найти в данных точки, в некотором смысле изолированные от остальных[4]. Если в пространстве задана некоторая метрика *p(x1, x2)*, то необходимо задать следующие понятия:

— Аномалии – точки, не попадающие ни в один кластер. К данным применяется один из алгоритмов кластеризации; размер кластера, в котором оказалась точка, объявляется её показатель аномальности.

— Локальная плотность в аномальных точках низкая. Для данной точки показателем аномальности объявляется локальная плотность, которая оценивается некоторым непараметрическим способом.

— Расстояние от данной точки до ближайших соседей велико.

В качестве показателя аномальности может выступать:

— расстояние до k-го ближайшего соседа;

— среднее расстояние до k ближайших соседей;

— медиана расстояний до k ближайших соседей;

— гармоническое среднее до k ближайших соседей;

— доля из k ближайших соседей, для которых данная точка является не более чем k-ым соседом и много другое.

1.6.3.1 Базовые понятия

Метрические методы хорошо подоходят в случае когда данные не размечены. Сложность вычисления прямо пропорциональна как размерности данных m,как и их количеству n. При росте набора данных наблюдается экспоненциальный рост сложности вычислений. Однако, эти методы хорошо проявляют себя на ограниченных наборах данных[10]. Следовательно такие методы как k-ближайших соседей(так же известный как обучение на основе примеров, и описанный поздее) с нотацией ассимт-

потического роста $O(n^2m)$ недопустимы для наборов данных с большой размерности, если их размерность не может быть уменьшена.

Существуют много различных вариации алгоритма k-ближайших соседей для обнаружения аномалий, но все они основаны на вычислении некой метрики "расстояния до соседей такой как Евклидово расстояние или расстояние Махаланобиса. Евклидово расстояние задается следующей формулой:

$$\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{1.8}$$

и является просто расстоянием между двумя точками, когда как расстояние Махаланобиса, задаваемое следующей формулой

$$\sqrt{(x-\mu)^T C^{-}1(x-\mu)} \tag{1.9}$$

вычисляет расстояние от точки до центра тяжести (μ), определяемого формулой коррелированных атрибутов, заданных матрицей ковариации (C). Расстояние Махаланобиса рассчитывается значительно дольше по сравнению с евлидковым по сравнению с евклидовым расстоянием для для больших объемов данных, поскольку оно требует пройти через весь набор данных, чтобы идентифицировать корреляции атрибутов.

1.6.3.2 Оптимизация Рамасвани

Точка p является выбросом, если не более n - 1 других точек в наборе данных имеют более высокий D_m(расстояние до m соседей), где m задается. Например на рисунке 1.2 черная точка является наиболее удаленной от соседей, следовательно она является выбросом. Красные точки расположены рядом друг с другом, однако расстояние до других точек велико, следовательно они тоже являются аномалиями. Такой подход вприимчив к вычислительному росту, потому что должна быть вычислена матрица расстояний точек друг от друга, поэтому Рамасвани в 2000 году предложил оптимизацию метода k-ближайших соседей(с англ. k-Nearest Neighbour - k-NN) в виде составления ранжированного списка потенциальных выбросов.

Оптимизация Рамасвани заключается в разбиении данных на ячейки.Если какая-либо ячейка и ее ближайшие соседи содержат больше, чем k точек, то точки в ячейке считаются лежащими в плотной области поэтому содержащиеся там точки вряд ли будут выбросами. Если же почти все ячейки содержат больше,чем k точек, а какие-то ячейки содержат меньше, чем k точек, то тогда все точки, лежащие в ячейках, которые содержат менее k элементов, помечаются аномальным. Следовательно, необходимо обработать только небольшое количество ячеек, которые ранее не были помечены и только относительно небольшое количество расстояний необходимовычислить для обнаружений аномалий.

1.6.4 K ближайших соседей

Алгоритм работы метода:

— Выбирается число K-число соседей, относительно

— Устанавилается граница показателя аномальности, на основе которой будет определятся метка элемента(задается в процентах относительно среднего показателя расстояния)

— На основе метрики, рассчитывающей расстояния между элементами, высчитается расстояние между всеми элементами и всеми его соседями.

— Полученный результат сортируется на

— На основе полученных расстояний и границы показателя аномальности элементам присваиваются метки.

В качестве метки, рассчитывающей расстояния между элементами можно использовать следующую формула:

$$L = \sum_{0}^{k} x_j 0 - x_j i \tag{1.10}$$

где $x_j i$ - значение j-того атрибута элемента до которого ищется расстояние, а x_0-искомый элемент.

1.6.4.1 Методы Кнора-Реймонда и Байерса-Рейтери

Кнор и Реймонд предложили свой эффективный метод KHH подхода обучения без учителя[11]. Если m из k ближайших соседей (где m<k) лежат в пределах определенного порогового значения d, тогда считается, что данные точки лежат в достаточно плотной области распределения данных, подлежащей классификации и подлежат классификации как нормальные, в противном случае они помечаются как аномальные.

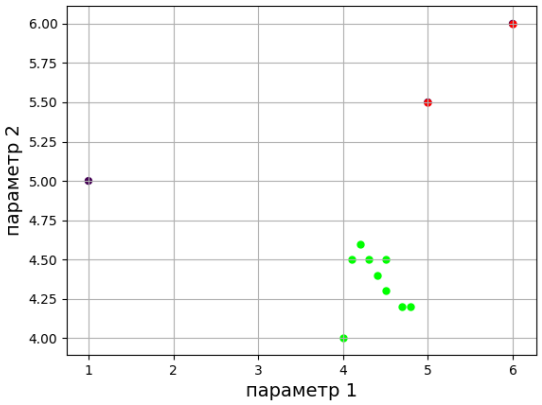

Рисунок 1.2 — Пример k-ближайших соседей

Очень похожий метод был придуман для идентефикации наземных мин на спутниковых снимках поверхности Земли Байеросом с соавторстве с Рейтери[12](этот метод можно использовать и для других целей) Он заключается в том, что берется m точек, для них ищется расстояние D_m. Если расстояние меньшего некого порогового значеня d,тогда считается, что данные точки лежат в достаточно плотной области распределения данных, подлежащей классификации и подлежат классификации как нормальные, в противном случае они помечаются как аномальные. Этот метод уменьшается количество варьируемых параметров, по сравнению с методом Кнора-Реймонда: остаются только параметры d и m, параметр k убирается. подход оригинальный подход k-NN, поскольку только k ближайших соседей должны быть вычислены для каждой точки, а не всей матрицы расстояния для всех точек

1.6.5 Метод Танга

Метод Танга заключается в вычислении средней цепочку расстояний между точкой p и k её соседями. Ранним расстояниям присваиваются более высокие веса, поэтому, если точка находится в разреженной области как черная точка на рисунке1.2, то путь до ее ближайших соседей будет относительно далеким, а среднее расстояние цепочки будет высоким. Этот метод выгодно отлчичается от вышеописанных тем, что учитывает как плотность, так и изоляцию. Рассмотрим рисокок 1.3. Очевидно, что черные точки являются аномалиями, а скопление зеленых точек - множеством "нормальных"точек. Однако, алгоритмы k-NN классификации могут столкнутся с проблемой того, что расстояние от черных точек до зеленого кластера примерно равно, значит эти точки можно отнести к одной группе и при определенных значениях параметров алгоритма эти точки не будут считаться аномалиями. Метод Танга поможет избежать таких ошибок при обнаружении выбросов. Однако метод является

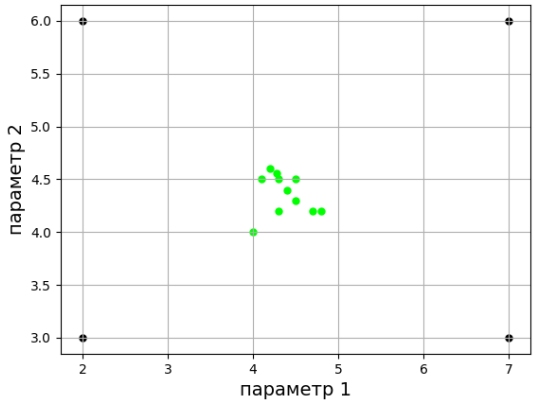

Рисунок 1.3 — Пример для метода Танга

вычислительно сложным с временем выполнения как у оринального k-NN, поскольку он полагается на вычисление путей между всеми точками и их k соседей.

1.6.6 Параметрическе методы

Вышеописанные методы плохо подходят для работы с большим объемом данных. Параметрические методы позволяют очень быстро пересчитывать модель для новых данных и подходит для больших наборов данных; модель растет только с сложностью модели, а не размером данных. Однако они ограничивают применимость, применяя предварительно выбранную модель распределения для проверки данных на аномальность. Т.е. предварительно априорно подбирается модель правдоподобности данных. Элементы , которые значительно отклоняются от этой модели считаются аномальными. Параметрический подход схож с линейным по описанию, но значительно отличается от него по принципу работы.

Одним из таких подходов является оценка эллипсоидой минимального объема[13], которая соответствует наименьшему допустимому эллипсоиду, покрывающему не меньше 50% точек выборки.

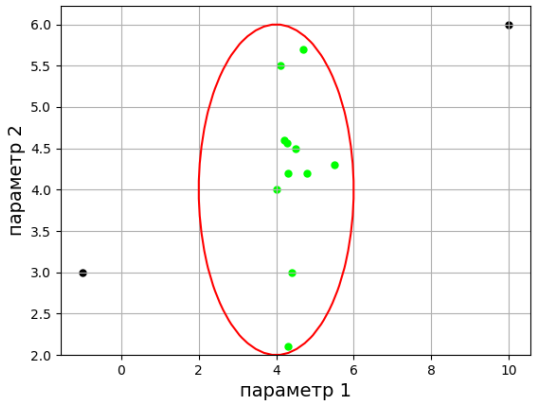

Рисунок 1.4 — Двухмерная проекция эллипсоиды минимального объема

1.6.7 Локальный коэффициент выбросов(LOF)

Этот метод является одним из самых известных алгоритмов обнаружения локальных аномалий. Недостатком метрических методов является тот факт, что все лежащие в их основе предположения верны лишь в дополнении друг с другом: локальная плотность точки, лежащей в центре небольшого кластера аномалий, может оказаться выше, чем для любой точки из большого кластера нормальных данных. Возможно и обратное: изолированная точка-аномалия может располагаться, например, в центре масс кластера нормальных точек, и тогда среднее расстояние от неё до соседей будет меньше, чем для нормальных точек. Это "свойство"метрических алгоритмов пытается учесть алгоритм локального коэффициентов выбросов(англ.Local Outlier Factor).

Чтобы вычислить LOF необходимо произвести следующие действия:

а) Для каждой записи найти всех соседей, расстояния до которых не превышает k. Их количество может быть больше, чем k.

б) Используя эти записи для каждой точки N_k, вычислить локальную плотность точки, основанную на локальной плотности достижимости(англ. local reachability density (LRD)):

$$LRD_k(x) = 1/(\frac{\sum_{o \in N_k(x)} d_k(x,o)}{|N_k(x)|}) \tag{1.11}$$

где $d_(x,o)$ расстояние достигаемости. За редким исключеним в качестве расстояния достигаемости используется евклидово расстояние [14]

в) Вычисляем LOF путем сравнения LRD записи с LRD соседей.

$$LOF(x) = \frac{\sum_{o \in N_k(x)} \frac{LRD_k(o)}{LRD_k(x)}}{|N_k(x)|} \tag{1.12}$$

Таким образом LOF является отношением локальных плотностей. Нормальные записи, плотности которых равны плотности их соседей, получают оценку около 1,0. Аномалии, которые имеют низкую локальную плотность, получат значительно более высокую оценку. Алгоритм полагаясь только на свою прямую окрестность,формирует оценку - величину, основанную основанное только на k-соседях. Конечно, глобальные аномалии также могут быть обнаружены, так как они имеют низкую LRD,по сравнению со своими соседями. Важно отметить, что в задачах обнаружения аномалий, где местные аномалии не представляют интереса, этот алгоритм будет генерировать множество ложных аномалий Настройка k имеет решающее значение для этого алгоритма.

Авторы алгоритма LOF рекомендуют использовать для вычисления k стратегию ансамблирования(алгоритм описан ниже). Берется интервал возможных значений k и с некоторым шагом для всех возможных значений k вычисляется показатели аномальности для каждого элемента выборки. Путем голосования определяется является ли эта ли эта точка аномалий. Однако, на практике такие рекомендации редко исользуют из-за их значительной вычислительной сложности.

1.6.7.1 Компонентный коэффициент выбросов(COF)

Компонентный коэффициент выбросов аналогичен LOF, но оценка плотности для записей выполняется иным способм В LOF k-ближайших соседей выбирают на основе евклидова расстояния. Это косвенно предполагает, что данные распределяются сферическим образом вокруг экземпляра. Если это допущение нарушено, например, если функции имеют прямую линейную корреляцию,то оценка плотности неверна. COF исправляет этот недостаток и оценивает локальную плотность окрестности с использованием метода кратчайшего пути, называемого расстоянием цепочки. Математически это расстояние цепочки является минимумом суммы всех

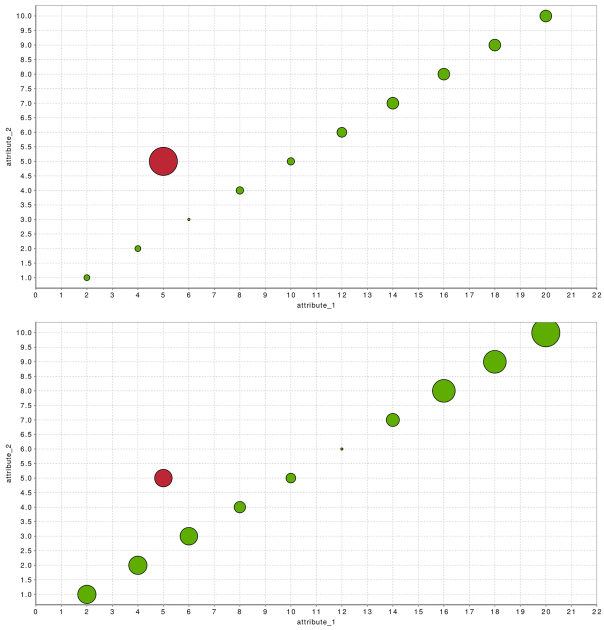

Рисунок 1.5 — Сравнение COF (сверху) с LOF (внизу) с использованием простого набора данных с линейной корреляцией двух атрибутов

расстояний, соединяющих все k соседей точки и саму точку. Например, когда функции, очевидно, коррелированы, этот подход оценки плотности работает значительно лучше [15]. На рисунке 5 показан результат для LOF и COF в сравнении для простого двумерного набора данных, где атрибуты имеют линейную зависимость. Можно видеть, что оценка плотности LOF не может обнаружить выброс, но COF удадается связать нормальные между собой для оценки локальной плотности.

1.7 Методы улучшения алгоритмов

1.7.1 Семпилрования

Большинство алгоритмов распознавания аномалий успешно работают на выборках малых размеров. Поэтому предлагается разбить начальный набор данных на несколько случайных выборок и усреднить результат. Размер этих выборок может быть как и случайным, там и фиксированного размера, но, как правило, он отличается от размеров исходного набора данных не меньше чем на порядок. Идея такого выбора заключается в том, что шумовые объекты попадут в выборки с низкой вероятностью; кластера нормальных данных будут представлены несколькими представителями, а кластера аномалий выродятся в изолированные точки. На основе этих выборок алгоритмы строят функции показателя аномальности, незначительно уступающему результату, полученному на основе анализа всех исходных данных.

Этот метод помогает значительно сократить вычислительную сложность, а так же уменьшить вероятность "подгона"алгоритма под конкретный набор данных. В силу особенностей задачи, необходимое условие отсут- ствия параметризации ал-

горитмов зачастую означает их детерминированность(в отсутсвиее стохастичности показатель аномальности однозначно определяется по заданной выборке). В общем случае при добавлении новых данных в общий набор данных, можно не пересчитывать заново показатель аномальности для всего набора данных, а добавить запуски алгоритма на новых данных в ансамбль(так называемый warm start[16])

1.7.2 Ансамблирование голосованием

Ансамблированием в задаче поиска аномалий называют использование нескольких различных алгоритмов с последующи усреднением их показателя аномальности. При использовании различных классов алгоритмов можно столкнуться с проблемой того, что показатель аномальности выглядит по-разному в различных алгоритмах и сравнить напрямую эти показатели некорректно. Поэтому традиционное приведение показателей значений различных функций к одному диапазону, например,к [0,1], будет некорректным. Существует несколько наиболее известных видов ансамблирования:

— Простое голосование

$$b(x) = F(b_1(x),...,b_T(x)) = \frac{1}{T}\sum_{t=1}^{T} b_t(x) \tag{1.13}$$

— Взвешенное голосование

$$b(x) = F(b_1(x),...,b_T(x)) = \frac{1}{T}\sum_{t=1}^{T} w_t b_t(x) \tag{1.14}$$

$$\sum_{t=1}^{T} w_t = 1, w_t \geqslant 0 \tag{1.15}$$

— Смесь экспертов

$$b(x) = F(b_1(x),...,b_T(x)) = \frac{1}{T}\sum_{t=1}^{T} w_t(x) b_t(x) \tag{1.16}$$

$$\sum_{t=1}^{T} w_t = 1, \forall x \in X \tag{1.17}$$

Простое голосование - это частный случай взвешенного голосования, а взвешенное голосование является частным случаем смеси экспертов.

Различные методы ансамблирования такие как беггинг, бустинг, стекинг и другие применяются для улучшения работы алгоритмов обучения с учителем [17]. Для алгоритмов обучения без учителя применяется простое голосование, т.к. задача изменения весов голосования нетривиальна в задаче обучения без учителя.

1.7.3 Итеративный отбор

Итеративный отбор основам на идее многократного применения алгоритмов ансамблирования. Преположим, построеная некоторая модель, описывающая нормальные данные. Эта модель построена на основе всех имеющихся данных, но точность этой модели невелика, она умеет определить только явные аномалии. Отсортировав все точки по показателю аномальности, мож- но выбрать k самых аномальных объекта в данных и исключить из данных. После этого можно перестроить модель и повторить вышеуказанные действия несколько раз, пока не будут достигнуты некоторые условия. При каждой итерации точность модели будет увеличиваться.

Идея итеративного отбора может быть обобщена различными способами. Результат работы одного алгоритма может быть использован для отсеивания явных аномалий и настройки нового алгоритма, не обязательно совпадающего с предыдущим, на оставшихся данных. х. Возможна и противоположная механика: по результатам работы одного алгоритма отбираются явные, гарантированные представители нормальных данных, и исключительно на них строится модель, их описывающая

1.8 Выводы

Существует большое число алгоритмов для нахождения аномалий. Некоторые из них опирается на априорные данные, некоторые не опираются. Для выбора подходящего алгоритма нахождения аномалий зачастую стоит учитывать характер данных, их размер и доступную априорую информацию. Несмотря на то, область знаний обнаружения аномалий активно развивается как часть современной науки, остается ещё много простора для исследования алгоритмов, модификации и создания новых.

2 Конструкторский раздел

В этом разделе приводится подробное описание разрабатываемого метода, выделяются основные его компоненты, описываются метрики, оценивающий метод. Так же приводится описание ПО, собирющее данные для анализа.

2.0.1 Метод обнаружения аномалий

В выводе аналитической части предлагается разработать новый метод обнаружения аномалий. Новый метод будет является результатом ансамблирования трех метрических методов. На вход методу подается файл с нормализованными значениями атрибутов, с фиксированным заранее известным количеством атрибутов для каждого элемента. Временная отметка элемента представляется в качестве отдельного нормализованного атрибута.

2.0.2 Локальный коэффициент выбросов

Этот метод описан в аналитической части. В результате его работы для каждого элемента набора устанавливается метка, позволяющая однозначно классифицировать принадлежность элемента к аномальным.

2.0.3 Компонентный коэффициент выбросов

Для подсчета метрики COF используется следующая формула:

$$L = \sum_{0}^{k} x_j 0 - x_j i \tag{2.1}$$

2.0.4 Ансамблирование методов

Каждый из вышеописанных методов инвариантен и иммутабелен относительно набора данных. В результате их работы получается три набора меток. На основе этих наборов формируется финвальный набор меток по следующему принципу: если элемент имеет две или более "аномальных"метки, то ему присвается "аномальная"метка, иначе - "нормальная"метка.

2.0.5 Анализ резульатов работы методов

Для проверки рабоспобности метода его нужно оценить при помощи наборов данных. Метрикой сравнения наборов данных был выбран AUC ROC — площадь под графиком, позволяющим оценить качество бинарной классификации, отображающим соотношение между долей объектов от общего количества носителей признака, верно классифицированных как несущих признак, и долей объектов от общего количества объектов, не несущих признака, ошибочно классифицированных как несущих

признак. При помощи этой метрики планируется оценить адекватность работы алго-

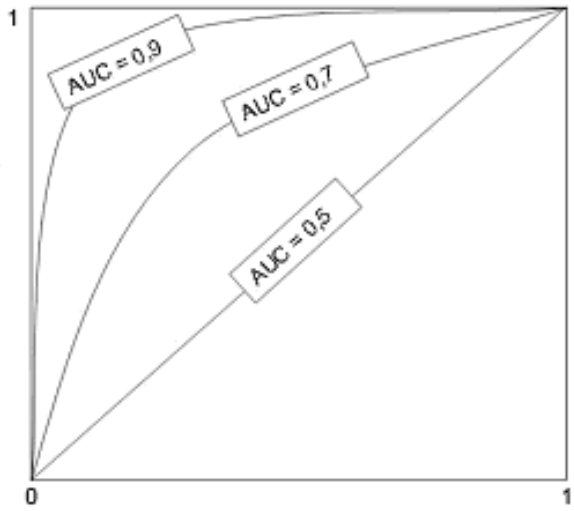

Рисунок 2.1 — Auc ROC

ритма на размеченных наборах данных и сравнить с другими методами.

2.0.6 Собирающее данные для анализа ПО

Для применения метода на практике было разработано ПО, которое позволяет собирать данные для анализа(телеметрию). В состав приложения будет входить плагин для графического редактора, backend-сервер, frontend. Так же на бекенде будет размещена база данных где будут размещаться "сырые"данные. Результатом

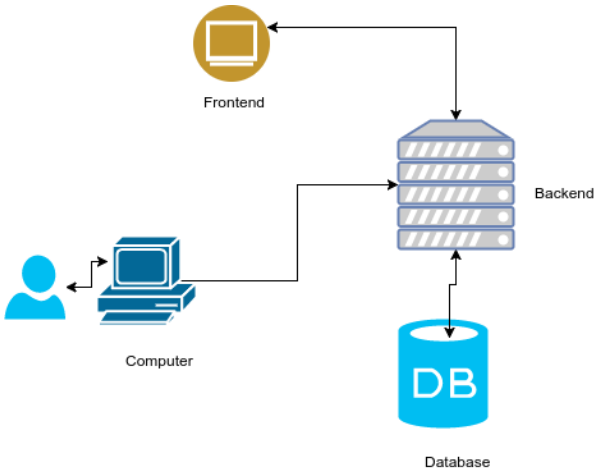

Рисунок 2.2 — Общая архитектура приложения

работы ПО будет неразмеченный набор данных, содержащий фиксированное количество атрибутов.

2.0.6.1 Собираемые данные

Основые собираемые данные приведены в таблице 2.1:

2.0.6.2 Клиентская часть

Был разработан плагин для графического редактора Krita, который позво ляет собирать телеметрию с пользователей. Телеметрия отправляется через опре деленные интервалы времени на удаленный сервер. Записи о действиях и инстру ментах отправляются каждые n минут. Информация о компьютере отправляется 1 раз при установке Krita. После этого в файл настроек записывается информация о том, что больше информацию о компьютере отправлять не стоит. Это позволяет избежать "замусоривания"отправляемой информации за счёт отсуствия дублирова ния. Информация об ассертах отправляется по мере необходимости. Если программа находится debug-режиме, программа аварийно завершает свою работу после любого ассерта. После того как аварийный ассерт сработал, он записывается в файл конфига Krita. При следующей запуске программы, он будет прочитан оттуда и отправлен на удаленный сервер при старте программы. Если программа собрана в release-версии, то при ассерте, не приводящем к аварийному завершению программы, информация об ассерте в рантайме программы отправляется на удаленный сервер. Собираемые метрики агрегируются на стороне клиента в http-запрос. Пример тела запроса пред ставляет собою JSON, пример этого JSON приведен ниже(форматирование нарушено в целях наглядности).

Листинг 2.1 — Тело http-запроса

```
1 {
2 "Name": [
3 {
4 "Param1": 1,
5 "Param2": 4581,
6 "Timestamp": 12214
7 },
```

2.0.6.3 Серверная часть

На сервере метрики обработываются и заносятся в JSON виде "как есть". Сервер умеет отдавать метрики в нормализованном виде за опредленный промежу ток времени, а так же

2.0.6.4 Динамическое изменение данных

Инструменты и действия(actions) могут меняться достаточно часто в про цессе разработки графическего редактора. Поэтому неразумно задавать статически эти метрики в коде. В коде бекенд-сервера реализована поддержка добавления но вых элементов. Раз в сутки просыпается новая горутина(легковесный тред), которая пробегается по базе данных и ищет новые инструменты и действия. После этого

она записывает их в текстовый файл. Подобное разумно применять не только для инструментов и действий, но и для любых часто изменяющихся данных. В будущем возможно расширения этой системы.

Таблица 2.1 — Описание собираемых данных

Информация о	Источник данных	Данные
Инструменты	KisTool::activate, KisTool:deactivate	CountUse float64 Time float64 ToolName string Timestamp float64
Действия	KisMainWindows actioncollection()	CountUse float64 TimeUse float64 ActionName string
Свойства изображений	KisDocument::saveFile()	ColorProfile string ColorSpace string Height float64 Width float64 Size float64 NumLayers float64 Timestamp float64
Информация о компьютере	KisDocument::saveFile()	AppVersion string CompilerVersion string CompilerType string CpuArchitecture string CpuCount float64 CpuFamily float64 CpuIsIntel bool CpuModel float64 LocaleLanguage string OpenGlGlslVersion string OpenGlRenderer string OpenGlVendor string PlatformOs string PlatformVersion string QtVersion string ScreenDpi string ScreenHeight string ScrennWidth string Timestamp float64
Ассерты	kis_assert_common()	AssertFile string AssertLine float64 AssertText string Count float64 IsFatal bool Timestamp float64

3 Технологический раздел

НЕ ГОТОВО

3.1 Клиентская часть

3.1.1 Библиотека KUserFeedback

В качестве вспомогательной библиотеки для сбора стастики используется библиотека KUserFeedback компании KDAB. Эта библиотека включает в себя C++ Qt клиентскую часть, а так же сервер, написанный на PHP[?] Нам не требуется их сервер и значительная часть функций, мы будем использовать только часть собирающую телеметрию. KUserFeedBack позволяет собрать библиотеку по частям и линковать к нашему приложению только необхоимые модули. В программе используется модуль Core. Модуль Core содержит абстракный класс источника данных, от которого можно наследовать различные источники данных. Ниже представлено описание этого класса. В дальнейшем мы создадим классы, которые наследуются от этого абстрактого класса. В этих классах будут переопределеные все чисто виртуальные функции. Ключевую роль будет играть переопределение функции *data()* - именно это переопределение несет смысловую нагрузку класса.

Листинг 3.1 — Описание базового класса библиотеки

```
1  class KUSERFEEDBACKCORE_EXPORT AbstractDataSource
2  {
3  public:
4  virtual ~AbstractDataSource();
5  QString name() const;
6  virtual QString description() const = 0;
7  virtual QVariant data() = 0;
8  virtual void load(QSettings *settings);
9  virtual void store(QSettings *settings);
10 virtual void reset(QSettings *settings);
11 Provider::TelemetryMode telemetryMode() const;
12 void setTelemetryMode(Provider::TelemetryMode mode);
13
14 protected:
15
16 explicit AbstractDataSource(const QString &name, Provider::TelemetryMode
       mode = Provider::DetailedUsageStatistics, AbstractDataSourcePrivate *dd
       = nullptr);
17 void setName(const QString &name);
18 class AbstractDataSourcePrivate* const d_ptr;
19 };
20 }
```

4 Исследовательский раздел

4.1 Время дизеринга раличных алгоритмов

Рассмотрим время работы различных алгоритмов для различных размеров изображения.

	Размер, пиксели	Время, мкс
White noise	133x90	862
Blue noise	133x90	930
Brown noise	133x90	934
Violet noise	133x90	937
Pink noise noise	133x90	930
Floyd-SD	133x90	1200
F. Floyd-SDe	133x90	1093
JJN	133x90	1909
White noise	458x458	15735
Blue noise	458x458	19374
Brown noise	458x458	19432
Violet noise	458x458	18787
Pink noise noise	458x458	18129
Floyd-SD	458x458	27173
F. Floyd-SDe	458x458	26424
JJN	458x458	47201
White noise	458x458	194376
Blue noise	458x458	200577
Brown noise	458x458	208400
Violet noise	458x458	251294
Pink noise noise	458x458	258775
Floyd-SD	458x458	251294
F. Floyd-SDe	458x458	387104
JJN	458x458	857481

Из рассмотрения вынесены алгоритм Юлиомы в вследствие того, что он значительно медленней других алгоритмов(2732568 мкс для изображения 113x90) в и алгоритм Байера, реализованный при помощи шейдеров, вследствии того, что он не не укладывается в рамки требуемой палитры (при этом он работает очень быстро 64 мс для изображени 640x480).

4.2 Качество получаемого изображения

	PSNR	SSIM
White noise	33.2894	0.914778
Blue noise	36.1756	0.971626
Brown noise	33.32370	0.915767
Violet noise	37.63480	0.984574
Pink noise	36.4484	0.974718
Floyd-SD	37.0553	0.979173
F. Floyd-SDe	36.8401	0.976452
JJN	37.30740	0.981688
Yliouma	36.2359	0.967796
Without dithering	37.6348	0.984574

Несмотря на то, что некоторые сложные алгоритмы дизеринга диффузии ошибок обещат получения хорошего качества изображений,некоторые алгоритмы случайного дизеринга на конкретных изображениях дают лучший результат. Для того чтобы получить наилучший результат дизеринга, следует проанализировать результаты дизеринга нескольких изображений и выбрать среди них наилучшее. Так же следует отметить некоторую необъективность метрик: результат метрик не всегда совпадает с человеческим восприятием картинки.

4.3 Размер получаемого изображения

	Разрешение, пикс	Размер, кб	Исх. раз., кб
White noise	900x675	186	2373 bmp,1779 png
Blue noise	900x675	135	
Brown noise	900x6750	186	
Violet noise	900x675	98	
Pink noise	900x675	1158	
Floyd-SD	900x675	1273	
F. Floyd-SDe	900x675	143	
JJN	900x675	117	
White noise	3984x32355	3431	50344 bmp,37758 png
Blue noise	3984x3235	2570	
Brown noise	3984x3235	3432	
Violet noise	3984x3235	1950	
Pink noise	3984x3235	2406	
Floyd-SD	3984x32355	3605	
F. Floyd-SDe	3984x3235	4269	
JJN	3984x3235	3716	

Из вышеприведенной таблицы, можно заметить, размер изображения после дизеринга значительно уменьшиается,достирается выигрыш в размере изображение до 15 раз, в зависимости от иходного контейнера изображения и выбранного способра дизеринга.

Заключение

В данной работе были реализованы различные алгоритмы дизеринга, было произведено сравнение и анализ этих алгоритмов. Программа позволяет получить изображение схожего визуального качества при значительном уменьшении размера. Был получен вывод о том, что для различных целей следует использовать различные алгоритмы дизеринга, универсального алгоритма дизеринга не существует. Программа не привязана к какой-то конкретной операционной системе и может быть скомпилирована и запущена на всех популярных ОС.

Список использованных источников

1. *machinelearning.ru/*. Выборка. https://goo.gl/7gjJ6p.

2. *определения, ГОСТ 20886-85: Организация данных в системах обработки данных. Термины и.* http://www.gostrf.com/normadata/1/4294832/4294832686.pdf.

3. *Википедия*. https://en.wikipedia.org/wiki/Data_set.

4. *Дьяконов, Александр*. Поиск аномалий (Anomaly Detection) / Александр Дьяконов. — 2017. https://goo.gl/Z43Ne9.

5. *F.E., Grubbs*. Procedures for Detecting Outlying Observations in Samples. Technometrics / Grubbs F.E. — 1969.

6. *Moya M.M., Hush D.R.* Network Constraints and Multi-objective Optimization for One-class Classification. Neural Networks / Hush D.R. Moya M.M. — 1996.

7. *Chandola V Banerjee A, Kumar V.* Anomaly Detection: A Survey / Kumar V. Chandola V, Banerjee A. — ACM Computing, 2009.

8. *Goldstein M, Uchida S.* Behavior Analysis Using Unsupervised Anomaly Detection / Uchida S. Goldstein M. — The 10th Joint Workshop on Machine Perception and Robotics, 2014.

9. *Андрей, Гахов*. Интеллектуальный анализ данных / Гахов Андрей. — Харьковский национальный университет имени В.Н. Карамзина, 2014.

10. *Hodge V., Austin J.* A survey of outlier detection methodologies / Austin J. Hodge V. — Artificial intelligence review, 2004.

11. *Knox, Edwin M.* Algorithms for Mining Distance-Based Outliers in Large Datasets / Edwin M. Knox, Raymond T. Ng. — University of British Columbia, 1998.

12. *S. Bayers, A.Raftery*. Nearest Neighbor Clutter Removal for Estimating Features in Spatial Point Processes / A.Raftery S. Bayers. — Journal of the American Statistical Association, 1998.

13. *Rousseeuw, Leroy*. Robust Regression and Outlier Detection / Leroy Rousseeuw. — John Wiley and Sons, 1996.

14. *Goldstein, Markus*. A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data / Markus Goldstein. — Seiichi Uchida, 2016.

15. *M., Goldstein*. Anomaly Detection in Large Datasets / Goldstein M. — University of Kaiserslauterna, 2014.

16. *B.Chu Chia-Hua Ho, Cheng-Hao Tsa*. Warm Start for Parameter Selection of Linear Classifiers / Cheng-Hao Tsa B.Chu, Chia-Hua Ho. — National Taiwan University, 2015.

17. *Александр, Гущин.* Методы ансамблирования обучающихся алгоритмов / Гущин Александр. — Московский физико-технический институт, 2015.