Реферат

Отчёт 80 страниц, 35 рисунков, 30 таблиц, 38 формул, 12 источников.

АНОМАЛИИ, ВЫБРОСЫ, ПОИСК АНОМАЛИЙ, АНАЛИЗ ДАННЫХ, СТАТИСТИКА, ПОСЕЩАЕМОСТЬ САЙТА, СРЕДНЕКВАДРАТИЧНОЕ ОТКЛОНЕНИЕ, ФАКТОР ЛОКАЛЬНОГО ОТКЛОНЕНИЯ, LOCAL OUTLIER FACTOR

Объектом исследования являются статистические данные посещаемости интернет-ресурса.

Цель работы - поиск и разработка метода, способного на основе статистических данных сайта за предыдущие периоды времени, выявлять и проводить анализ аномалий в текущих статистических данных.

В процессе работы были рассмотрены существующие методы статистики и кластеризации, которые применимы для решения данной проблемы. Также был проведён анализ с целью выявления дополнительных данных, способствующих достижению цели.

В результате были выбраны два метода: фактор локального отклонения (local outlier factor) и метод, основанный на среднеквадратичном отклонении. Была выявлена значимость данных о скорости изменения данных посещаемости. Проведено тестирование работы методов, а также метода, основанного на совместном применении выбранных методов.

Эффективность совместного использования методов определяется уменьшением количества выявленных лжеаномалий. Для улучшения результативности поиска аномалий в статистических данных необходимо использовать выявленные данные о скорости изменения рассматриваемых данных.

Содержание

Введение

С появлением первых носителей информации, начался процесс беспрерывного накопления знаний. Рост информационных технологий, прогресс в области электронных носителей, технологий передачи и обмена данными скорость притока информации только увеличилась.

Постоянный рост накопленного объёма данных обуславливает рост актуальности обработки этих данных. В связи с этим появились такие терминологии, как «большие данные» (англ. Big Data), «анализ данных» (англ. Analysis of data) и «интеллектуальный анализ данных» (англ. Data Mining).

Термин «большие данные» характеризует совокупности данных с возможным экспоненциальным ростом, которые слишком велики, слишком фрагментированы или слабо структурированы для анализа традиционными методами. [1]

«Анализ данных» - область математики и информатики, занимающаяся построением и исследованием наиболее общих математических методов и вычислительных алгоритмов извлечения знаний из экспериментальных данных. [2]

«Интеллектуальный анализ данных» - собирательное название, используемое для обозначения совокупности методов обнаружения в данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности. [3]

Основными направлением интеллектуального анализа данных является извлечение и применение практических данных. Классы методов, используемые в этом направлении, включают в себя решение задач классификации, кластеризации, прогнозирования и др.

Один из классов задач является задача определения отклонений или выбросов (англ. Deviation Detection). Решение данной задачи позволяет обнаружить и проанализировать данные, наиболее отличающиеся от общего множества данных - выявление нехарактерных шаблонов.

Методы Решение данной задачи может применяться во многих областях, требующих оперативного выявления отклонений от какого-либо типичного сценария получения данных. Например, к таким областям может относиться безопасность клиента банка (резкое увеличение сумм транзакций может говорить о захвате данных клиента злоумышленником). Постоянный рост поступающей информации, увеличение уровня безопасности и контроля обуславливает потребность в решении данного класса задач.

Также к таким областям относится контроль электронного ресурса в сети. Если у пользователя есть интернет-ресурс и информация о посещаемости этого ресурса, то

резкое падение показателей может свидетельствовать об отказе оборудования или других технических неполадках.

1. Аналитический раздел

1.1. Постановка задачи

Целью данной работы является поиск и разработка метода, способного на основе статистических данных сайта за предыдущие периоды времени, выявлять и проводить анализ аномалий в текущих статистических данных.

В данной работе рассматривается выявление и анализ аномалий на примере статистических данных посещаемости сайта.

Для достижения поставленной цели необходимо выполнить следующие основные задачи:

- исследовать явление аномалий в статистических данных,
- выполнить анализ существующих алгоритмов для выявления аномалий,
- разработать метод анализа данных, являющийся комбинацией существующих алгоритмов для выявления аномалий и позволяющий выявлять аномалии в статистических данных,
- протестировать работоспособность метода,
- найти возможности повышения результативности метода,
- исследовать работу методов на различных видах аномалий и среднесуточной посещаемости сайта.

В результате приведённого анализа будут сформулированы требования для разработки приложения.

1.2. Анализ аномалий

Явление аномалий широко распространено в различных областях деятельности. В статистике аномалии в данных называют выбросами (англ. outlier).

1.2.1. Выброс

Выброс - результат измерения, выделяющийся из общей выборки.

Причины выбросов:

- Ошибки измерения
- Необычная природа входных данных
- Выбросы являются частью распределения

Выбросами могут быть данные, которые резко отличаются от общей выборки. Такие данные могут быть максимумом или минимумом в заданной выборке. Но максимум и минимум в выборке не обязательно будут выбросами.

Методы статистики, которые устойчивы к выбросам, называют робастными. Результатами неробастных методов могут быть искажённые результаты. Например, медиана является робастной характеристикой, в то время как выборочное среднее – нет.

Ещё одной проблемой выбросов является отсутствие строгого математического определения выброса. Результат классификации наблюдения как выброса является очень субъективным. Такая оценка, как правило, проводится экспертом в области, сопряжённой с получением оцениваемых данных.

Также наличие совокупности выбросов может привести к изменению оценки общей выборки, так как увеличивается вероятность классификации выбросов как нормальных данных.

1.2.2. Аномалии посещаемости сайта

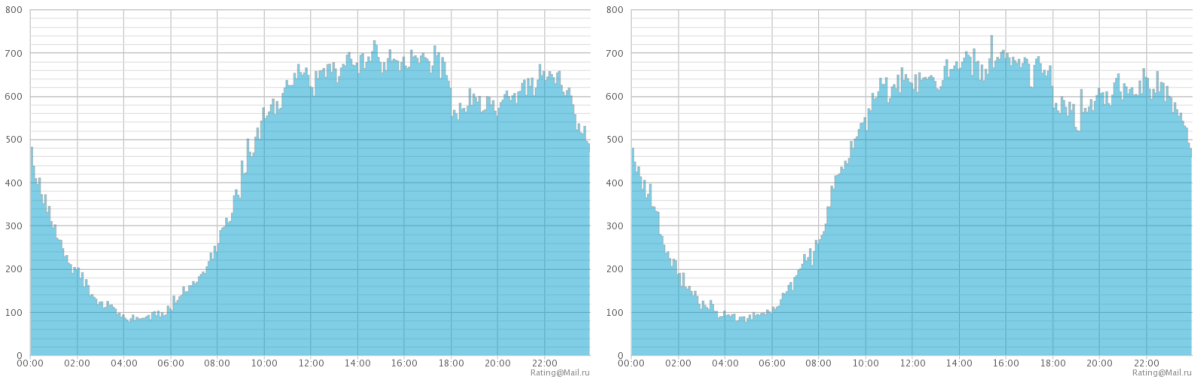
Аномалиями в посещаемости сайта являются выбросы, причинами которых является необычная природа входных данных.

В данной работе в качестве входных данных рассматриваются данные о количестве пользователей, которые перешли на определённый интернет-ресурс, в определённые интервалы времени.

Выявление аномалий в таких данных также является нетривиальной задачей. Рассмотрим задачу определения аномалий на примере сайта http://forum.ixbt.com/ (форум iXBT.com) со средней посещаемостью сайта 65.000 посетителей в день и сайта http://articles.org.ru/ (каталог программиста) со средней посещаемостью 500 посетителей в день.

Рассмотрим данные о посещаемости сайта http://forum.ixbt.com/ за 2 апреля 2014 года (рисунок 1а). Данные представляют собой среднюю количественную характеристику посещаемости сайта за интервалы времени, равные 5 минутам. В отрезок времени с 17:55 до 18:05 можно наблюдать резкое падение посещаемости. Однако среднее количество посетителей меняется всего с показателя 650 до 570 в течение 10 минут, что можно трактовать не только как аномалию, но и как естественный поведение (например, люди уходят с работы, выключают компьютер, закрывают форум). Также, учитывая данные за 10 апреля 2014 года (рисунок 1б), и схожее падение

посещаемости в рассматриваемый период времени, предположение о естественном поведении становится более вероятным.



а б

Рисунок 1 - график посещаемости сайта http://forum.ixbt.com/

а) за 2 апреля 2014 года; б) за 10 апреля 2014 года.

Таким образом, при естественном поведении, можно выделить схожие данные за одинаковые интервалы времени в различные дни. При этом аномальные значения будут определяться преимущественно скоростью изменения данных о посещаемости.

Рассмотрим данные о посещаемости сайта http://articles.org.ru/ за 12 апреля (рисунок 2). В период времени 15:55 до 16:00 можно наблюдать резкий рост посещаемости. Однако, вследствие низкой посещаемости ресурса в целом, такое поведение нельзя рассматривать как аномалию.

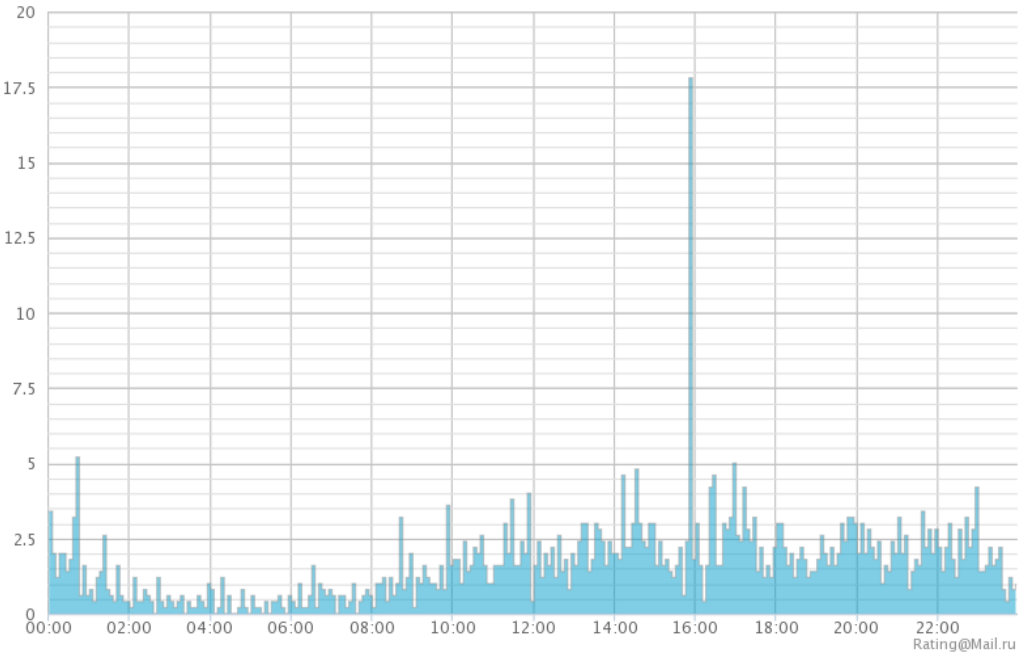


Рисунок 2 - график посещаемости http://articles.org.ru/ за 12 апреля 2014 года.

В связи с этим, для выявления аномалий в данных с низкой посещаемостью целесообразнее использовать данные преимущественно о посещаемости ресурса.

1.3. Исследование существующих методов анализа данных

В связи с выявленными зависимостями аномальности данных от признаков аномалии, будут рассмотрены следующие методы, основываясь на которых можно будет выявлять аномалии.

В контексте анализа данных, следующие методы будут рассмотрены с учётом таких характеристик, как точность и полнота данных. Точность данных – это доля выборки, которая действительно принадлежит искомому классу данных относительно всей выборки, которую метод отнес к этому классу. Полнота данных – это доля найденных данных, принадлежащих искомому классу, относительно всех аномалий в заданной выборке.

1.3.1. Методы статистики

Методы, рассматриваемые в этом разделе, являются методами прикладной статистики для анализа статистических данных.

Межквантильное расстояние

Метод, основанный на межквантильном расстоянии, применяется для исключения выбросов из выборки. В этом случае, выбросами считаются данные, которые не попадают в межквартильный диапазон:

$$[Q_1 - k(Q_3 - Q_1), Q_3 + k(Q_3 - Q_1)] \quad (1)$$

где Q_1 – нижний квартиль, Q_3 – верхний квартиль, K – положительная константа.

Квартили – условные значения, которые разделяют вариационный ряд на две части. 25% значений меньше нижнего квартиля и 75% значений меньше верхнего квартиля [4]. Результативность данного метода достигается за счёт увеличения полноты аномальных данных за счёт снижения точности.

Преимущества метода – аномалии будут лежать за пределами межквартильного диапазона.

Недостатки метода – широкий диапазон значений, лежащих за пределами межквартильного расстояния, будет ошибочно классифицирован как диапазон аномальных значений.

Среднеквадратичное отклонение

Среднеквадратичное отклонение - показатель рассеивания значений случайной величины относительно её математического ожидания. Определяется как квадратный корень из дисперсии случайной величины.

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad (2)$$

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (3)$$

где σ - среднеквадратичное отклонение, n – размер выборки, x_i – случайная величина, $\bar{x}$ – математическое ожидание заданной выборки.

Большое значение среднеквадратического отклонения показывает большой разброс значений в представленном множестве со средней величиной множества выборки. Маленькое значение, соответственно, показывает, что значения в множестве сгруппированы вокруг среднего значения.

Со среднеквадратичным отклонением нормального распределения связано правило трёх сигм. Согласно этому правилу приблизительно с 0,9973 вероятностью значение нормально распределённой случайной величины лежит в интервале ($\bar{x}$ – 3σ, $\bar{x}$ + 3σ). Использование данного правила совместно с предположением о том, что закон распределения данных о посещаемости сайта близок к нормальному закону распределения, может точно выделить данные, которые являются аномалиями. В отличии от метода, связанного с межквартильным расстоянием, происходит увеличение точности данных, классифицируемых как аномалии, за счёт снижения полноты. Так как признаками аномалий являются резкие изменения в данных, то полнота будет достаточной для покрытия резких перепадов.

Преимущества метода – данные, которые будут лежать за пределами диапазона трёх сигм, с наибольшей вероятностью окажутся аномалиями. Так же, можно корректировать интервал в соответствии с определённой степенью покрытия.

Недостатки метода – закон распределения выборки должен быть близок к нормальному закону распределения.

Линии Боллинджера

Линии Боллинджера - инструмент технического анализа, активно применяющийся в исследовании финансовых рынков, отражающий текущие отклонения цены акции, товара или валюты [5].

Данный метод помогает оценить, как расположены цены относительно нормального торгового диапазона. Линии Боллинджера создают рамку, в пределах которой значения считаются нормальными. Линии Боллинджера строятся в виде верхней и нижней границы вокруг скользящей средней, при этом ширина полосы пропорциональна среднеквадратическому отклонению от скользящей средней за анализируемый период времени.

Скользящая средняя – функция, значения которой в каждой точке определения равны среднему значению исходной функции за предыдущий период. Скользящие средние обычно используются с данными временных рядов для сглаживания краткосрочных колебаний и выделения основных тенденций или циклов.

В линиях Боллинджера используется простая скользящая средняя, рассчитанная относительно данных за предыдущие периоды (обычно рассматривается 20-тикратный период времени). В вычислении верхней и нижней границы скользящей средней используется среднеквадратичное отклонение за тот же период времени, что и простая скользящая средняя (коэффициент пропорциональности обычно равен 2).

Рисунок 3 - пример линий Боллинджера [6].

Данный метод сигнализирует о выходе данных за пределы допустимого размаха колебаний данных. Чем больше резких изменений данных происходит, тем шире будет допустимый размах. Изменением коэффициента пропорциональности ширины полосы можно влиять на полноту данных, классифицируемых как нормальные. При этом, чем выше данный коэффициент, тем выше вероятность, что данные, которые вышли за пределы допустимого размаха – аномалии, то есть повышается точность.

Преимущества метода – настраиваемый и адаптированный к данным, которые характеризуются резкими перепадами.

Недостатки метода – неспособность учёта данных за более давние периоды времени.

Расстояние Махаланобиса

Расстояние Махаланобиса - мера расстояния между векторами случайных величин. С помощью расстояния Махаланобиса можно определять сходство неизвестной и известной выборки. Оно отличается от расстояния Евклида тем, что учитывает корреляции между переменными. Расстояние Махаланобиса также широко используется в кластерном анализе и методах классификации[7].

Формально, расстояние Махаланобиса от многомерного вектора $x = (x_1, x_2, ..., x_n)^T$ до множества со средним значением $\mu = (\mu_1, \mu_2, ..., \mu_n)^T$ и матрицей ковариации S определяется следующим образом

$$D_M(x) = \sqrt{(x-\mu)^T S^{-1} (x-\mu)}. \quad (4)$$

Матрица ковариации - это матрица, составленная из попарных ковариаций элементов одного или двух случайных векторов. Ковариация - математическое ожидание произведения случайных величин $X = x - MX$ и $Y = y - MY$ [8]. Пусть $X_1, X_2, ..., X_n$, $Y_1, Y_2, ..., Y_n$ — выборки $X_{(n)}$, $Y_{(n)}$ случайных величин, определённых на одном и том же вероятностном пространстве. Тогда ковариацией между выборками $X_{(n)}$ и $Y_{(n)}$ является:

$$\operatorname{cov}(X_{(n)}, Y_{(n)}) = \frac{1}{n} \sum_{t=1}^{n} \left(X_t - \overline{X}\right) \left(Y_t - \overline{Y}\right) \quad (5)$$

В общем случае, расстояние Махаланобиса используется для определения расстояния между точкой в N-мерном пространстве и центром множества, заданного набором точек в этом N-мерном пространстве. Если расстояние между заданной точкой и центром масс меньше среднеквадратичного отклонения, то можно заключить, что вероятность принадлежности точки множеству высока. Чем дальше точка, тем больше вероятность того, что она не принадлежит множеству. Если множество не сферическое, то оно может быть задано матрицей ковариаций множества.

Чтобы использовать расстояние Махаланобиса в задаче определения принадлежности заданной точки одному из N классов, нужно найти матрицы ковариации всех классов. Как правило, это делается на основе известных выборок из каждого класса (в рамках поставленной задачи, класса аномалий и класса нормальных данных). Затем необходимо рассчитать расстояние Махаланобиса от заданной точки до каждого класса и выбрать класс, для которого это расстояние минимально.

Преимущества метода – учитывает сложность множества.

Недостатки метода – необходимость иметь известные выборки для каждого класса, необходимость представления данных, как точек в N-мерном пространстве.

Тест Диксона (Dixon's Q test)

Тест Диксона предназначен для выявления и отбрасывания аномальных значений на небольшой выборке. Этот тест применяется один раз на одно множество данных. Для выборки, отсортированной по возрастанию, для каждой пары ближайших соседей вычисляется значение Q:

$$Q = \frac{\text{gap}}{\text{range}} \quad (6)$$

где range – разница между наибольшим и наименьшим значением выборки, gap – разница между парой ближайших соседей выборки, отсортированной по возрастанию.

Каждое значение Q сравнивается с табличным Q в зависимости от степени надёжности, требуемой для определения аномального значения. Например, существует выборка из 10 элементов, в которой для одной из пар значений Q = 0.455. Согласно таблице 1, для степени доверия равном 90%, значение из данной пары будет аномальным (0.455 > 0.412). Для степени доверия в 95%, данное значение будет считаться нормальным (0.455 < 0.466).

Таблица 1 - критические значения теста Диксона для различной степени доверия. [9]

Q\n	4	5	6	7	8	9	10
$Q_{90\%}$:	0.765	0.642	0.560	0.507	0.468	0.437	0.412
$Q_{95\%}$:	0.829	0.710	0.625	0.568	0.526	0.493	0.466
$Q_{99\%}$:	0.926	0.821	0.740	0.680	0.634	0.598	0.568

Преимущества метода – простота и возможность получения результатов с определённой степенью доверия.

Недостатки метода – метод эффективен только для небольшой выборки (n < 30) [9].

1.3.2. Методы кластеризации

Задача кластеризации – задача, в которой имеется множество объектов, разделённых некоторым образом на классы, которые изначально не заданы. Требуется построить алгоритм, способный классифицировать произвольный объект из исходного множества.

В данном разделе будут рассмотрены методы кластеризации, которые относятся к классу методов «обучения без учителя» (т.е. методов, выполняющих поставленную задачу, без вмешательства со стороны экспериментатора) и которые будут способствовать решению исходной задачи.

Формальная постановка задачи кластеризации

Пусть X — множество объектов, Y — множество номеров (имён, меток) кластеров. Задана функция расстояния между объектами p(x, x'). Имеется конечная обучающая выборка объектов $X^m = x_1, ..., x_m$ принадлежащая X. Требуется разбить выборку на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из объектов, близких по метрике p, а объекты разных кластеров существенно отличались. При этом каждому объекту x_i принадлежащему X^m приписывается номер кластера y_i.

Алгоритм кластеризации — это функция $a : X \to Y$, которая любому объекту x принадлежащему X ставит в соответствие номер кластера y принадлежащему Y. Множество Y в некоторых случаях известно заранее, однако чаще ставится задача определить оптимальное число кластеров, с точки зрения того или иного критерия качества кластеризации.

Цели кластеризации

В зависимости от прикладной задачи различают следующие цели кластеризации[10]:

- Понять структуру множества объектов, разбив его на группы схожих объектов. Упростить дальнейшую обработку данных и принятия решений, работая с каждым кластером по отдельности.
- Сократить объём хранимых данных в случае сверхбольшой выборки, оставив по одному наиболее типичному представителю от каждого кластера.
- Выделить нетипичные объекты, которые не подходят ни к одному из кластеров.

Для решения поставленной задачи необходим бинарный алгоритм, в котором множество Y состоит из двух типов кластеров – нормальных значений и аномальных.

Метод k-средних (k-means)

Метод k-средних является одним из наиболее популярных алгоритмов кластеризации. Идея алгоритма состоит в том, чтобы минимизировать суммарное квадратичное отклонение точек кластеров от центров масс этих кластеров:

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2 \quad (7)$$

где V – целевая функция, k – количество кластеров, S – множество кластеров, μ_i - центр масс кластера, x_j – множество векторов, принадлежащих кластеру S.

Алгоритм разбивает множество элементов на заранее заданное число кластеров k. На каждой итерации алгоритма происходит вычисление центров масс кластеров, полученных на предыдущей итерации и новое разбиение множества на кластеры по наименьшему расстоянию до рассчитанного центра масс кластеров. Алгоритм завершается, когда на какой-то итерации центр масс кластеров не изменился.

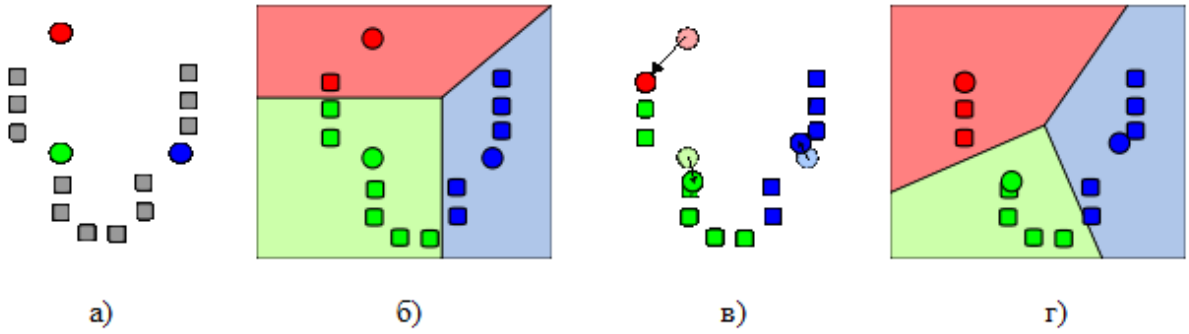


Рисунок 4 - демонстрация работы алгоритма: а) выбираются произвольные центры кластеров; б) разбиение элементов множества на кластеры; в) вычисление нового центра масс; г) новое разбиение на кластеры.

Существуют различные модификации данного алгоритма (например, k-medians, k-means++).

Преимущества метода – алгоритм прост в реализации.

Недостатки метода – зависимость результата работы алгоритма от выбора исходных центров кластеров, их оптимальный выбор для поиска аномалий неизвестен. Неизвестно количество кластеров, необходимых для поиска аномалий.

Метод с-средних (c-means)

Метод с-средних относится к классу нечётких алгоритмов. В результате разбиения объектам выборки соответствует определённая степень принадлежности к каждому из кластеров. Таким образом, все точки принадлежат всем кластерам (возможно, с нулевой степенью принадлежности). Идея алгоритма состоит в том, чтобы минимизировать сумму

$$E = \sum_{j=1}^{k} \sum_{x_i \in P} u_{i,j}^{m} ||x_i - c_j|| \quad (8)$$

где E – целевая функция, P – множество объектов, k – количество кластеров, c_j - центр jого кластера.

В начале работы алгоритма случайным образом задаются центры кластеров. На каждой итерации алгоритма в каждом кластере для каждого объекта высчитывается степень принадлежности данному кластеру в зависимости от расстояния от объекта до центра этого кластера и суммы расстояний от объекта до каждого центра кластеров. Для каждого кластера, на основе степени принадлежности каждой точки к этому кластеру, рассчитывается центр этого кластера. Алгоритм продолжает работу, пока смещение центра каждого кластера не будет превышать определённого значения.

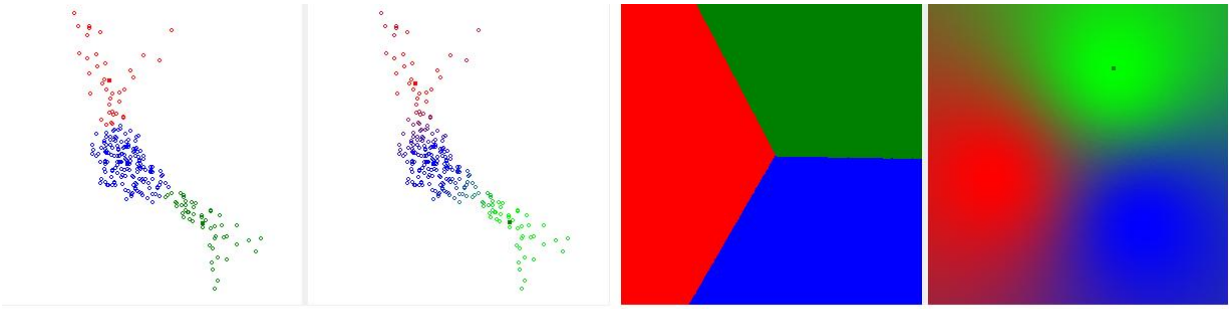


Рисунок 5 - результаты кластеризации k-means(а, в) и c-means(б, г) на 500 (а, б) и 150000 (в, г) объектов соответственно.

Преимущества алгоритма – получение значений степени принадлежности, поддающихся анализу.

Недостатки метода – сложность интерпретации полученных значений.

Использование данного метода в решении задачи поиска аномалий возможно при использовании в выборке значений, определённо характеризуемых как аномальные, для того, чтобы установить пороговое значение принадлежности объекта к кластеру, который будет выделен как аномальный (рисунок 6).

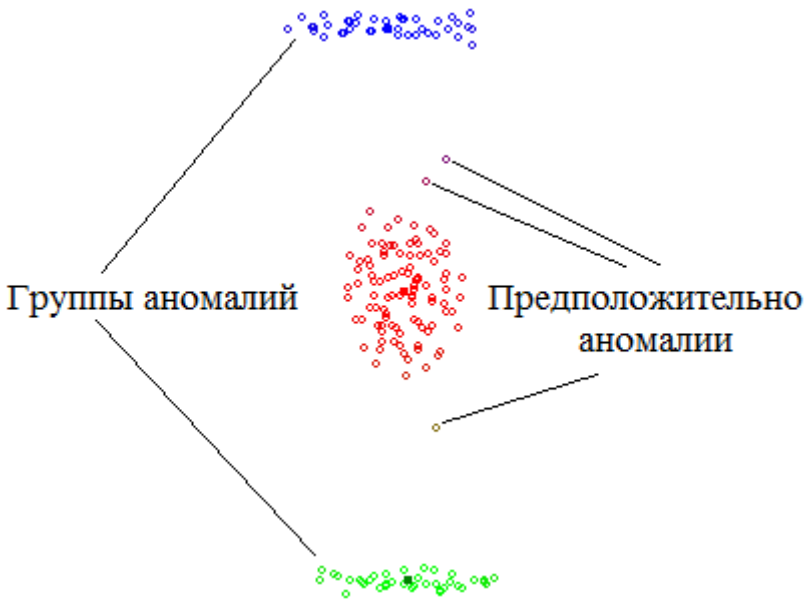


Рисунок 6 - кластеризация выборки с наличием двух различных типов аномальных данных.

Алгоритм FOREL

FOREL - алгоритм кластеризации, основанный на идее объединения в один кластер объектов в областях их наибольшего сгущения. Цель такой кластеризации - разбить выборку на такое число таксонов, чтобы сумма расстояний от объектов кластеров до центров кластеров была минимальной по всем кластерам. Задача состоит в том, чтобы выделить группы максимально близких друг к другу объектов, которые в силу гипотезы схожести и будут образовывать наши кластеры.

$$F = \sum_{j=1}^{k} \sum_{x \in K_j} \rho(x, W_j), \quad (9)$$

где F – целевая функция, k – количество кластеров, x – объект, принадлежащий кластеру, p – функция расстояния.

Для работы алгоритма необходимо задавать R - радиус поиска локального сгущения объектов. На каждой итерации алгоритма случайным образом выбирается объект из выборки, находим все объекты, лежащие в пределах сферы радиуса R с центром в этом объекте, внутри этой сферы выбираем центр тяжести и делаем его центром новой сферы. Таким образом, на каждом шаге происходит сдвиг сферы в сторону локального сгущения объектов выборки. После того как центр сферы стабилизируется, все объекты внутри сферы будут принадлежать кластеру и будут удалены из выборки. Этот процесс повторяется до тех пор, пока вся выборка не будет кластеризована.

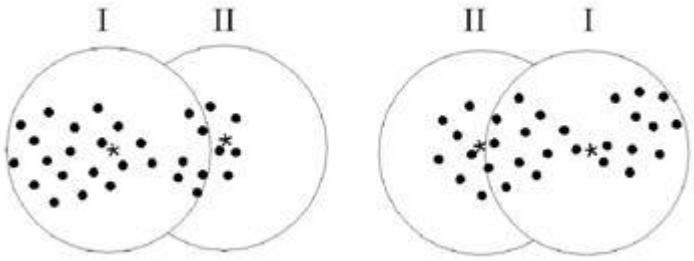


Рисунок 7 - пример разбиений множества на два кластера.

Существуют различные подходы к выбору центра тяжести в зависимости от задачи. Это может быть:

- центр масс,
- объект, сумма расстояний до которого минимальна, среди всех внутри сферы,
- объект, который внутри сферы радиуса R содержит максимальное количество других объектов из всей выборки,
- объект, который внутри сферы меньшего радиуса содержит максимальное количество объектов.

Преимущества метода – наглядность визуалицаии кластеризации, возможность регулирования кластеризации через R.

Недостатки метода – зависимость работы алгоритма от выбора исходных центров кластеров, необходимость знаний о ширине кластеров. Для поиска аномалий необходимо также проводить исследование полученных кластеров.

Local outlier factor

Local outlier factor [11] (фактор локального отклонения) – метод, позволяющий оценить степень принадлежности рассматриваемого объекта к группе ближайших по отношению к нему объектов. Данный метод не является методом кластеризации, но был получен синтезом алгоритмов DBSCAN и OPTICS[12] и направлен на выявление значений, отклоняющихся от ближайшей совокупности объектов.

Основная идея алгоритма состоит в сравнении локальной плотности объекта с локальными плотностями его объектов (рисунок 8).

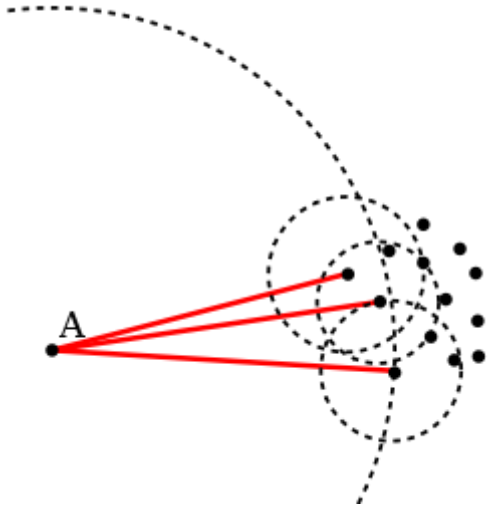


Рисунок 8 - точка A имеет меньшую плотность, чем её соседи.

Плотность объекта рассчитывается на основе k ближайших соседей и соседей, которые попадают в область, ограниченную k соседями (здесь нет связи с методом классификации под названием «алгоритм k ближайших соседей»). Сравнивая плотности объектов с плотностями их соседей можно выделить регионы с одинаковой плотностью и объекты, чья плотность меньше плотности их соседей. Такие объекты будут являться аномалиями.

Для рассчитываемого объекта вычисляется k-distance – k-расстояние, в пределах которого лежат его k-соседей и соседей, которые попадают в эту область (таким образом, может приниматься в расчет больше, чем k соседей). Данное расстояние используется для расчета расстояния достижимости (reachability distance) от одной точки до другой:

$$\text{reachability-distance}_k(A, B) = \max\{\text{k-distance}(B), d(A, B)\} \quad (10)$$

где d – расстояние между объектами.

Расстояние достижимости от объекта A до объекта B – это расстояние между этими объектами, которое при этом не меньше, чем k-расстояние объекта B. Таким образом, расстояние достижимости между k-соседями объекта и объектом будут одинаковы. Данное расстояние несимметрично и используется для получения более стабильных результатов. Например, на рисунке 9 показано расстояние достижимости от точек B, C, D до точки A при использовании 3-ближайших соседей. Таким образом, точка D не входит в множество 3-ближайших соседей точки A.

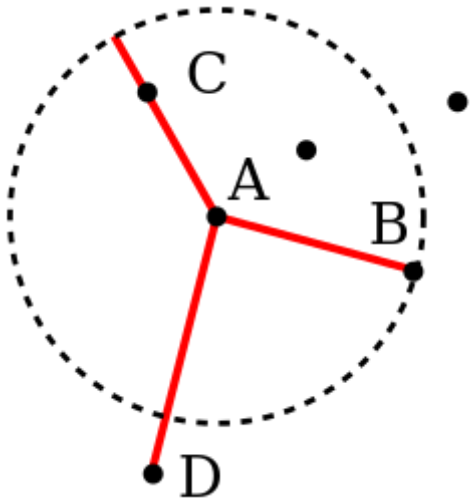


Рисунок 9 - расстояние достижимости от точек B, C, D до точки A.

Основываясь на расстоянии достижимости, рассчитывается локальная плотность объекта.

$$\operatorname{lrd}(A) := 1/\left(\frac{\sum_{B \in N_k(A)} \text{reachability-distance}_k(A,B)}{|N_k(A)|}\right) \quad (11)$$

где $N_k(A)$ – множество соседей объекта A.

Локальная плотность обратно пропорциональна среднему значению расстояния достижимости от k-ближайших соседей точки A до точки A.

Основываясь на локальной плотности объекта A и k-соседей объекта A рассчитывается искомое значение, LOF - фактор локального отклонения.

$$\operatorname{LOF}_k(A) := \frac{\sum_{B \in N_k(A)} \frac{\operatorname{lrd}(B)}{\operatorname{lrd}(A)}}{|N_k(A)|} = \frac{\sum_{B \in N_k(A)} \operatorname{lrd}(B)}{|N_k(A)|} / \operatorname{lrd}(A) \quad (12)$$

Фактор локального отклонения является отношением средней локальной плотности соседей объекта к локальной плотности самого объекта. Таким образом, если средняя локальная плотность соседей выше, чем локальная плотность самого объекта – то данный объект отклоняется от своих соседей.

Если LOF = 1, то объект сравним с ближайшими соседями. Соответственно, чем больше значение LOF, тем менее вероятно, что объект принадлежит группе своих ближайших соседей.

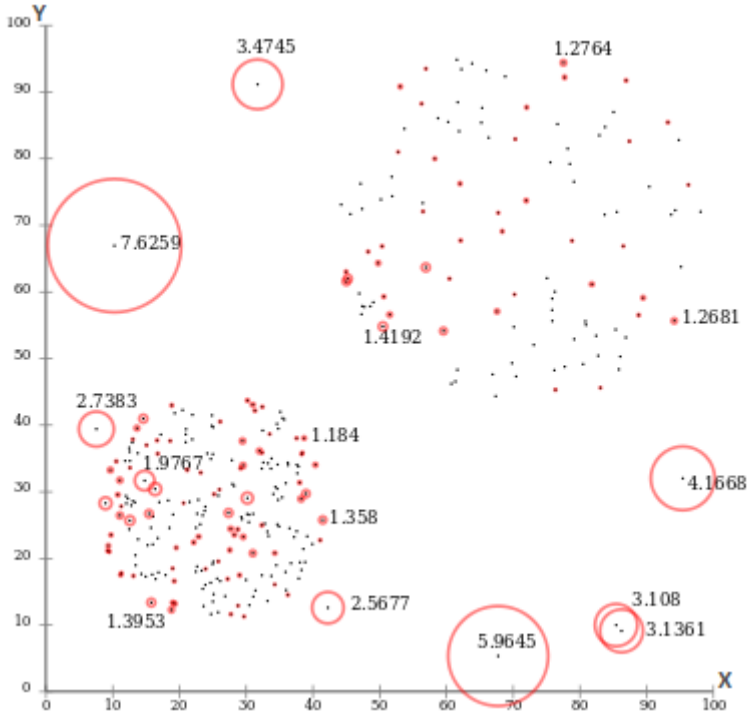


Рисунок 10 - пример расчета LOF.

Преимущества метода – получение значения, доступного для интерпретации и анализа. Основывается на плотности значения ближайших соседей.

Недостатки метода – сложность интерпретации относительно допустимого предела, превышение которого будет строго соответствовать аномальному значению. Низкая производительность. Относительно поставленной задачи, в случае достаточно большого количества аномалий в выборке, рассчитанные значения LOF будут являться нормальными; возможность ложного реагирования на данные, которые лежат в определенном интервале между двумя совокупностями нормальных данных.

1.4. Методы сглаживания данных

Процесс поступления данных, как правило, является хаотичным. Во время снятия наблюдения с различных измерительных приборов неминуемы шумы или разбросы данных в пределах определённого интервала. Для преодоления подобных проблем используют методы фильтрации и сглаживания данных.

Значение посещаемости сайта является однозначным и не имеет шумов. Однако, в зависимости от длины интервала, на котором рассматривается значение зафиксированной посещаемости в период этого интервала, соседские значения посещаемости могут резко различаться. При таком различии данные достаточно сложно поддаются анализу. Поэтому, для улучшения качества анализа, необходима предобработка данных. Далее будут рассмотрены методы сглаживания данных.

Метод простого скользящего среднего

Метод скользящего среднего активно применяется в статистике и экономике для сглаживания временных рядов, в технике при обработке сигналов.

При использовании метода простого скользящей средней выходные значения вычисляются на основе среднего арифметического значений исходного ряда за установленный период:

$$S_t = \frac{1}{n}\sum_{i=0}^{n-1} p_{t-i} = \frac{p_t + p_{t-1} + \ldots + p_{t-n+1}}{n} \qquad (13)$$

где S_t - значение скользящего среднего в момент времени t, p_{t-i} - значение функции в момент времени t-i, n – количество значений ряда, используемых для расчёта скользящего среднего.

На основе метода простой скользящей средней существует метод со сдвигом окна. Данный метод рассматривает не только предыдущие значения функции, но и последующие:

$$S_t = \frac{1}{n}\sum_{i=-k}^{n-k-1} p_{t-i} = \frac{p_{t-k} + p_{t-k+1} + \ldots + p_{t-n+k+1}}{n} \qquad (14)$$

где S_t - значение скользящего среднего со сдвинутым на k окном в момент времени t, k – сдвиг окна.

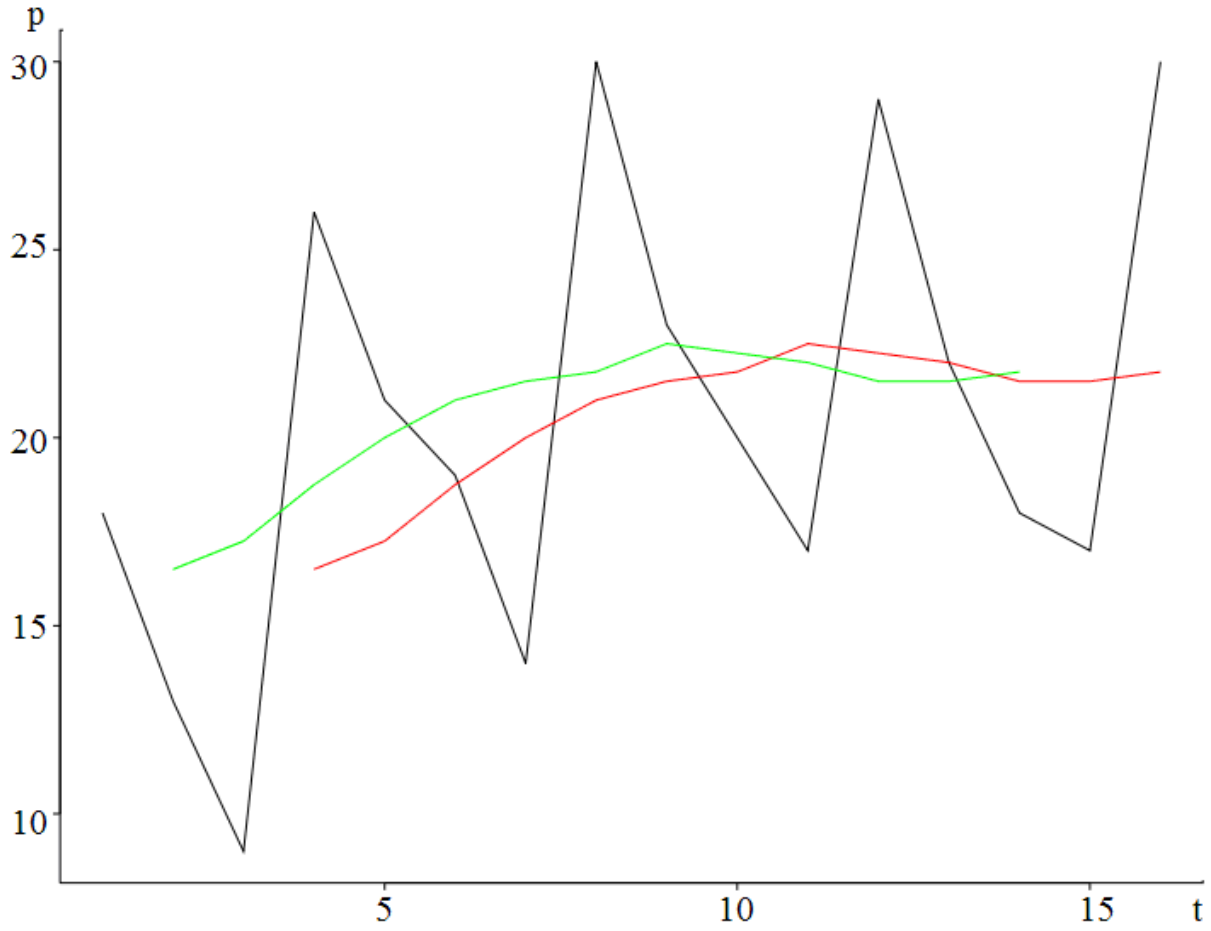


Рисунок 11 - пример простой скользящей средней по 4ём значениям функции. Красный график – скользящая средняя без сдвига окна, зелёный – со сдвигом окна на k = 2.

Преимущества метода – метод прост для понимания и прост в реализации.

Недостатки метода – в случае со сдвинутым окном первые и последние уровни ряда теряются (не сглаживаются). В случае без сдвига окна теряются только первые уровни. Не учитываются веса данных относительно рассчитываемого момента времени.

Взвешенное скользящее среднее

Взвешенное скользящее среднее - скользящее среднее, при вычислении которого вес каждого члена исходного ряда, начиная с меньшего, равен соответствующему члену арифметической прогрессии. То есть, при вычислении взвешенного скользящего среднего для временного ряда, последние значения исходного ряда считаются более значимыми чем предыдущие, причём функция значимости линейно убывающая:

$$W_t = \frac{np_t + (n-1)p_{t-1} + ... + (n-i)p_{t-i} + ... + 2p_{t-n+2} + p_{t-n+1}}{n + (n-1) + ... + 2 + 1} = \frac{2}{n(n-1)} \sum_{i=0}^{n-1} (n-i)p_t \quad (15)$$

где W_t - значение взвешенного скользящего среднего в момент времени t, p_{t-i} - значение функции в момент времени t-i, n – количество значений ряда, используемых для расчёта взвешенного скользящего среднего.

Принцип взвешенного скользящего среднего используется также в экспоненциально взвешенном скользящем среднем.

Преимущества метода – метод использует веса данных относительно рассчитываемого момента времени.

Экспоненциально взвешенное скользящее среднее

Экспоненциально взвешенное скользящее среднее основано на общей идее экспоненциального сглаживания. Экспоненциальное сглаживание активно используется при прогнозировании временных рядов:

$$s_t = \begin{cases} c_1 & : t = 1 \\ s_{t-1} + \alpha \cdot (c_t - s_{t-1}) & : t > 1 \end{cases} \quad (16)$$

где s_t - сглаженный ряд, c_t - исходный ряд, α - коэффициент сглаживания ($0 < \alpha < 1$). Коэффициент сглаживания характеризует скорость уменьшения весов, чем меньше его значение, тем больше влияние предыдущих значений на текущую величину среднего.

Первое значение экспоненциального скользящего среднего принимается равным первому значению исходной функции.

Существует также модифицированное скользящее среднее, которое является частным случаем экспоненциального взвешенного скользящего среднего:

$$M_t = \frac{p_t + (n-1)M_{t-1}}{n} \quad (17)$$

где M_t - значение модифицированного скользящего среднего, при этом коэффициент сглаживания:

$$\alpha = \frac{1}{n} \quad (18)$$

Достоинство метода - метод использует веса данных относительно рассчитываемого момента времени. Степень значимости весов убывает экспоненциально.

1.5. Методы получения данных

Для создания и тестирования приложения по обнаружению и анализу аномалий в статистических данных посещаемости сайта необходимо получить реалистичную

модель данных. Для достижения реалистичной модели можно использовать следующие методы:

- искусственная генерация данных (это могут быть случайные данные, заданные в пределах определённого диапазона, или данные, подчиняющиеся определённому закону распределения),
- использование данных счётчиков с открытой статистикой, используемых в интернет-сайтах.

Существуют сервисы, предоставляющие зарегистрированным пользователям возможность просматривать подробную статистику о посещаемости их электронных ресурсов.

Данные о посещаемости предаются с помощью кода «счётчика» используемого сервера, который имплантируется в код сайта. «Счётчик» содержит исполняемый код (обычно, это скачивание однопиксельной картинки). Таким образом, при загрузке страницы происходит запрос к сервису. Сервис обрабатывает запрос, при этом получает все необходимые данные о пользователе. На основании групп запросов сервис формирует используемую статистику.

К сервисам, которые предоставляют такие возможности, относятся: «Яндекс.Метрика» (https://metrika.yandex.ru), «Рейтинг@mail.ru» (http://top.mail.ru), «Rambler Топ100» (http://top100.rambler.ru/), «LiveInternet» (http://www.liveinternet.ru/) и другие. Существуют сервисы, которые предоставляют статистику электронного ресурса любому посетителю с разрешения владельца этого ресурса (например «Рейтинг@mail.ru»).

Преимуществом использования данных реальных счётчиков перед искусственной генерацией данных является реальная модель данных. Для использования данного метода необходимо выбрать репрезентативные интернет-сайты для проведения соответствующих исследований.

Выводы

На основании проведённого анализа можно выделить следующие функциональные требования к приложению:

1) Для формирования данных для анализа необходимо использовать реальные данные, взятые с сервиса, предоставляющего открытую статистику для сайтов.
2) Необходимо обеспечить сохранность и агрегацию данных с целью их оперативного и гибкого использования.

3) Исходя из сложности математического представления предмета исследования, необходимо использовать по меньшей мере два метода поиска аномалий, предоставляющих возможность для анализа. Наиболее оптимальными методами с точки зрения простоты и результативности для решения поставленной задачи являются метод среднеквадратичного отклонения и метод фактора локального отклонения. Метод среднеквадратичного отклонения позволяет варьировать необходимый интервал, основанный на всей совокупности однотипных данных, основан на общей выборке и позволяет выявить нехарактерные данные (резкого роста или резкого падения данных). Целевое назначение метода фактора локального отклонения направлено непосредственно на анализ отклонений данных, поэтому полученные результаты будут отражать степень отклонения от ближайшего локального скопления данных, что будет в наибольшей степени доступно для интерпретации.
4) Необходимо представить наглядное отображение данных и результатов анализа.
5) Выбранные методы поиска аномалий будут являться удовлетворительными для решения поставленной задачи, если они выявят нетипичные данные в текущем периоде по сравнению с данными за предыдущие периоды.

2. Конструкторский раздел

Для разработки качественного приложения необходимо составить логически осмысленную и наглядную структуру. Данная структура должна описывать:

- формат используемых данных,
- способ хранения данных,
- метод получения и изменения данных,
- алгоритмы выявления и анализа аномалий в данных,
- использование данных в этих алгоритмах.

2.1. Представление данных

Для наращивания осмысленной структуры, в данной работе используются данные, взятые с сервиса статистики «Рейтинг@mail.ru», так как этот сервис предоставляет доступ к данным о посещаемости электронного ресурса, при условии, что владелец данного ресурса открыл доступ к этой статистике. Также сервис обладает удобным API, с помощью которого можно получить в том числе данные о посещаемости сайта.

Для получения данных о посещаемости необходимо совершить http запрос по адресу «http://top.mail.ru/json/mdynamics?id=ID&date=DATE», где подстрока до знака «?» означает адрес страницы, по которой можно получить данные о посещаемости с top.mail.ru в формате JSON, а подстрока после знака «?» означает параметры запроса. К этим параметрам относятся:

- ID – идентификатор электронного ресурса, зарегистрированного в данном сервисе,
- DATE – дата дня, за который необходимо получить статистику.

Структура ответа сервиса представлена в таблице 2.

Таблица 2 - структура данных ответа сервиса.

Имя	Значение
elements	Последовательность данных с указанием промежутка времени и значения посещаемости.
what	Вид данных выборки.
error	Наличие ошибок.
date	Дата дня, за который необходимо получить статистику в формате «YYYY-MM-DD».
date_x	Дата дня, за который необходимо получить

	статистику в формате «число месяц».
date_xs	Дата дня, за который необходимо получить статистику в формате «число месяц сокр.».
date_prev	Дата предыдущего дня в формате «YYYY-MM-DD».
date_prev_x	Дата предыдущего дня в формате «число месяц».
date_next	Дата следующего дня в формате «YYYY-MM-DD».
date_next_x	Дата следующего дня в формате «число месяц».
Is_today	Является ли текущая дата сегодняшним днём.
Weekday	День недели.
category_name	Категория электронного ресурса.
category_nick	Тематика электронного ресурса.
title	Название электронного ресурса.
url	Адрес электронного реусурса.

Пример ответа сервиса:

{

“elements”:

[{“hour”:”00”,”minute”:”00”,”value”:78.4},…,{“hour”:”23”,”minute”:”55”,”value”:83.6}],

“what”: “hits”,

“error”: 0,

“date”: “2014-06-11”,

“date_x”: “13 июня”,

“date_xs”: “13 июн”,

“date_prev”: “2014-06-12”,

“date_prev_x”: “12 июня”,

“date_next”: “2014-06-14”,

“date_next_x”: “14 июня”,

“is_today”: “0”,

“weekday”: 3,

“category_name”: “Интернет \u003e Интернет-услуги”,

“category_nick”: “Internet-Service”,

“title”: “Рейтинг@Mail.Ru”,

“url”: “http://top.mail.ru”

}

Исходя из структуры ответа сервиса, необходимо хранение данных в следующем виде:

Таблица 3 - необходимая структура хранения данных.

Имя	Значение
id	id счётчика в сервисе top.mail.ru.
title	Название электронного ресурса.
date	Дата полученных данных.
time	Время полученных данных.
value	Значение посещаемости ресурса в соответствующую дату и время.

Так как данные о посещаемости, передаваемые с ответом от сервиса, являются данными, зафиксированными за 5тиминутный отрезок времени, то для работы с такими данными нет необходимости в применении алгоритма сглаживания.

2.2. Структура базы данных

С учётом выявленной структуры хранения данных, для агрегации этих данных, удобства доступа и гибкости использования была разработана структура базы данных, состоящая из двух таблиц:

Таблица 4 - структура таблицы имён.

Таблица имён	
Целое	ID счётчика.
Строка	Название электронного ресурса.

Таблица 5 - структура таблицы данных.

Таблица данных	
Целое	ID счётчика.
Дата	Дата полученных данных.
Время	Время полученных данных.
Вещественное	Значение посещаемости.

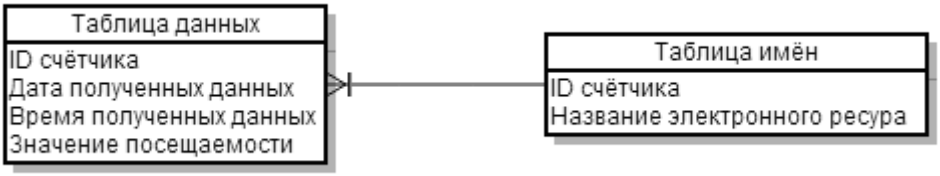


Рисунок 12 - ER-диаграмма базы данных.

Данная структура проста, наглядна и может быть легко реализована на большинстве существующих СУБД.

2.3. Описание алгоритмов выявления аномалий

В соответствии с поставленной задачей и выводами из аналитического раздела было принято решение использовать алгоритм LOF и алгоритм, основанный на среднеквадратичном отклонении (3G). Данные алгоритмы работают непосредственно со значениями, поэтому на вход алгоритмам подаётся массив объектов представленных в формате {ключ, значение}, где ключом является дата и время, а значением – значение посещаемости в этот период.

Так как аномалии выявляются на выборке за определённый период относительно выборок за прошлые периоды, то нет необходимости искать аномалии на всей выборке данных.

2.3.1. Алгоритм LOF

Схема алгоритма LOF базируется на выкладках, приведённых в 1.3.2. Данный алгоритм реализуется в линейном одномерном пространстве. Для реализации алгоритма для каждого объекта необходимо держать список соседей и k-distance(), lrd(). Схема алгоритма приведена на рисунке 13. Для повышения эффективности за k-ближайших соседей будем принимать k соседей, расположенных на различной дистанции и всех соседей, входящих в k-расстояние (k-distance).

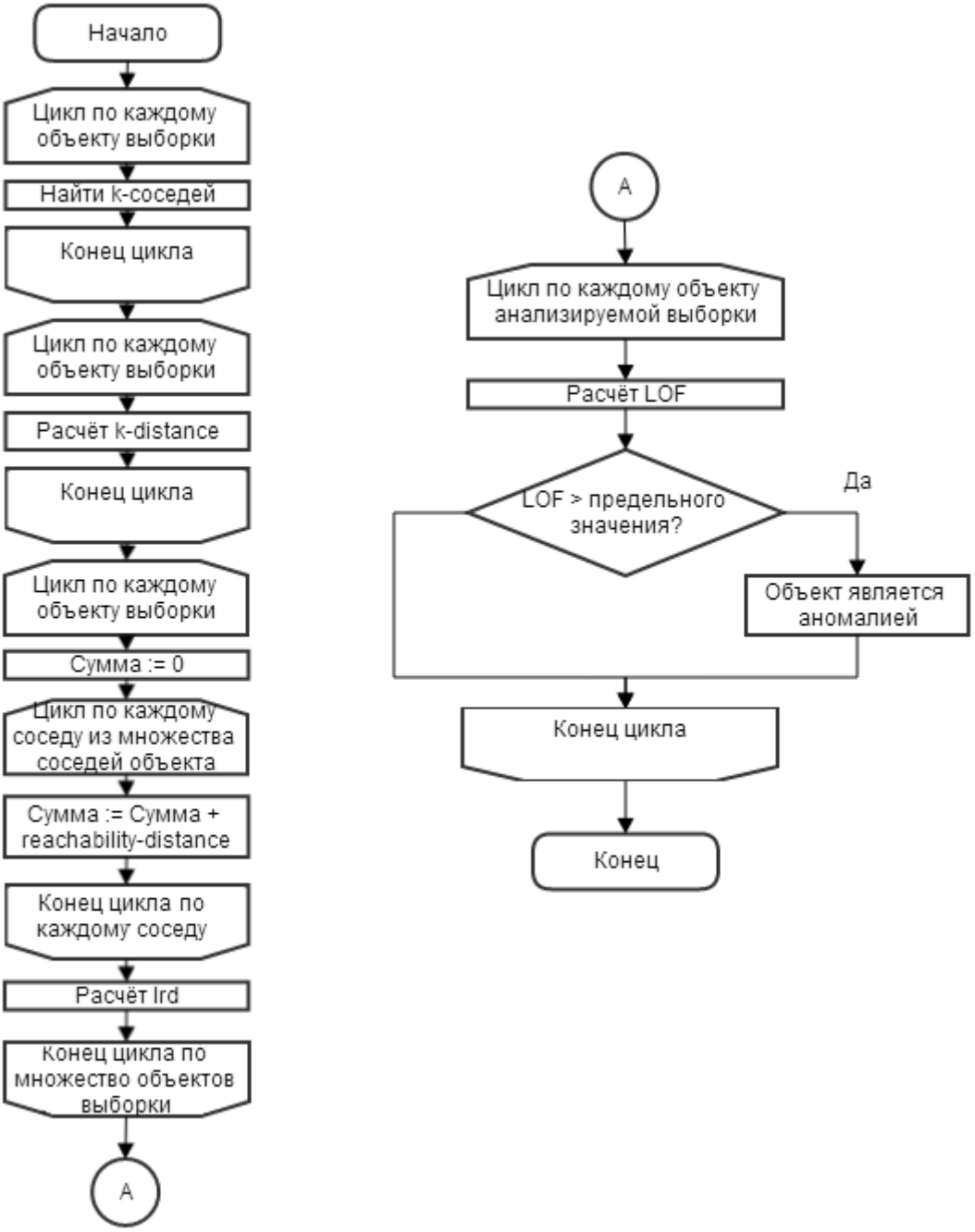


Рисунок 13 - Схема алгоритма LOF.

В результате поиска k-соседей для каждого объекта выборки формируется множество соседей $N_k(A)$. Для расчёта reachability-distance используется формула 10. На основе суммы reachability-distance по каждому из соседей происходит расчёт lrd по формуле 11. Аналогичным образом, основываясь на lrd соседей объекта анализируемой выборки происходит расчёт LOF по формуле 12.

В данном алгоритме предусмотрена возможность регулирования порогового значения LOF и количество k рассматриваемых соседей. Выбор подходящего значения является интуитивным и определяется экспертом в области, где применяется этот алгоритм.

Верхняя оценка сложности алгоритма O(n log n) [11].

2.3.2. Алгоритм 3G

Схема алгоритма 3G базируется на выкладках, приведённых в 1.3.1. Схема алгоритма приведена на рисунке 14. Для оправдания использования данного метода используется приближение, описанное в 4.1, о предположении характера используемых данных, как данных подчиняющихся нормальному закону распределения.

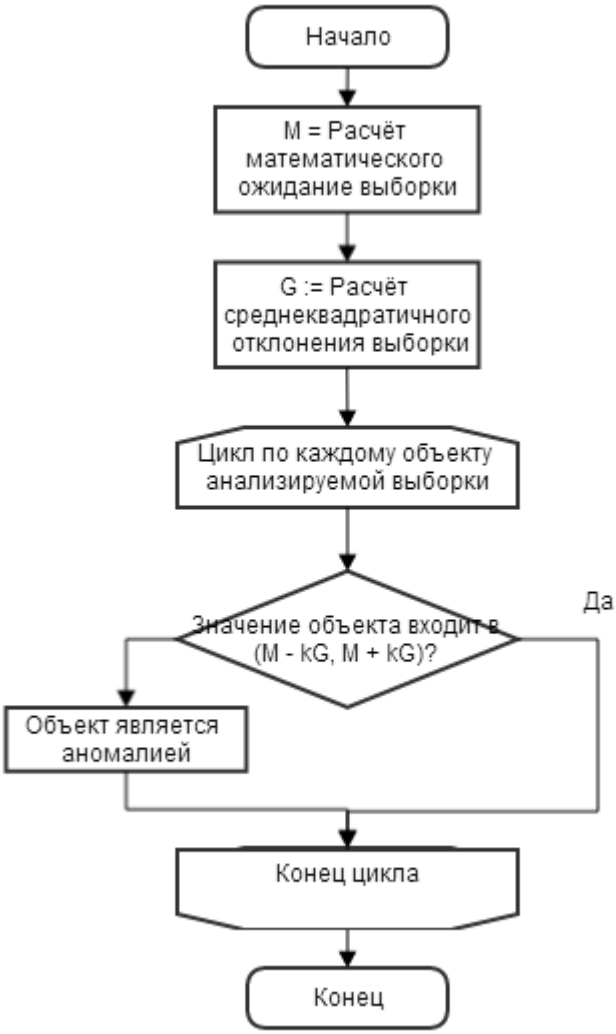


Рисунок 14 - схема алгоритма 3G.

Расчёт среднеквадратичного отклонения и математического ожидания производятся соответственно по формулам 2 и 3.

В данном алгоритме есть возможность регулирования значения интервала, основанного на среднеквадратичном отклонении, через коэффициент k. Так, в интервал (M – G, M + G), где M – математическое ожидание выборки, попадает 68% значений, в интервал (M – 2G, M + 2G) попадают 95% значений, в интервал (M – 3G, M + 3G) – 99%.

Верхняя оценка сложности алгоритма O(n).

2.4. Описание применения алгоритмов выявления аномалий

С учётом статистической значимости значений скорости изменения посещаемости, описанных в 4.1, целесообразно использовать эти значения наравне с данными о посещаемости.

Таким образом, в приложении необходимо предусмотреть возможность использования двух типов данных и двух алгоритмов. Для этого требуется наглядный и непротиворечивый алгоритм последовательности выбора и обработки данных и отображения результатов (рисунки 15-17).

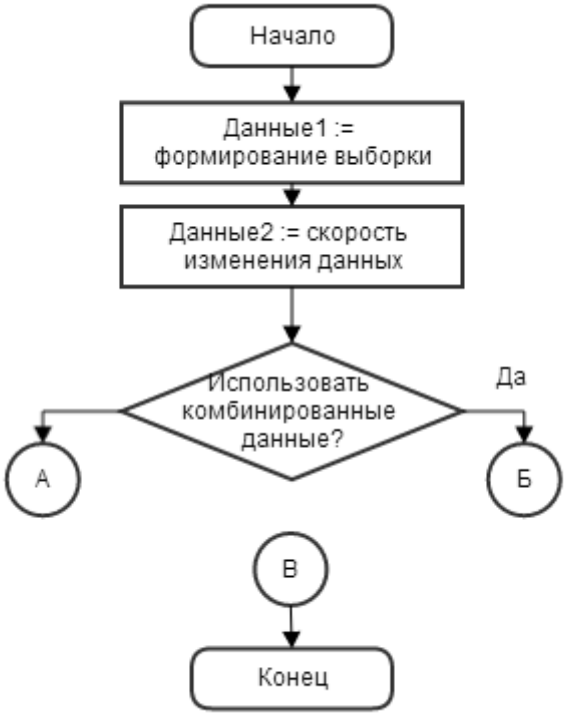


Рисунок 15 – схема алгоритма процесса формирования и обработки данных.

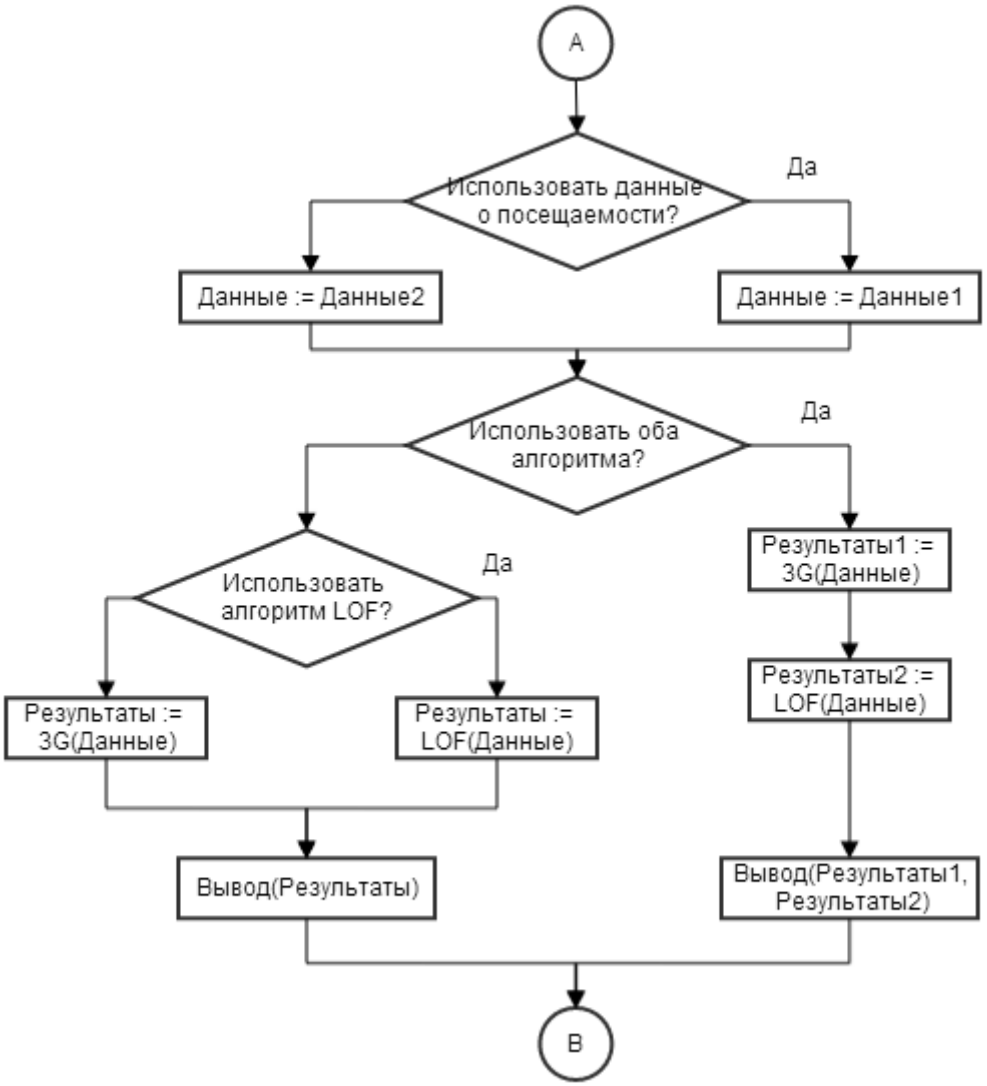


Рисунок 16 - схема алгоритма процесса формирования и обработки данных. Обработка данных в случае использования одного типа данных.

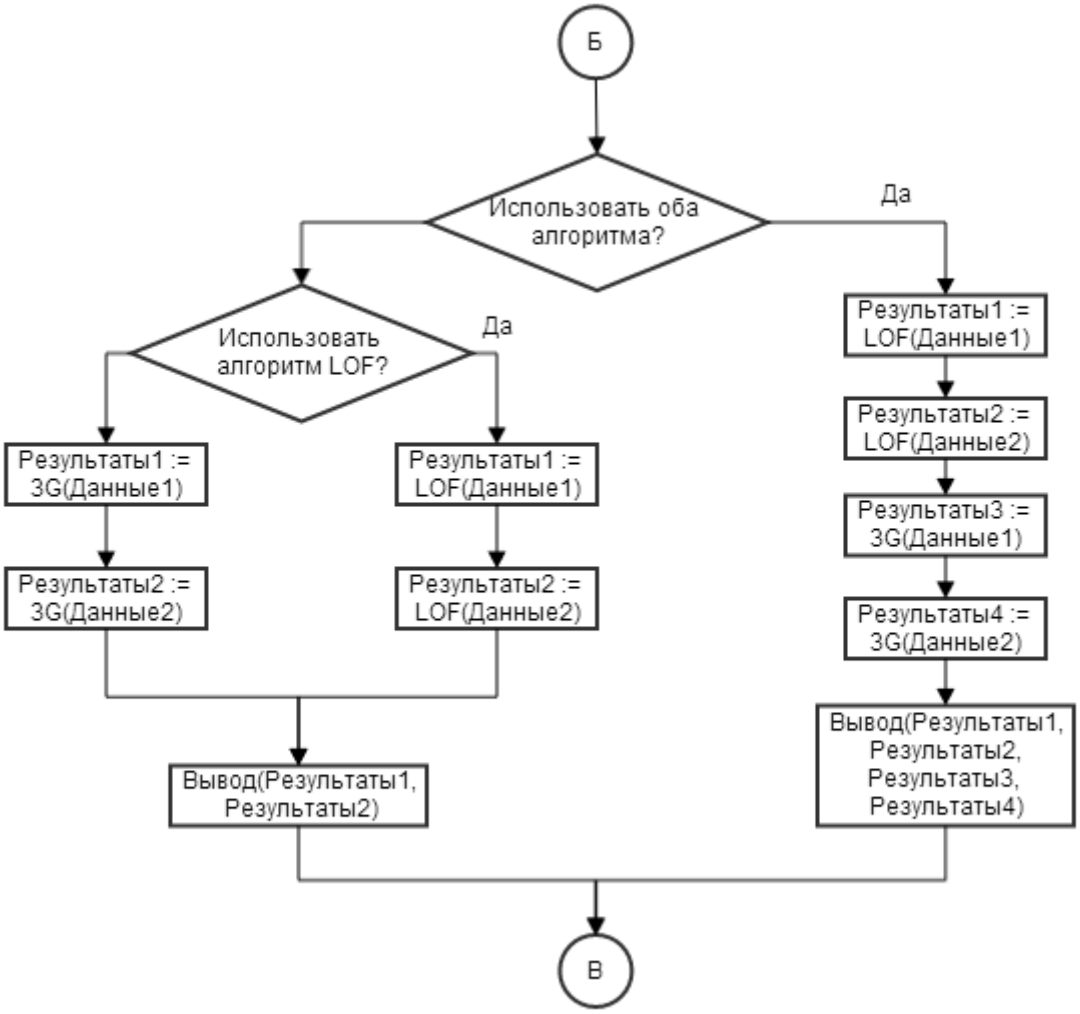


Рисунок 17 - схема алгоритма процесса формирования и обработки данных. Обработка данных в случае использования двух типов данных.

Проектируемое приложение предполагает следующий сценарий работы:

- пользователь создаёт выборку данных за определённый период,
- определяет длину выборки для анализа из последних значений созданной выборки,
- выбирает необходимый алгоритм для проведения анализа данных,
- получает интерпретируемые результаты работы алгоритма.

Формализация функциональных задач, решение которых необходимо реализовать в проектируемом приложении, представлена в диаграмме функционального моделирования:

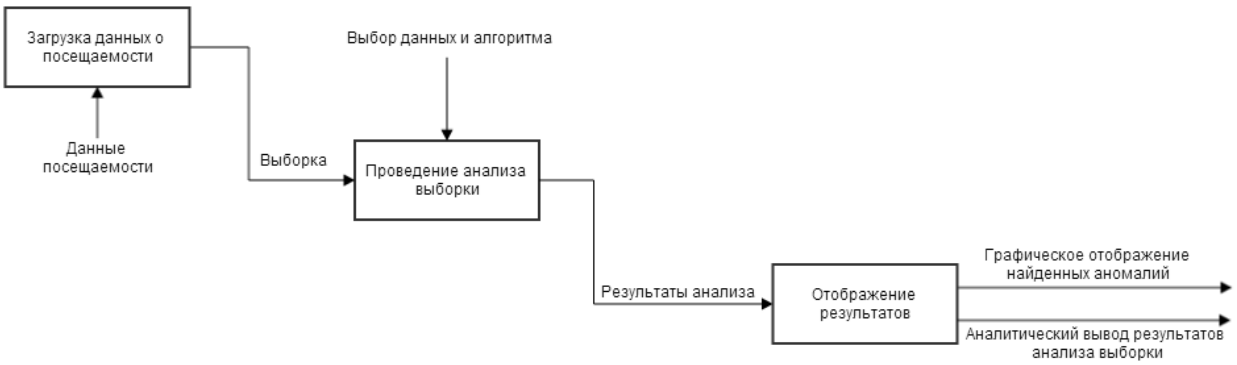


Рисунок 18 - IDEF диаграмма проектируемого приложения.

Имея группу данных о посещаемости и группу данных скорости изменения показателя посещаемости за период анализа и за предыдущие периоды, а также алгоритмы выявления и анализа аномалий в этих данных, можно составить диаграмму потока данных в проектируемом программном продукте:

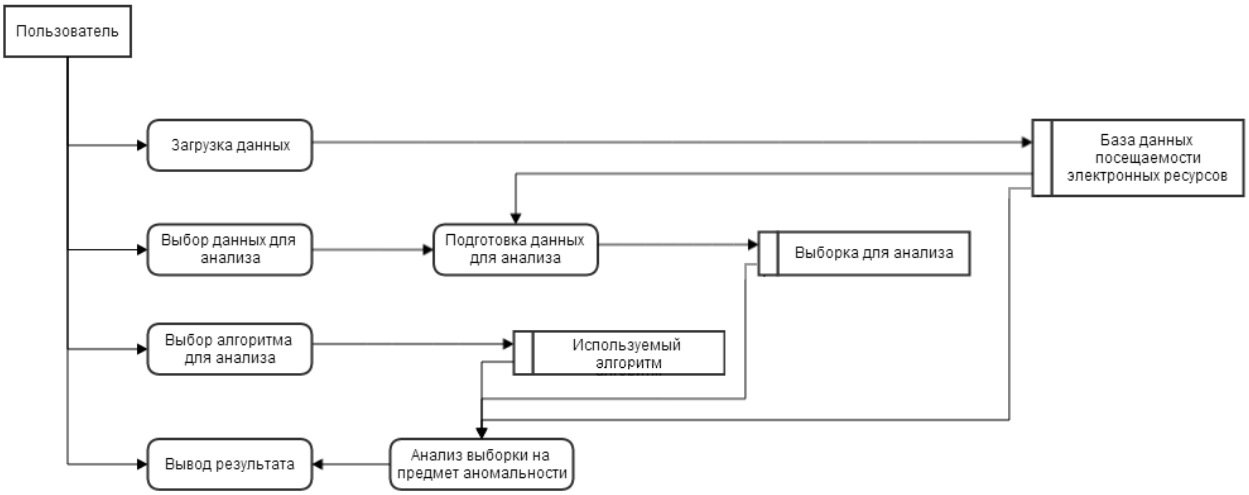


Рисунок 19 - DFD диаграмма проектируемого приложения.

С учётом функционального проектирования и структурного анализа данных, проектируемое приложение будет отражать структуру, представленную на рисунке 20.

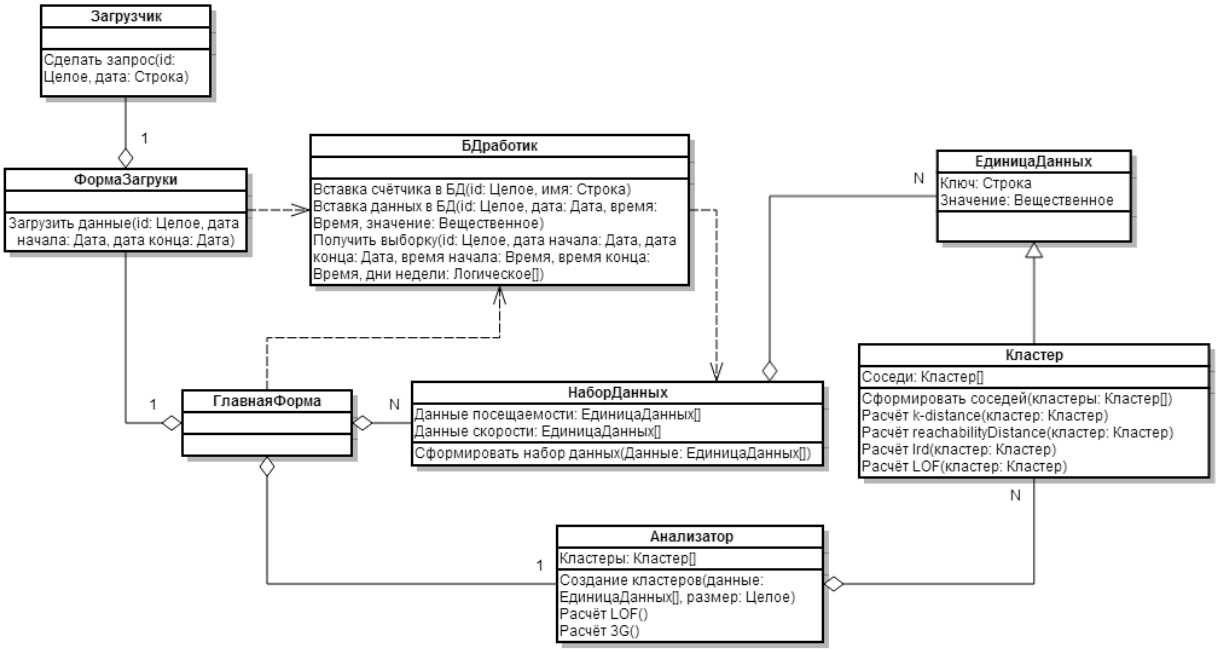


Рисунок 20 – диаграмма классов проектируемого приложения.

3. Технологический раздел

3.1. Выбор средств разработки

3.1.1. Выбор ЯП

Для разработки данного приложения было принято использовать язык программирования C# с использованием IDE visual studio 2010, так как с использованием этого языка достигаются следующие преимущества:

- скорость разработки,
- использование объектно-ориентированной парадигмы,
- эффективная работа с использованием Visual Form Application, простота проектирования интерфейса,
- широкая поддержка языка, обширная документация, большое количество модулей сторонних разработчиков,
- удобство разработки с использованием SQL server.

3.1.2. Выбор СУБД

Для реализации базы данных, описанной в 2.2, было принято использовать Microsoft SQL Server, так как использование этой СУБД достигаются следующие преимущества:

- скорость разработки,
- возможность централизованного доступа к базе данных с различных приложений при установленном Microsoft SQL Server,
- простота использования при разработке на языке C#.

3.2. Организация и доступ к БД

С использованием Microsoft SQL Server была спроектирована база данных, описанная в 2.2. Созданная реализация полностью соответствует проектируемой.

3.2.1. Структура БД

Созданная база данных представлена на диаграмме:

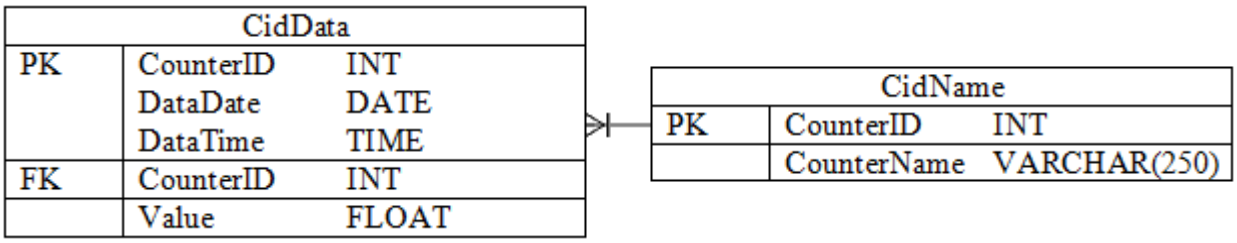


Рисунок 21 - ER-модель созданной базы данных.

3.2.2. Сценарии создания

Для реализации базы данных был создан скрипт с использованием языка T-SQL для создания базы, таблицы имён и таблицы данных:

```
CREATE DATABASE Attendance;  -- Создание базы данных с данными о посещаемости

USE Attendance;

CREATE TABLE CidName
(
	CounterID INT NOT NULL,                    -- ID счётчика
	CounterName VARCHAR (250) NOT NULL,  -- Название сайта
);

CREATE TABLE CidData  -- Создать таблицу "Сотрудники"
(
	CounterID INT NOT NULL,    -- ID счётчика
	DataDate DATE NOT NULL,  -- Дата
	DataTime TIME NOT NULL,  -- Время
	Value FLOAT NOT NULL       -- Значение посещаемости за 5 минут
);

-- Добавление ключей
ALTER TABLE CidName
ADD
CONSTRAINT PK_CounterID PRIMARY KEY (CounterID);

ALTER TABLE CidData
ADD
```

CONSTRAINT PK_DataID PRIMARY KEY (CounterID, DataDate, DataTime),
CONSTRAINT FK_CounterID FOREIGN KEY (CounterID) REFERENCES CidName (CounterID);

3.2.3. Доступ к данным через приложение

Для организации доступа к данным в приложении был написан класс SQLworker. Данный класс изолирует работу с Microsoft SQL Server от остальной части приложения. Данный класс обеспечивает следующую функциональность:

Таблица 6 - основные функции SQLworker.

Имя метода	Принимаемые параметры	Действие
insertCounter	Id счётчика, имя счётчика	Вставляет или производит обновление номера счётчика электронного ресурса и его названия в таблице CidName.
insertData	Id счётчика, дата полученных данных, время полученных данных, значение посещаемости	Вставляет или производит обновление номера счётчика, дату, время и значение посещаемости в этот период в таблице CidData. Если счётчик отсутствует – происходит вызов insertCounter
getData	Id счётчика, дата начала выборки, дата конца выборки, начало интервала времени выборки, конец интервала времени выборки, дни недели выборки	Возвращает данные из таблицы CidData в выбранные дни, в выбранный период времени, по выбранным дням недели.

Данная функциональность позволяет создавать наиболее удобные выборки для проведения анализа данных на предмет аномальности значений.

3.3. Получение данных

Так как в работе используется сервис статистики top.mail.ru, то требуется сделать запрос к этому сервису, получить и обработать ответ.

Осуществление запроса и получение ответа осуществляется через класс WebRequest, расположенный в модуле System.Net. Оболочка над классом расположена в классе Downloader.

Ответ на запрос приходит в формате JSON. Для работы с данным форматом используется сторонний модуль Newtonsoft.Json. Данный модуль позволяет создавать динамические объекты на основе текстового представления данных в формате JSON. Формат ответа описан в 2.1.

На основе ответа производится вставка данных в базу данных с использованием метода insertData класса SQLworker.

Для обеспечения стабильной работы серии запросов между запросами происходит задержка в 50 мс.

3.4. Руководство пользователя

3.4.1. Системные требования

Так как приложение разработано с помощью C# и Microsoft SQL Server, для успешного запуска необходимо обеспечить следующие программно-аппаратные требования:

- процессор с тактовой частотой не ниже 2 ГГц,
- ОЗУ объёмом не меньше 512 Мб,
- не меньше 64 Мб свободного места на жёстком диске,
- ОС: Windows XP, Windows Vista, Windows 7,
- наличие Microsoft SQL Server 2008 r2,
- наличие .NET framework версии 4.0 и выше.

3.4.2. Настройка подключения к СУБД

Для создания базы данных создан специальный скрипт «create.sql» на языке T-SQL, располагающийся в папке «db script». Данный скрипт создаёт базу данных «Attendance», две таблицы «CidName» и «CidData», и связи между ними.

В папке с приложением находится файл «connectionString.txt» – строка подключения к базе данных. По умолчанию содержит следующую строку: «server = .\sqlexpress; integrated security = true; database = Attendance».

3.4.3. Внешний вид

Интерфейс программы выглядит компактным и немного нетривиальным. Поэтому необходимо разъяснение основным функциональным особенностям данного приложения.

Программа реализована с помощью двух форм. Основной функционал программы сосредоточен в форме «поиск и анализ аномалий посещаемости сайта» (рисунок 22).

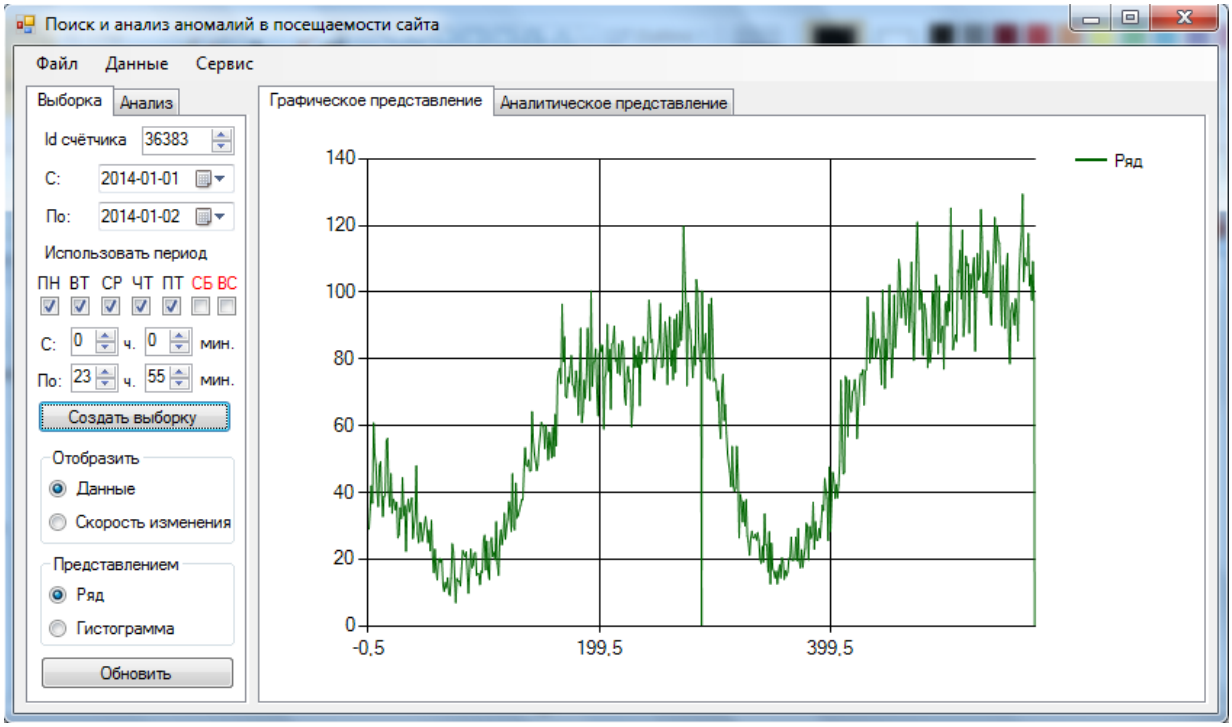


Рисунок 22 – главная форма приложения с выбранными данными за два дня, представленные в виде графика.

Данная форма состоит из окна настроек для выборки данных, выбора данных (данные, скорость изменения данных), выбора представления выбранных данных (ряд, гистограмма), представление данных в графической и аналитической форме.

Второе окно можно вызвать с помощью последовательности команд «данные»-«получить данные»:

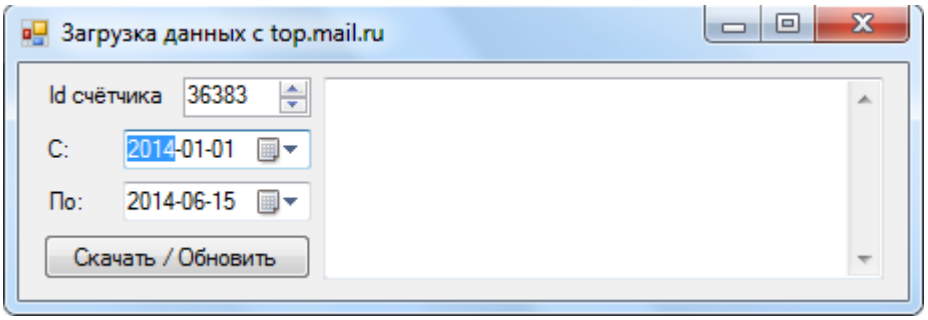


Рисунок 23 – форма загрузки данных с top.mail.ru.

В данной форме есть возможность выбора периода получения данных и номера счётчика соответствующего электронного ресурса для загрузки данных в базу данных.

3.4.4. Элементы управления

Создание выборки

Окно создания выборки представляет собой инструмент для формирования определённой выборки данных с учётом времени выборки и дня недели. С помощью этого инструмента можно формировать демонстрационные выборки как за весь период дня в течение нескольких дней, так и за конкретный час в определённый день недели на протяжении нескольких месяцев.

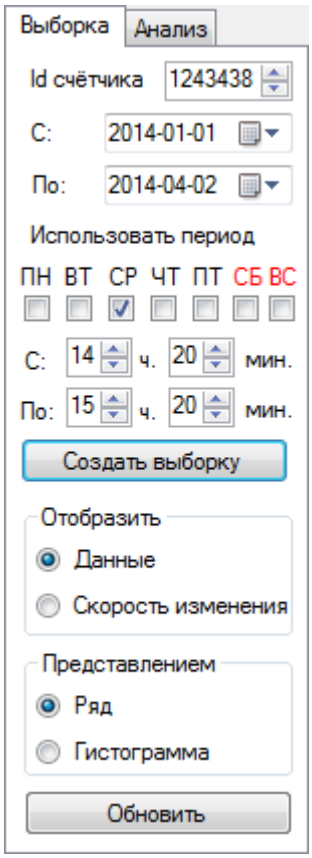
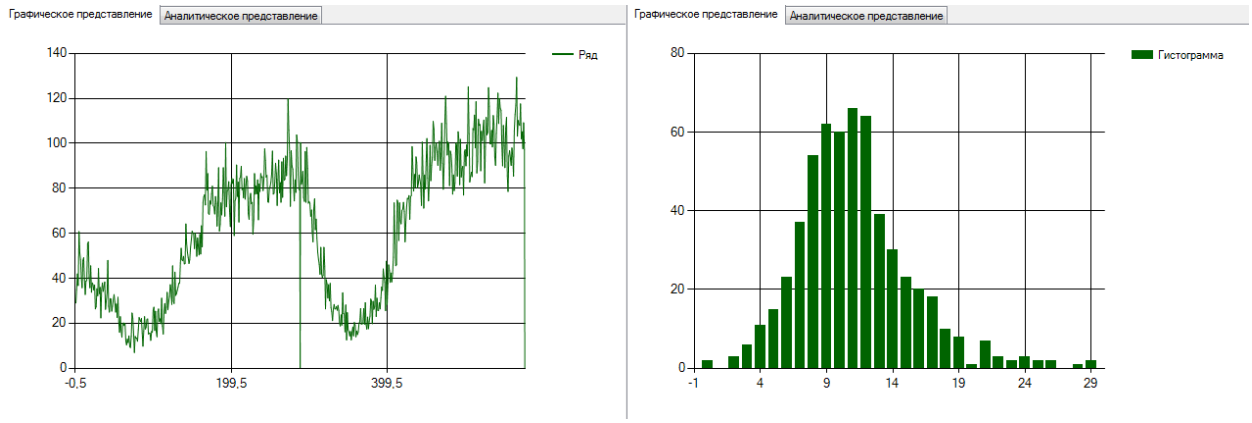


Рисунок 24 – окно создания выборки.

При создании выборки приложение даёт возможность манипулировать как данными посещаемости, так и скоростью изменения этих данных. Также данным окном можно выбирать способ отображения данных (в представлении последовательности данных или в виде гистограммы).

Графическое представление

Окно графического представления данных предоставляет два метода описательной статистики для данных, выбранных пользователем. Первый метод – представление ряда данных в виде графика. При этом граничные интервалы разделены провалами (рисунок 25 – а), по оси абсцисс – ключи, которые находятся в аналитическом представлении, которым можно сопоставить время. Второй метод – представление данных в виде гистограммы (рисунок 25 – б), по оси абсцисс – ключи, которые находятся в аналитическом представлении, которым можно сопоставить интервалы значений в которые попали данные. Данный метод также помогает оценить вид распределения выборки



а) б)

Рисунок 25 – графическое представление данных в виде а) графика; б) гистограммы.

Также графическое представление используется для предоставления информации об аномалии в анализируемой выборке (рисунок 28).

Аналитическое представление

Окно аналитического представления предоставляет информацию, представленную с помощью методов описательной статистики, в аналитической форме. Также в этом окне присутствует инструмент для изменения данных в базе данных для создания выбросов в данных в целях тестирования (рисунок 26). Для возврата исходного значения можно воспользоваться загрузкой данных за тот день, за который было изменено значение.

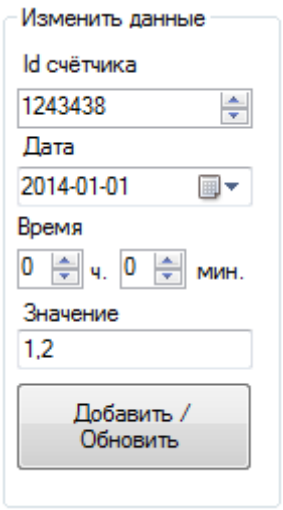


Рисунок 26 – окно изменения данных.

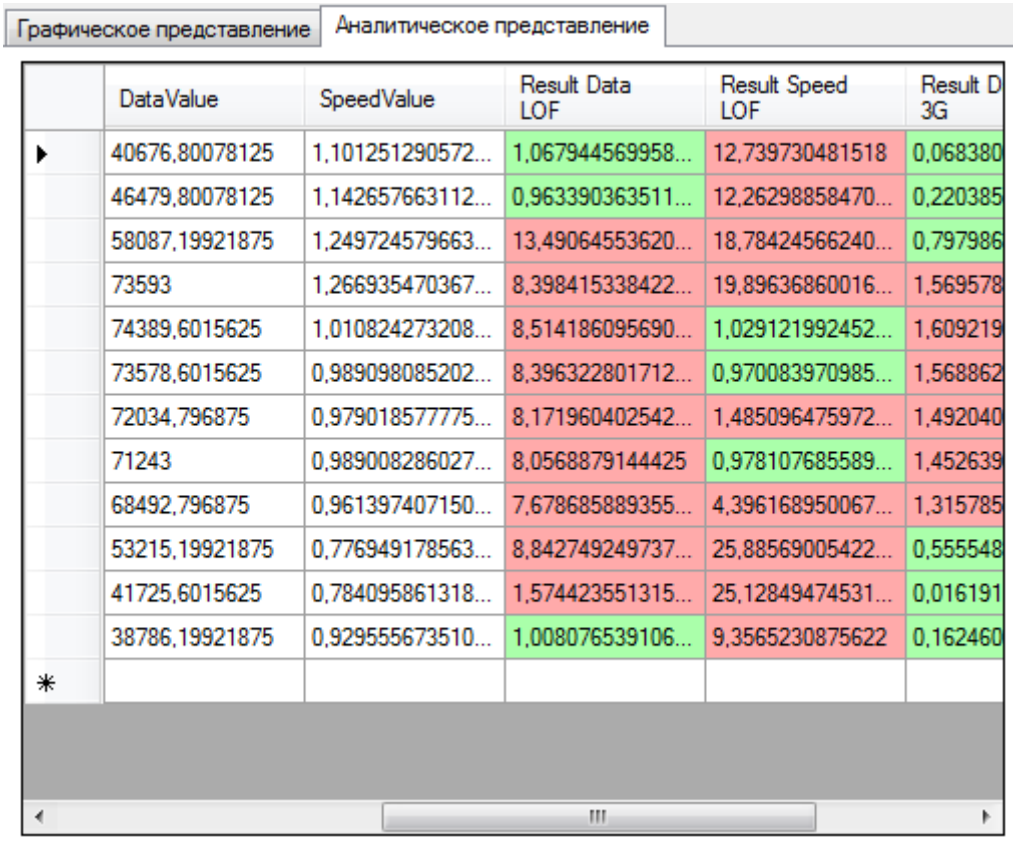
Для работы с изменёнными данными необходимо переформировать выборку.

Проведение анализа

Окно проведения анализа позволяет выбрать необходимый тип данных (посещаемость или скорость её изменения), алгоритм анализа и размер анализируемой выборки (анализ производится над последними данными в сформированной выборке).

Для графического отображения результатов анализа используется график анализируемой выборки (рисунок 28). Для аналитического отображения результатов используются интуитивно-понятные цвета, которые сигнализируют о преодолении заданного порогового значения (рисунок 27).

Пороговое значение для метода 3G считается 1.0, так как аналитическое представление результата численно показывает насколько близко значение к установленной границе, где 1.0 показывает, что значение лежит на границе.



	DataValue	SpeedValue	Result Data LOF	Result Speed LOF	Result D 3G
▶	40676,80078125	1,101251290572...	1,067944569958...	12,739730481518	0,068380
	46479,80078125	1,142657663112...	0,963390363511...	12,26298858470...	0,220385
	58087,19921875	1,249724579663...	13,49064553620...	18,78424566240...	0,797986
	73593	1,266935470367...	8,398415338422...	19,89636860016...	1,569578
	74389,6015625	1,010824273208...	8,514186095690...	1,029121992452...	1,609219
	73578,6015625	0,989098085202...	8,396322801712...	0,970083970985...	1,568862
	72034,796875	0,979018577775...	8,171960402542...	1,485096475972...	1,492040
	71243	0,989008286027...	8,0568879144425	0,978107685589...	1,452639
	68492,796875	0,961397407150...	7,678685889355...	4,396168950067...	1,315785
	53215,19921875	0,776949178563...	8,842749249737...	25,88569005422...	0,555548
	41725,6015625	0,784095861318...	1,574423551315...	25,12849474531...	0,016191
	38786,19921875	0,929555673510...	1,008076539106...	9,3565230875622	0,162460
*					

Рисунок 27 – аналитическое представление результатов алгоритмов анализа над комбинированной выборкой.

Для возможности сравнения работы алгоритмов, а также поведения данных и скорости изменения данных, предусмотрены комбинированные режимы: режим работы алгоритмов анализа и режим использования данных о посещаемости и о скорости изменения посещаемости.

При использовании комбинированных данных, графическое отображение результатов производиться в случае превышения порогового значения для каждого из типов данных.

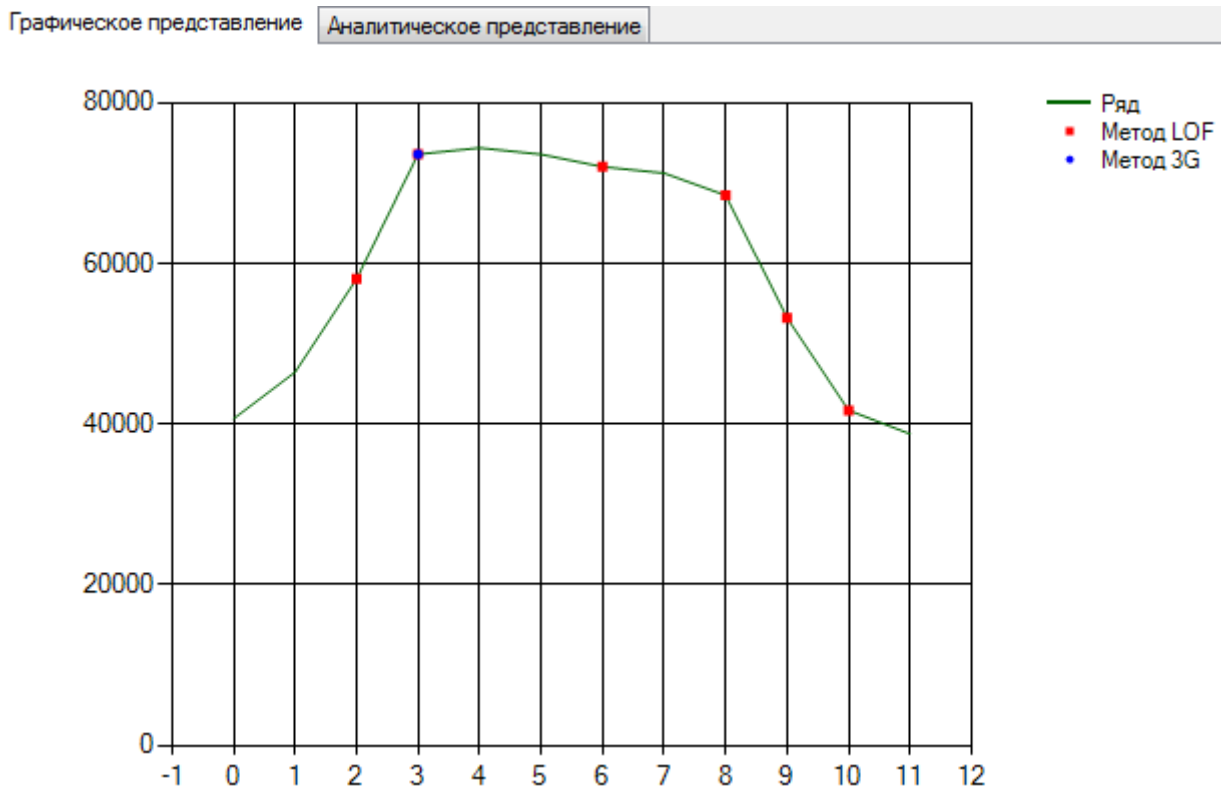


Рисунок 28 – графическое отображение результатов работы алгоритмов над комбинированными данными.

3.4.6. Алгоритм работы пользователя

Типичный процесс работы пользователя в данном приложении представляет собой следующий алгоритм:

1. Загрузить информацию о посещаемости интересующего электронного ресурса при наличии его статистики в открытом доступе. Информацию о доступе и номере счётчика можно найти на сайте http://top.mail.ru.
2. Сформировать выборку за определённый промежуток времени в определённые дни недели и определённые часы дня.
3. При необходимости внести правки в данные и переформировать выборку.
4. Выбрать тип данных для анализа и алгоритмы для анализа.
5. Провести анализ. Получить результаты.
6. При необходимости откалибровать пороговые значения для дальнейшего использования.

4. Исследовательский раздел

Для решения задачи поиска аномалий в статистических данных посещаемости сайта необходимо узнать природу этих данных, найти вспомогательные закономерности, выбрать необходимые алгоритмы, провести испытания.

4.1. Исследование характера данных

Посещаемость обеспечивается преимущественно пользователями, интересующимися информацией, содержащейся на сайте. Помимо данной категории пользователей посещаемость обеспечивается случайными посещениями от пользователей, перешедших с других ресурсов. Единичный доступ получают специальные поисковые роботы.

Пользователи, составляющие целевую аудиторию сайта, посещают сайт с некоторой закономерностью. Такая закономерность может обуславливаться различными факторами – среднестатистический период рабочего дня, временная направленность сайта (ночные чаты, утренние новости и т.д.), время публикации новостей или событий, ключевыми интересами общества в определённые промежутки времени и др.

Таким образом, для посещаемости определённого сайта можно выделить условную закономерность в характере этих данных. На рисунке 29 можно проследить изменение уровня посещаемости на протяжении недели. Периоды пиковой активности приходятся на дневное время. Небольшое падение характерно в период с 17 до 19 вечера, предположительно в конец рабочего дня.

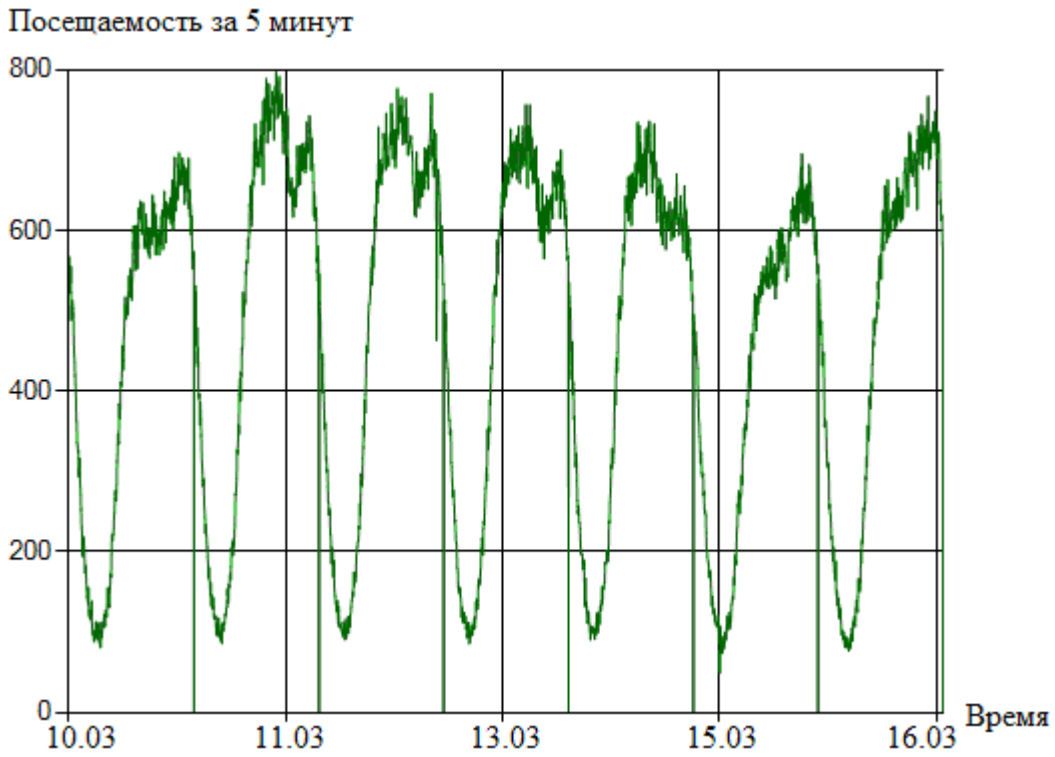


Рисунок 29 – посещаемость сайта http://forum.ixbt.com/
за неделю (с 2014-03-10 по 2014-03-16).

В выходные уровень посещаемости закономерно ниже, чем в будние дни. Данная тенденция прослеживается на протяжении нескольких недель. Также для любого буднего дня прослеживается закономерность спада в период окончания рабочего дня. На рисунке 30 можно проследить, что характер поведения данных в целом однороден для характерных дней недели.

Исходя из однородности поведения данных для характерных дней недели, для увеличения точности поиска аномалий необходимо сузить рассматриваемые интервалы времени в характерные дни. Например, на рисунке 31 взят интервал с 11:00 до 12:00 каждую среду с 2014-02-03 по 2014-04-23 для сайта http://forum.ixbt.com/.

Точный уровень данных в конкретные промежутки времени может довольно резко различаться (так как посещаемость в целом хаотична благодаря пользователям, которые попадают на сайт случайно). При этом общее соотношение посетителей за конкретный период относительно предыдущего периода будет отклоняться в меньшей степени. Таким образом, введём понятие скорости изменения данных, как отношение данных, полученных за текущий период, к данным, полученным за предыдущий период.

Преимущество данных о скорости изменения данных о посещаемости относительно самих данных о посещаемости в том, что данные о скорости имеют меньший разброс значений. Соответственно, выявить аномальные значения изменения

скорости проще. С увеличением посещаемости, разброс таких данных только уменьшается.

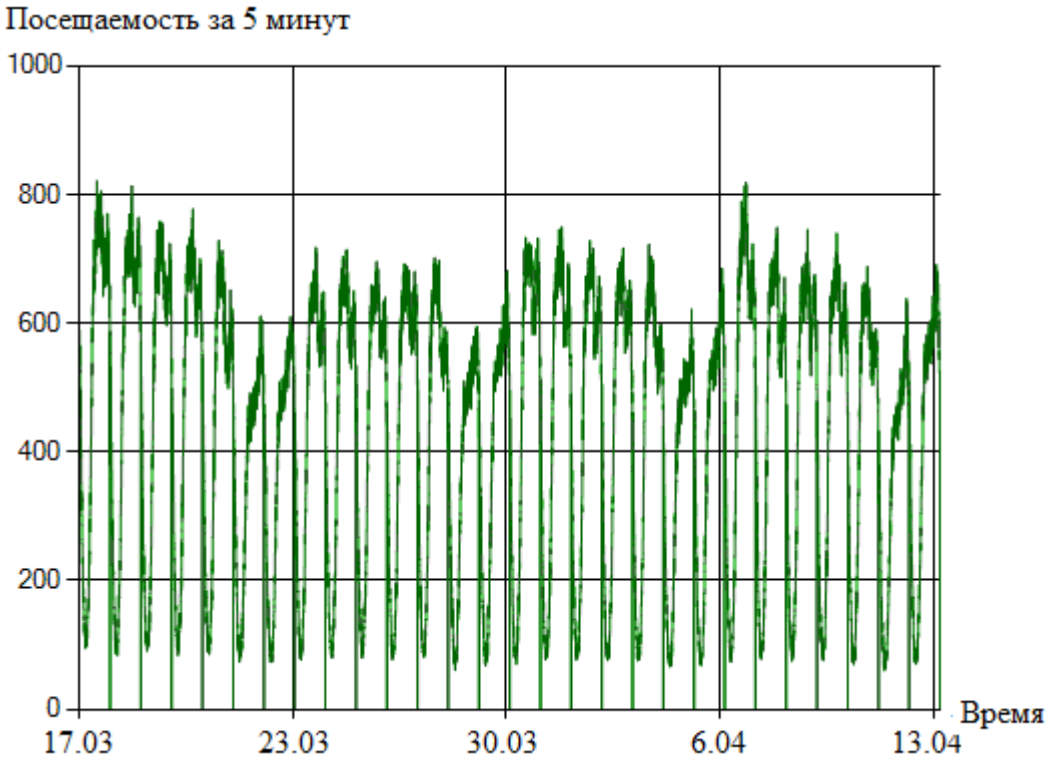


Рисунок 30 – посещаемость сайта http://forum.ixbt.com/ с 2014-03-17 по 2014-04-13.

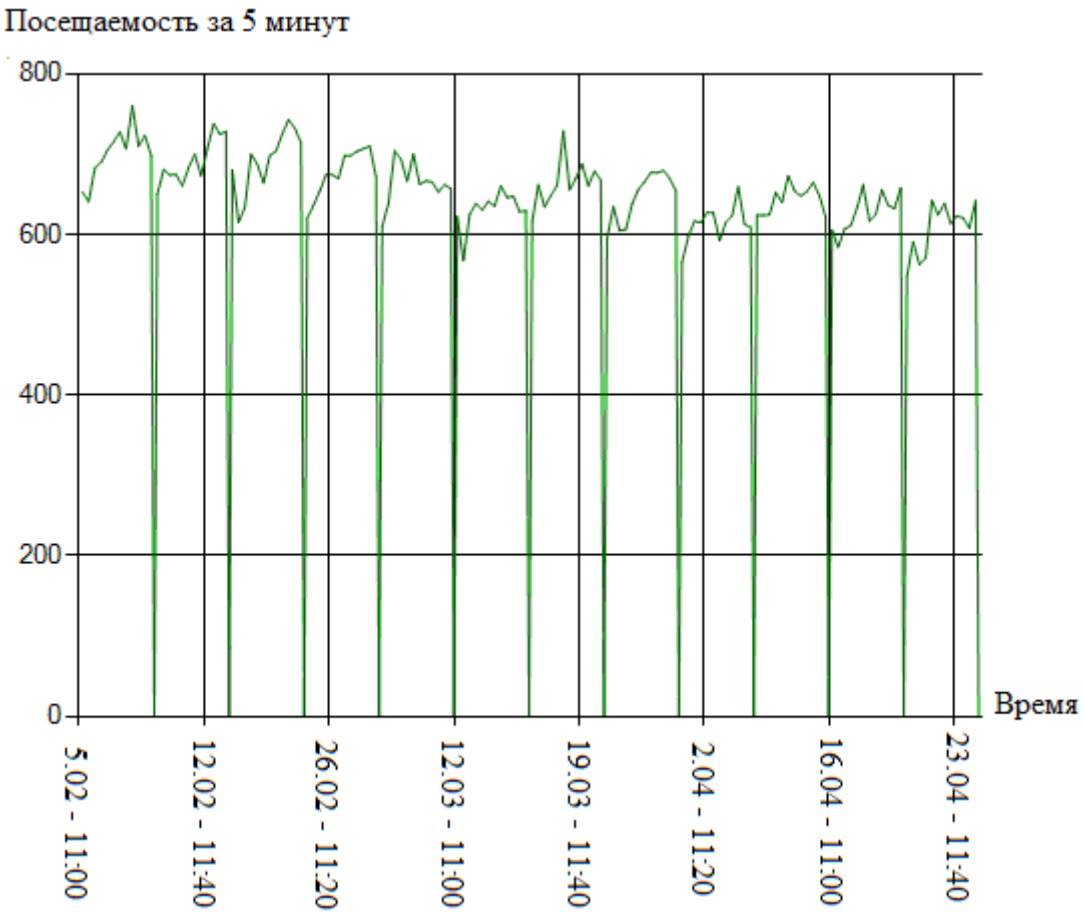


Рисунок 31 – посещаемость сайта http://forum.ixbt.com/ с 11:00 до 12:00 каждую среду с 2014-02-03 по 2014-04-23.

Так как у сайта есть общая тенденция к изменению совокупной посещаемости с течением длительного времени, для анализа необходимо учитывать наиболее поздние данные (например, за последние полгода).

4.2. Поведение алгоритмов LOF и 3G.

Алгоритмы LOF и 3G имеют различную природу. 3G основан на теории статистики в выборке данных, LOF является синтезом алгоритмов кластеризации и классификации. Следовательно, зафиксированные алгоритмами аномалии могут быть различны.

Основным недостатком алгоритма LOF перед алгоритмом 3G состоит в том, что данный алгоритм может находить аномалии не только на граничных значениях, отличных от общей выборки, но и в значениях, лежащих между граничных значений. Данные, ошибочно классифицируемые как аномалии, будем называть лжеаномалиями.

Основным недостатком алгоритма 3G является то, что анализируемые данные не обязательно будут подчиняться нормальному закону распределения. В данной работе

используется приближение о том, что данные являются нормально распределёнными. При этом, ожидается, что метод будет давать хорошие результаты при анализе скорости изменения данных, т.к. эти данные более приближены к нормальным (рисунок 32).

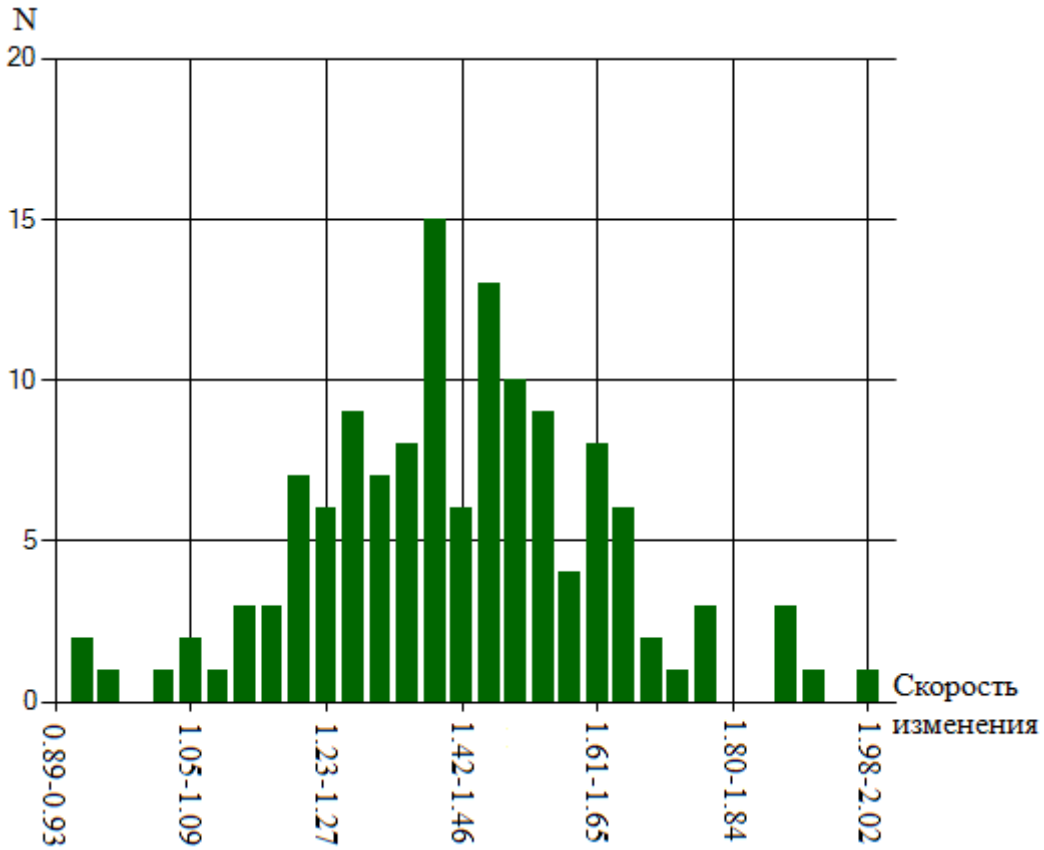


Рисунок 32 – распределение данных о скорости изменения посещаемости сайта http://forum.ixbt.com/ с 11:00 до 12:00 каждую среду с 2014-02-03 по 2014-04-23.

Для тестирования алгоритмов была проведена серия экспериментов. Алгоритм LOF использует 10 соседей и пороговое значение равное 1,5. Алгоритм 3G использует интервал (MX – 3G, MX + 3G). В экспериментах использовались выборки данных с 11:00 до 12:00 каждую среду с 2014-02-03 по 2014-04-23 для данных с различной среднесуточной посещаемостью. В расчёт были взяты случайные сайты представляющие малую среднесуточную посещаемость (меньше 1000), среднюю (от 1000 до 500 000), крупную (от 500 000). Используемые сайты:

- «Каталог программиста» http://articles.org.ru/ (среднесуточная посещаемость 500 посетителей).
- «Конференция iXBT.com: всё о железе и не только» http://forum.ixbt.com/ (среднесуточная посещаемость 65 000 посетителей).
- «Мой Мир@Mail.Ru» http://my.mail.ru (среднесуточная посещаемость 3 000 000 посетителей).

Данные о посещаемости этих сайтов находятся в открытом доступе и были скачены с сайта http://top.mail.ru. В данном тестировании алгоритмы применялись для трёх случаев:

1. В случае отсутствия возмущений – без аномалий.
2. В случае внесения возмущений, отклоняющих значение не более чем на 10% - слабых аномалий.
3. В случае внесения возмущений, отклоняющих значение более чем на 10% - сильных аномалий.

Результаты экспериментов приведены в таблице 7.

Таблица 7 – результаты тестирования алгоритмов LOF и 3G. Обозначения: НН – не найдены, НВ – найдены все, ЛА – лжеаномалии, ПС – по скорости, ПД – по данным.

Алгоритмы	LOF			3G		
Посещаемость	Малая	Средняя	Крупная	Малая	Средняя	Крупная
Без аномалий	НН, 3ЛА ПС	НН	НН	НН, 2ЛА ПС	НН	НН
Слабые аномалии	НВ, 5ЛА ПС	НВ	НВ ПС, 50% ПД	НВ	НВ ПД, 50% ПС	НН ПД, 50% ПС
Сильные аномалии	НВ, 5ЛА ПС	НВ	НВ ПС, 50% ПД	НВ	НВ	НВ ПС, 50% ПД

Из результатов эксперимента можно увидеть, что при использовании алгоритма 3G практически не возникает лжеаномалий, однако общий процент найденных аномалий уменьшается при увеличении посещаемости ресурса. Также из результатов следует, что алгоритм LOF является более чувствительным к слабым аномалиям. Большое количество лжеаномалий определяется достаточно большой разницей (в процентном соотношении) значений для сайтов с малой посещаемостью. Для таких сайтов рекомендуется резко завышать пороговое значение алгоритма LOF.

При использовании варианта совместной работы данные считаются аномальными, если оба алгоритма просигнализируют об аномальности для одного и того же типа данных.

Таблица 8 – результаты тестирования комбинации алгоритмов. Обозначения: НН – не найдены, НВ – найдены все, ЛА – лжеаномалии, ПС – по скорости, ПД – по данным.

Алгоритм	Комбинированный		
Посещаемость	Малая	Средняя	Крупная
Без аномалий	НН, 1ЛА ПС	НН	НН

Слабые аномалии	НВ ПД, НН ПС, 1 ЛА ПС	НВ: 50% ПС, 50% ПД	НВ ПС, НН ПД
Сильные аномалии	НВ ПД, НВ ПС	НВ: 50% ПС, НВ ПД	НВ ПС, НН ПД

Исходя из полученных результатов, чем больше среднесуточная посещаемость электронного ресурса, тем эффективнее работает анализ по данным о скорости посещаемости. При этом анализ по данным о посещаемости помечает данные, как аномальные в случае действительно критичных отклонений.

4.3. Выводы

Для результативного поиска аномалий необходимо использовать наиболее поздние данные, учитывать данные о скорости изменения посещаемости. Наиболее статистически значимая выборка получается путём выбора данных в определённые дни недели и временные диапазоны в эти дни. Рекомендуется анализировать конкретный элемент выборки на предмет аномальности, рассматривая статистическую выборку с пороговыми часовыми ограничениями в полчаса в обе стороны от времени этого элемента.

Для наибольшего покрытия выявленных аномалий необходимо применять комплекс методов. Целесообразнее проводить анализ на сайтах со средней или крупной среднестатистической посещаемостью. Алгоритм LOF порождает лжеаномалии. Алгоритм 3G может оказаться недостаточно точным. Для результативного поиска аномалий достаточно связки алгоритмов LOF и 3G откалиброванных под нужды пользователя.

Наиболее полное покрытие обеспечивается при рассмотрении данных как аномальных при каждом срабатывании на данных о скорости и данных о посещаемости для обоих алгоритмов. При этом, для повышения эффективности работы на сайтах с малой среднесуточной посещаемостью рекомендуется использовать более высокие пороговые значения.

5. Организационно-экономический раздел

5.1. Организация и планирование процесса разработки

Организация и планирование процесса разработки приложения или программного комплекса при традиционном методе планирования предусматривает выполнение следующих работ:

- формирование состава выполняемых работ и группировка их по стадиям разработки;
- расчет трудоемкости выполнения работ;
- формирование профессионального состава и расчет количества исполнителей;
- определение продолжительности выполнения отдельных этапов разработки;
- расчет календарного графика выполнения разработки;
- контроль выполнения календарного графика.

5.2. Формирование состава выполняемых работ и группировка их по стадиям

Техническое задание. Постановка задач. Определение пакета прикладных программ, состава и структуры информационной базы. Выбор языков программирования. Предварительный выбор методов выполнения работы. Разработка календарного плана выполнения работ.

Эскизный проект. Предварительная разработка структуры входных и выходных данных. Разработка общего описания алгоритмов решения задач. Разработка пояснительной записки. Согласование и утверждение эскизного проекта.

Технический проект. Разработка алгоритмов решения задач. Разработка пояснительной записки. Согласование и утверждение технического проекта. Разработка структуры программы. Разработка программной документации и передача ее для включения в технический проект. Внесение правок в структуры, анализ и определение формы представления входных и выходных данных. Выбор конфигурации технических средств.

Рабочий проект. Комплексная отладка задач и сдача в опытную эксплуатацию. Разработка проектной документации. Программирование и отладка программ. Описание контрольного примера. Разработка программной документации.

Внедрение. Подготовка и передача программной документации для сопровождения с оформлением соответствующего акта. Передача программной продукции в фонд алгоритмов и программ. Проверка алгоритмов и программ решения задач, корректировка документации после опытной эксплуатации программного продукта.

Планирование длительности этапов и содержания проекта осуществляется в соответствии с ЕСПД ГОСТ 34.603-92 и распределяет работы по этапам, как показано в таблице 9.

Таблица 9 - распределение работ проекта по этапам.

Основные стадии	**№**	**Содержание работы**
1. Техническое задание	1	Постановка задачи
	2	Выбор средств проектирования, разработки и СУБД
2. Эскизный проект	3	Разработка структурной схемы системы
	4	Разработка структуры базы данных
	5	Разработка алгоритмов доступа к данным
	6	Разработка алгоритмов решения частных задач
3. Технический-рабочий проект	7	Реализация алгоритмов доступа к данным
	8	Реализация алгоритмов решения частных задач
	9	Разработка пользовательского интерфейса
	11	Реализация пользовательского интерфейса
	12	Отладка и тестирование всего комплекса информационной среды
	13	Исправление ошибок и недочетов
	14	Разработка документации к системе
	15	Итоговое тестирование системы
4. Внедрение	16	Установка и настройка ПП

5.3. Расчет трудоемкости выполнения работ

Трудоемкость разработки программной продукции зависит от ряда факторов, основными из которых являются следующие:

- степень новизны разрабатываемого программного комплекса;
- сложность алгоритмов его функционирования;
- объем используемой информации, вид ее представления и способ обработки;
- уровень используемого алгоритмического языка программирования (чем выше уровень языка, тем меньше трудоемкость).

Исходные данные расчета представлены в таблице 10.

Таблица 10 - факторы, влияющие на разработку.

Степень новизны разрабатываемого проекта	Группа новизны В - разработка программной продукции, имеющей аналоги.
Степень сложности алгоритма функционирования	3 группа сложности - программная продукция, реализующая алгоритмы стандартных методов решения задач.
По виду представления исходной информации	Группа 12 - исходная информация представлена в форме документов, имеющих одинаковый формат и структуру, требуется форматный контроль информации.
Структура выходных документов	Требуется вывод на печать одинаковых документов, вывод информационных массивов на машинные носители.

Трудоемкость разработки программной продукции $\tau_{ПП}$ может быть определена как сумма величин трудоемкости выполнения отдельных стадий разработки ПП из выражения:

$\tau_{ПП} = \tau_{ТЗ} + \tau_{ЭП} + \tau_{ТП} + \tau_{РП} + \tau_{В}$, (19), где

$\tau_{ТЗ}$ – трудоемкость разработки технического задания на создание ПП;

$\tau_{ЭП}$ – трудоемкость разработки эскизного проекта ПП;

$\tau_{ТП}$ – трудоемкость разработки технического проекта ПП;

$\tau_{РП}$ – трудоемкость разработки рабочего проекта ПП;

$\tau_{В}$ - трудоемкость внедрения разработанного ПП.

Трудоемкость разработки технического задания рассчитывается по формуле:

$\tau_{ТЗ} = T_{РЗ}^{3} + T_{РП}^{3}$, (20), где

$T_{РЗ}^{3}$ – затраты времени разработчика постановки задач на разработку ТЗ, чел.-дни;

$T_{РП}^{3}$ – затраты времени разработчика программного обеспечения на разработку ТЗ, чел.-дни.

Значения величин $T_{РЗ}^{3}$ и $T_{РП}^{3}$ рассчитываются по формулам

$T_{РЗ}^{3} = t_{З} \cdot K_{РЗ}^{3}$; (21)

$T_{РП}^{3} = t_{З} \cdot K_{РП}^{3}$, (22), где

$t_{З}$ – норма времени на разработку ТЗ на программный продукт в зависимости от функционального назначения и степени новизны разрабатываемого ПП, чел.-дни;

$K_{РЗ}^{З}$ – коэффициент, учитывающий удельный вес трудоемкости работ, выполняемых разработчиком постановки на стадии ТЗ;

$K_{РП}^{З}$ – коэффициент, учитывающий удельный вес трудоемкости работ, выполняемых разработчиком программного обеспечения на стадии ТЗ.

$t_З = 36$ [чел.-дн.]

$K_{РЗ}^{З} = 0,75$

$K_{РП}^{З} = 0,25$

$\tau_{ТЗ} = 36 \cdot (0,75 + 0,25) = 36$ [чел.-дн.]

Аналогично рассчитывается трудоемкость эскизного проекта ПП $\tau_{ЭП}$:

$\tau_{ЭП} = T_{РЗ}^{Э} + T_{РП}^{Э}$ (23)

$T_{РЗ}^{Э} = t_Э \cdot K_{РЗ}^{Э} = 50 \cdot 0,65$

$T_{РП}^{Э} = t_Э \cdot K_{РЗ}^{Э} = 50 \cdot 0,35$

$\tau_{ЭП} = 50 \cdot (0,65 + 0,35) = 50$ [чел.-дн.]

Трудоемкость разработки технического проекта $\tau_{ТП}$ зависит от функционального назначения ПП, количества разновидностей форм входной и выходной информации и определяется как сумма времени, затраченного разработчиком постановки задач и разработчиком программного обеспечения, т.е.

$\tau_{ТП} = (t_{РЗ}^{Т} + t_{РП}^{Т}) \cdot K_В \cdot K_Р,$ (24), где

$t_{РЗ}^{Т}$, $t_{РП}^{Т}$ – норма времени, затрачиваемого на разработку ТП разработчиком постановки задач и разработчиком программного обеспечения соответственно, чел.-дни;

$K_В$ – коэффициент учета вида используемой информации;

$K_Р$ – коэффициент учета режима обработки информации.

Значение коэффициента $K_В$ определяется из выражения:

$K_В = (K_П \cdot n_П + K_{НС} \cdot n_{НС} + K_Б \cdot n_Б) / (n_П + n_{НС} + n_Б)$ (25), где

$K_П$, $K_{НС}$, $K_Б$ – значения коэффициентов учета вида используемой информации для переменной, нормативно-справочной информации и баз данных соответственно;

$n_П$, $n_{НС}$, $n_Б$ – количество наборов данных переменной, нормативно-справочной информации и баз данных соответственно.

$K_Р = 1,36$, $K_В = 1,27$

$\tau_{ТП} = (38 + 16) \cdot 1,27 \cdot 1,36 = 94$ [чел.-дн.]

Трудоемкость разработки рабочего проекта $\tau_{РП}$ зависит от функционального назначения ПП, количества разновидностей форм входной и выходной информации, сложности алгоритма функционирования, сложности контроля информации, степени

использования готовых программных модулей, уровня алгоритмического языка программирования и определяется по формуле:

$\tau_{РП} = K_К \cdot K_Р \cdot K_Я \cdot K_З \cdot K_{ИА} \cdot (t_{РЗ}^{Р} + t_{РП}^{Р})$, (26), где

$K_К$ – коэффициент учета сложности контроля информации;

$K_Я$ – коэффициент учета уровня используемого алгоритмического языка программирования;

$K_З$ – коэффициент учета степени использования готовых программных модулей;

$K_{ИА}$ – коэффициент учета вида используемой информации и сложности алгоритма ПП;

$t_{РЗ}^{Р}$, $t_{РП}^{Р}$ – норма времени, затраченного на разработку РП на алгоритмическом языке высокого уровня разработчиком постановки задач и разработчиком программного обеспечения соответственно, чел.-дни.

Значение коэффициента $K_{ИА}$ определяется из выражения

$K_{ИА} = (K_П' \cdot n_П + K_{НС}' \cdot n_{НС} + K_Б' \cdot n_Б) / (n_П + n_{НС} + n_Б)$, (27), где

$K_П'$, $K_{НС}'$, $K_Б'$ – значения коэффициентов учета сложности алгоритма ПП и вида используемой информации для переменной, нормативно-справочной информации и баз данных соответственно.

$K_К = 1$

$K_Р = 1{,}44$ (для рабочего проекта)

$K_Я = 1$

$K_З = 0{,}5$

$t_{РЗ}^{Р} = 12$ [чел.-дн.]

$t_{РП}^{Р} = 70$ [чел.-дн.]

$K_П' = 1$

$K_{НС}' = 0{,}48$

$K_Б' = 0{,}4$

$K_{ИА} = 0{,}63$

$\tau_{РП} = (12 + 70) * 1 * 1.44 * 1 * 0.5 * 0.63 = 38$ [чел.-дн.]

Так как при разработке ПП стадии «Технический проект» и «Рабочий проект» объединены в стадию «Техно-рабочий проект», то трудоемкость ее выполнения $\tau_{ТРП}$ определяется по формуле:

$\tau_{ТРП} = 0{,}85 \cdot \tau_{ТП} + \tau_{РП}$ (28)

$\tau_{ТРП} = 0{,}85 \cdot 94 + 38 = 118$ [чел.-дн.]

Трудоемкость выполнения стадии внедрения $\tau_В$ может быть рассчитана по формуле:

$\tau_В = (t_{РЗ}^{В} + t_{РП}^{В}) \cdot K_К \cdot K_Р \cdot K_З$, (29), где

$t_{РЗ}^{В}$, $t_{РП}^{В}$ – норма времени, затрачиваемого разработчиком постановки задач и разработчиком программного обеспечения соответственно на выполнение процедур внедрения ПП, чел.-дни.

$K_Р = 1,26$

$t_{РЗ}^{В} = 16$ [чел.-дн.]

$t_{РП}^{В} = 24$ [чел.-дн.]

$\tau_В = (16 + 24) \cdot 1 \cdot 1,26 \cdot 0,5 = 26$ [чел.-дн.]

Подставляя полученные данные в (19), получим:

$\tau_{ПП} = 36 + 50 + 118 + 26 = 230$ [чел.-дн.]

Таблица 11 - трудоемкости по стадиям разработки проекта.

Этап	**Трудоемкость этапа**	**№ работы**	**Содержание работы**	**Трудоемкость чел-дн.**
1	36	1	Постановка задачи	26
		2	Выбор средств проектирования, разработки и СУБД	10
2	50	3	Разработка структурной схемы системы	15
		4	Разработка структуры базы данных	10
		5	Разработка алгоритмов доступа к данным	10
		6	Разработка алгоритмов решения частных задач	15
3	118	7	Реализация алгоритмов доступа к данным	16
		8	Реализация алгоритмов решения частных задач	15
		9	Разработка пользовательского интерфейса	10
		10	Реализация пользовательского интерфейса	14
		11	Отладка и тестирование всего комплекса информационной среды	20
		12	Исправление ошибок и недочетов	8
		13	Разработка документации к системе	20
		14	Итоговое тестирование системы	15

Этап	Трудоемкость этапа	№ работы	Содержание работы	Трудоемкость чел-дн.
4	26	15	Установка и настройка ПП	26
Всего	230 чел-дн			230 чел-дн

5.4. Расчет количества исполнителей

Средняя численность исполнителей при реализации проекта разработки и внедрения ПО определяется соотношением: $N = \frac{Q_p}{F}$, где:

Q_p - затраты труда на выполнение проекта (разработка и внедрение ПО),

F - фонд рабочего времени.

Величина фонда рабочего времени определяется соотношением:

$F = T \cdot F_m$, (30), где

T - время выполнения проекта в месяцах. T = 4 мес.;

F_M - фонд времени в текущем месяце, который рассчитывается из учета общества числа дней в году, числа выходных и праздничных дней:

$$F_м = \frac{t_p \cdot (D_K - D_B - D_П)}{12}, \quad (31), \text{ где}$$

t_p - продолжительность рабочего дня;

D_K - общее число дней в году;

D_B - число выходных дней в году;

$D_П$ - число праздничных дней в году.

$$F_м = \frac{t_p \cdot (D_K - D_B - D_П)}{12} = \frac{8 \cdot (365 - 103 - 13)}{12} = 166 \quad (32)$$

$F = T \cdot Fm = 4 \cdot 166 = 664$ $\quad$ $Qp = 8 \cdot 230 = 1840$

$N = \frac{Q_p}{F} = \frac{1840}{664}$ = 3 - число исполнителей проекта.

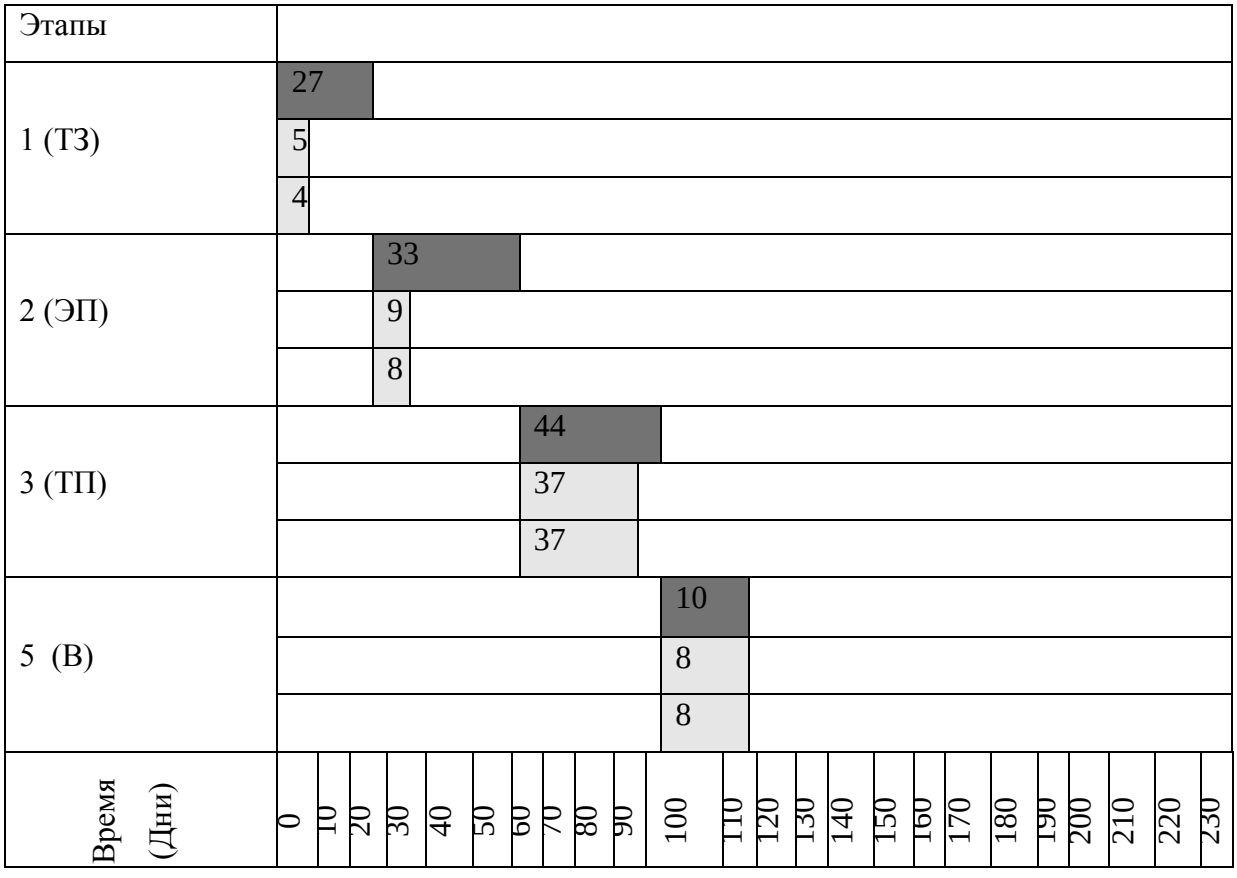
5.5. Календарный план-график разработки ПП

Планирование и контроль хода выполнения разработки проводится по календарному графику выполнения работ.

Таблица 12 - планирование разработки.

Стадия разработки	Трудоемкость	Должность исполнителя	Распределение трудоемкости	Численность
1.Разработка технического задания	36	Ведущий инженер Программист1 Программист2	27 5 4	1 1 1
2.Разработка эскизного проекта	50	Ведущий инженер Программист1 Программист2	33 9 8	1 1 1
3.Разработка технического проекта	118	Ведущий инженер Программист1 Программист2	44 37 37	1 1 1
4.Внедрение	26	Ведущий инженер Программист1 Программист2	10 8 8	1 1 1
Итого:	230			3

Таблица 13 - календарный ленточный график работ.



	Ведущий инженер	
	Программисты	

Вывод: при использовании оптимизации в результате параллельной работы ведущего инженера и программистов можно добиться сокращения срока разработки и внедрения программного продукта с 230 дней до 114 дней, т. е. в 2,02 раза.

5.6. Определение цены программной продукции

Затраты на выполнение проекта состоят из затрат на заработную плату исполнителям, затрат на закупку или аренду оборудования, затрат на организацию рабочих мест, и затрат на накладные расходы.

В таблицах 13-15 приведены затраты на заработную плату и отчисления на социальное страхование в пенсионный фонд, фонд занятости и фонд обязательного медицинского страхования (30 %). Для всех исполнителей предполагается оклад в размере 50000 рублей в месяц.

Таблица 14 - затраты на зарплату и отчисления на социальное страхование (начало).

	Февраль			март		
Исполнитель	рабочих дней	зарплата	ЕСН	рабочих дней	зарплата	ЕСН
1	20	41666	12499,8	24	50000	15000
2	5	10416	3124,8	9	18749	5624,7
3	4	8333	2499,9	8	16666	4999,8
Итого:	78539,5			111039,5		

Таблица 15 - затраты на зарплату и отчисления на социальное страхование (продолжение).

	апрель			май		
Исполнитель	рабочих дней	зарплата	ЕСН	рабочих дней	зарплата	ЕСН
1	24	50000	15000	22	45833	13749,9
2	8	16666	4999,8	22	45833	13749,9
3	8	16666	4999,8	22	45833	13749,9
Итого:	108331,6			178748,7		

Таблица 16 - затраты на зарплату и отчисления на социальное страхование (окончание).

	Июнь			∑, руб.
Исполнитель	рабочих дней	зарплата	ЕСН	
1	24	50000	15000	
2	15	31250	9375	
3	15	31250	9375	
Итого:	146250			622909,3

Расходы на материалы, необходимые для разработки программной продукции, указаны в таблице.

Таблица 17. Затраты на материалы.

№	Наименование материала	Единица измерения	Кол-во	Цена за единицу, руб.	Сумма, руб.
1	Бумага А4	Пачка 400 л.	2	200	400
2	Картридж для принтера HP P10025	Шт.	3	450	1350
Всего					1750

В работе над проектом используется специальное оборудование – персональные электронно-вычислительные машины (ПЭВМ) в количестве 3 шт. Стоимость одной ПЭВМ составляет 20000 рублей. Месячная норма амортизации К = 2,7%.

$$K_a = \frac{1}{n} * 100\% = \frac{1}{36} * 100\% \qquad (33)$$

Тогда за 5 месяцев работы расходы на амортизацию составят 8100 рублей.

R = 20000 · 3 · 0.027 · 5 = 8100 рублей.

Общие затраты на разработку программного продукта (ПП) составят 632759 рублей.

5.7. Расчет стоимости программного продукта

$$Ц = С + П_р \qquad (34)$$

$С$ - затраты на разработку программной продукции

$П_р$ - желаемая прибыль

$$П_р = \frac{(C - C_м) p_н}{100\%}$$ (35), где

$C_м$ - материальные затраты, руб./изд

$p_м$ - норматив рентабельности, принимаемый разработчиком

Ц = 632759 + (632759 – 8100 - 1750) · 0,25 = 788486,25 руб.

5.8. Расчет экономической эффективности

Основными показателями экономической эффективности является чистый дисконтированный доход (ЧДД) и срок окупаемости вложенных средств.

Чистый дисконтированный доход определяется по формуле:

$$ЧДД = \sum_{t=0}^{T} (R_t - З_t) \cdot \frac{1}{(1+E)^t}$$ (36),

где Т – горизонт расчета по месяцам;

t – период расчета;

Rt – результат, достигнутый на t шаге (стоимость);

Зt – затраты;

Е – приемлемая для инвестора норма прибыли на вложенный капитал.

Коэффициент Е установим равным ставке рефинансирования ЦБ РФ – 8.25% годовых (или 0.6875% в месяц). В виду особенности разрабатываемого продукта он может быть продан лишь однократно.

Коэффициент дисконтирования равен 1/(1 + Е) = 0,9933.

В таблице 18 приведен расчет ЧДД по месяцам работы над проектом.

Таблица 18 - расчет ЧДД.

Месяц	**Текущие затраты, руб.**	**Затраты с начала года, руб.**	**Текущий доход, руб.**	**ЧДД, руб.**
Февраль	88389,5	88389,5	0	-87505,6
Март	111039,5	199429	0	-195460
Апрель	108331,6	307760,6	0	-298620
Май	178748,7	486509,3	0	-467339
Июнь	146250	632759,3	788486,25	148094,8

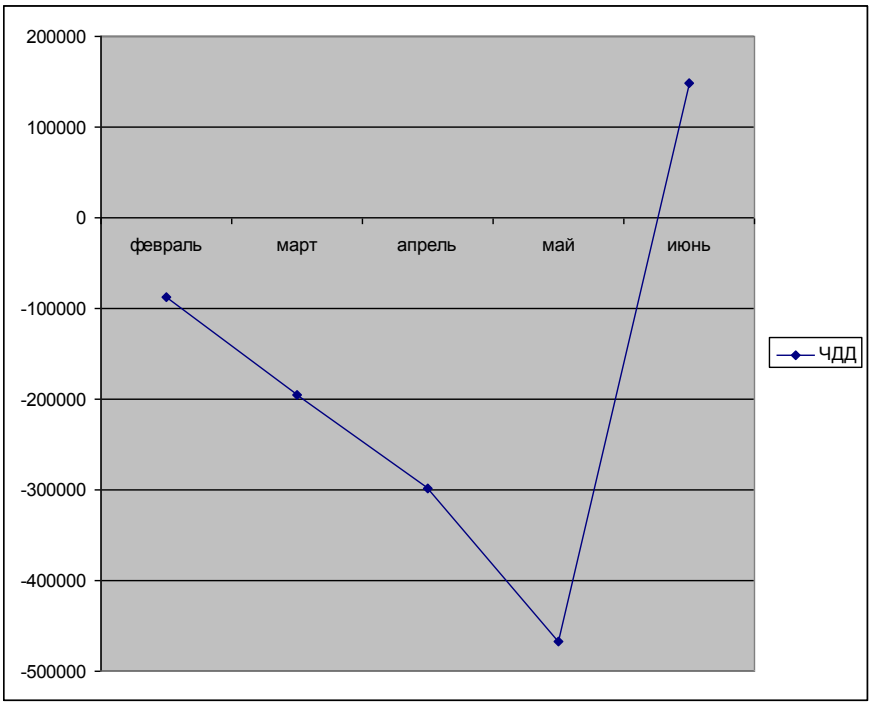


Рисунок 33 - график изменения чистого дисконтированного дохода.

5.9. Вывод

Согласно проведенным расчетам, проект является рентабельным.

Разрабатываемый проект позволит превысить показатели качества существующих систем и сможет их заменить.

Итоговый ЧДД составил 148094 рублей. Срок реализации проекта равен 5 месяцам.

6. Промышленная экология и безопасность

6.1. Анализ опасных и вредных факторов при разработке программного обеспечения и мероприятия по их устранению

Разработка программного обеспечения требует длительного взаимодействия с вычислительными системами. Основными факторами, влияющими на работу с персональными электронно-вычислительными машинами являются: освещённость, шум, вибрация, электромагнитные поля, статическое электричество, рентгеновское излучение, пожароопасность и травматизм. При длительном воздействии на организм вредные факторы негативно влияют на здоровье человека. Воздействие этих негативных факторов нормируется по СанПиН 2.2.2/2.4.1340-03.

6.2. Микроклимат

Работа, как программиста, так и пользователя относится к категории 1а, поскольку не предполагает больших физических усилий. Поэтому оптимальные нормы микроклимата для рабочего помещения программиста определяются таблицей (СанПиН 2.2.2/2.4.1340-03):

Таблица 19 – оптимальные нормы микроклимата

	Температура воздуха, ^{o}C	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	22-24	40-60	0,1
Теплый	23-25	40-60	0,1

Вредным фактором при работе с ЭВМ является также запыленность помещения. Этот фактор усугубляется влиянием на частицы пыли электростатических полей персональных компьютеров.

Для устранения несоответствия параметров указанным нормам проектом предусмотрено использование системы кондиционирования как наиболее эффективного и автоматически функционирующего средства.

Нормы СанПиН 2.2.4.1294-03 «Санитарно-гигиенические нормы допустимых уровней ионизации воздуха» определяют уровни положительных и отрицательных ионов в воздухе:

Таблица 20 - уровни ионизации воздуха помещений при работе на ВДТ и ПЭВМ.

Уровни	Число ионов в 1 см куб. воздуха	
	n^+	n^-
Минимально необходимые	400	600
Оптимальные	1500-3000	3000-5000
Предельно допустимые	50000	50000

Для обеспечения требуемых уровней предусмотрено использование системы ионизации Сапфир-4А.

Концентрация вредных химических веществ в помещениях с ПЭВМ не должна превышать «ПДК загрязняющих веществ в атмосферном воздухе населенных мест» ГН 2.1.6.789-99. Для выполнения указанных требований предусмотрено применение фильтров из активированного угля.

6.3. Шум и вибрации

Уровень вибрации не должен превышать допустимых норм вибрации. СанПиН 2.2.2.542-96 устанавливает следующие нормы на вибрацию (таблица 21).

Таблица 21 - допустимые нормы вибрации на рабочих местах с ВДТ и ПЭВМ.

Среднегеометрические частоты октавных полос, Гц	Допустимые значения по виброскорости	
	м/с	дБ
2	4,5x10	79
4	2,2x10	73
8	1,1x10	67
16	1,1x10	67
31,5	1,1x10	67
63	1,1x10	67
Корректированные значения и их уровни в дБ	2,0x10	72

При разработке программного обеспечения внутренними источниками шума являются вентиляторы, а также принтеры и другие периферийные устройства ЭВМ.

Внешние источники шума – прежде всего, шум с улицы и из соседних помещений. Постоянные внешние источники шума, превышающего нормы, отсутствуют.

Уровни звукового давления в расчётной точке при заданных уровнях звуковой мощности источников в серверном помещении объёмом 15х30х4 (рисунок 34), с двумя источниками шума, расположенными на полу, с расстояниями до источника 12 и 8 метров в соответствии с таблицей 22 рассчитаны в таблице 23 и проанализированы в соответствии с нормативными значениями СН 2.2.4/2.1.8.562-96 из таблицы 24.

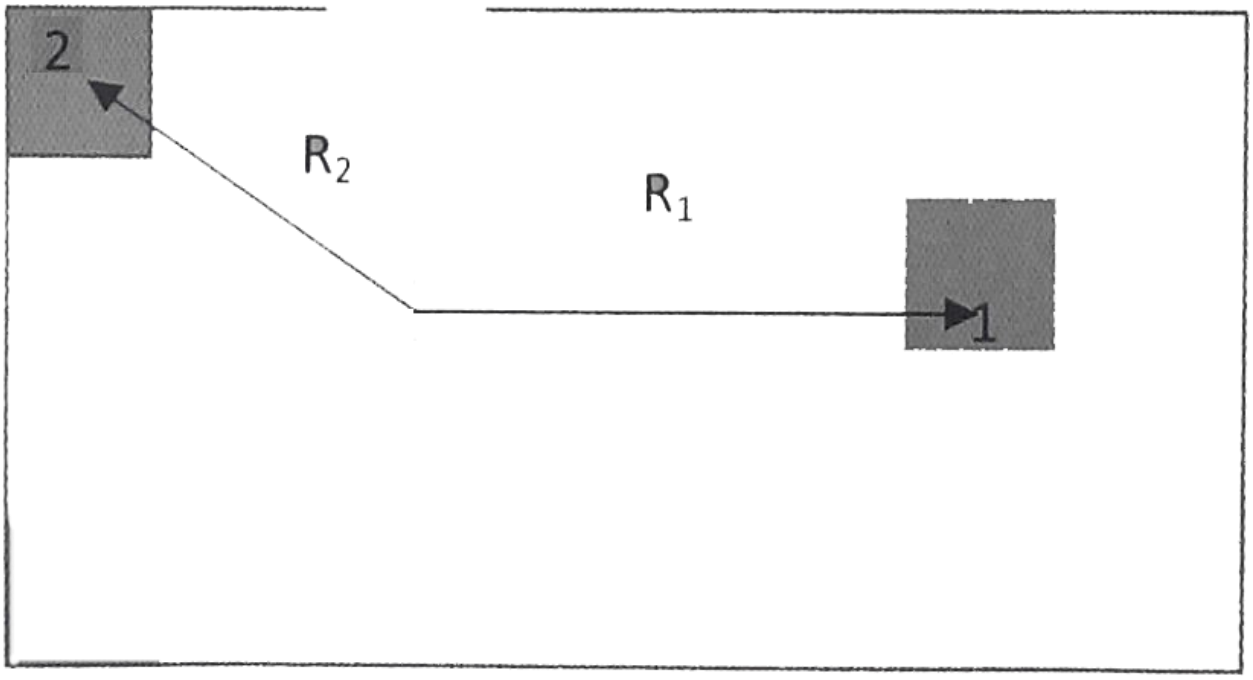


Рисунок 34 - схема серверного помещения с двумя источниками шума.

Таблица 22 - уровни звуковой мощности.

№ источника	Расстояние до источника	Уровни звуковой мощности (дБ)							
		63	125	250	500	1000	2000	4000	8000
1	12	90	91	98	99	97	93	91	86
2	8	84	82	84	91	94	94	91	91

Таблица 23 - уровни звуковой интенсивности.

№	Уровни звуковой мощности							
	63	125	250	500	1000	2000	4000	8000
1	74,436	75,436	82,039	82,043	78,595	72,749	68,442	61,221
3	69,636	67,636	69,338	75,623	77,669	76,619	72,581	71,852
L_sum	75,678	76,102	82,26	82,93	81,16	78,11	73,99	72,21

			6	5	7	2	7	2

Таблица 24 - Нормативные значения.

№	Уровни звуковой мощности							
	63	125	250	500	1000	2000	4000	8000
L_lim	83	74	68	63	60	57	55	54

Выводы по результатам: по всем октавным частотам превышен уровень шума. Необходимо провести ряд профилактических мероприятий, связанных с акустической обработкой помещения и акустической абсорбцией. Обработка пола, потолка, применение звукопоглощающих материалов для облицовки стен и потолка, установление дополнительных слоёв, обработка стен, установление звукопоглощающих панелей, шумоглушителей. В случае невозможности проведения таких мероприятий, обеспечить работников средствами индивидуальной защиты от шума (например, противошумы).

6.4. Освещение

Наиболее важным условием эффективной работы программистов и пользователей является соблюдение оптимальных параметров системы освещения в рабочих помещениях.

Естественное освещение осуществляется через светопроемы, ориентированные в основном на север и северо-восток (для исключения попадания прямых солнечных лучей на экраны компьютеров) и обеспечивает коэффициент естественной освещенности (КЕО) не ниже 1,5%.

В качестве искусственного освещения проектом предусмотрено использование системы общего равномерного освещения. В соответствии с СанПиН 2.2.2/2.4.1340-03 освещенность на поверхности рабочего стола находится в пределах 300-500 лк. Разрешается использование светильников местного освещения для работы с документами (при этом светильники не должны создавать блики на поверхности экрана).

Правильное расположение рабочих мест относительно источников освещения, отсутствие зеркальных поверхностей и использование матовых материалов ограничивает прямую (от источников освещения) и отраженную (от рабочих поверхностей) блескость. При этом яркость светящихся поверхностей не превышает 200 кд/кв.м, яркость бликов на экране ПЭВМ не превышает 40 кд/кв.м, и яркость потолка не превышает 200 кд/кв.м.

В соответствии с СанПиН 2.2.2/2.4.1340-03 проектом предусмотрено использование люминесцентных ламп типа ЛБ в качестве источников света при искусственном освещении. В светильниках местного освещения допускается применение ламп накаливания.

Применение газоразрядных ламп в светильниках общего и местного освещения обеспечивает коэффициент пульсации не более 5%.

Таким образом, проектом обеспечиваются оптимальные условия освещения рабочего помещения.

6.5. Визуальные параметры

Неправильный выбор визуальных эргономических параметров приводит к ухудшению здоровья пользователей, быстрой утомляемости, раздражительности. В этой связи, проектом предусмотрено, что конструкция вычислительной системы и ее эргономические параметры обеспечивают комфортное и надежное считывание информации.

Требования к визуальным параметрам, их внешнему виду, дизайну, возможности настройки представлены в СанПиН 2.2.2/2.4.1340-03. Визуальные эргономические параметры монитора и пределы их изменений приведены в таблице 29.

Таблица 29 - визуальные эргономические параметры ВДТ и пределы их изменений.

Наименование параметров	Пределы значений параметров	
	миним. (не менее)	максим. (не более)
Яркость знака (яркость фона), кд/кв.м (измеренная в темноте)	35	120
Внешняя освещенность экрана, лк	100	250
Угловой размер знака, угл. мин.	16	60

Для выполнения этих требований проектом предусмотрено использование современных мониторов, имеющих достаточно широкий набор регулируемых параметров. В частности, для удобного считывания информации реализована возможность настройки положения монитора по горизонтали и вертикали. Мониторы оснащены специальными устройствами и средствами настройки ширины, высоты, яркости, контраста и разрешения изображения. Кроме того, в современных мониторах зерно изображения имеет размер в пределах 0,27 мм, что обеспечивает высокую

четкость и непрерывность изображения. Наконец, на поверхность дисплея нанесено матовое покрытие, чтобы избавиться от солнечных бликов.

6.6. Расчет системы искусственного освещения

В зависимости от цели расчета при проектировании искусственного освещения приходится решать следующий ряд вопросов:

Выбрать или определить типы ламп и светильников. Для освещения предприятий службы быта следует применять газоразрядные лампы. Применение ламп накаливания целесообразно при температуре воздуха ниже 10 ^{o}C и падении напряжения в сети более 10% от номинального.

Выбор светильника должен производится с учетом его крепления, подвода электроэнергии, защиты от механических повреждений, взрыво- и пожароопасности (открытые, закрытые, пылевлагонепроницаемые, взрывоопасные, взрывозащищенные светильники).

1. Выбрать систему освещения. Наиболее экономичной является система комбинированного освещения, так как она создает наиболее равномерное светораспределение.

   При комбинированном освещении доля общего освещения в нем не должна быть меньше 10%.

2. Выбрать расположение светильников и определить их количество. Светильники, расположенные симметрично вдоль или поперек помещения, в шахматном порядке, рядами, ромбовидно, обеспечивают равномерное по площади освещение. Локализованное неравномерное размещение светильников производят с учетом местонахождения ПЭВМ, оборудования и т.д.

Наибольшая равномерность достигается:

- При шахматном расположении, если $\frac{r}{Hp} \le 1{,}7 \div 2{,}5$;
- При расположении прямоугольником, если $\frac{r}{Hp} \le 1{,}4 \div 2{,}0$,

где r – расстояние между светильниками; Hр – высота подвеса светильника над рабочей поверхностью: Hр = $H - h_{с} - h_{р.м.}$, где H – высота помещения, м; hс – высота подвеса светильника, м; hр.м. – высота рабочего места (hр.м. = 0,8 м), м.

Оптимальное расстояние от крайнего ряда светильников до стены: $r_к = (0{,}24 \div 0{,}3)r$.

При отсутствии рабочих поверхностей у стены: $r_к = (0{,}4 \div 0{,}5)r$.

Для исключения слепящего действия светильников общего освещения должно выполняться правило $H - h_c \le 2{,}5 \div 4$ м при мощности ламп Pл $\le$ 200 Вт. Необходимое число светильников при расположении квадратом составляет: $N_c = \frac{S}{r^2}$, где S – площадь помещения, м^2; r – длина стороны квадрата, м.

3. Определить нормируемую освещенность рабочего места по минимальному размеру объекта различия, фону, контрасту объекта с фоном в системе освещения.

Для расчета искусственного освещения используют три метода:

- Метод светового потока для общего равномерного освещения горизонтальной рабочей поверхности.
- Точечный метод для любой системы освещения.
- Метод удельной мощности для ориентировочных расчетов общего равномерного освещения.

Световой поток определяется по формуле:

$$Fл = \frac{Eн \cdot K \cdot S \cdot Z}{N \cdot \eta}, \quad (37)$$

где $F_л$ – световой поток лампы, лк; $E_н$ – нормированная освещенность, лк; S – площадь освещаемого помещения, м^2; K – коэффициент запаса (в соответствии со СНиП 23-05-95 для люминесцентных ламп производственных цехов предприятий службы быта $K = 1{,}6 \div 1{,}7$; для остальных помещений K = 1,5); Z – коэффициент минимальной освещенности, равный отношению средней освещенности к минимальной; N – число ламп; η – коэффициент использования светового потока, равный отношению потока, падающего на рабочую поверхность, к общему потоку ламп.

Коэффициент использования светового потока η зависит от к.п.д. светильника, коэффициента отражения потолка ($\rho_п$), стен ($\rho_с$), величины показателя помещения i, учитывающего геометрические параметры помещения, высоту подвеса светильника ($H_р$):

$$i = \frac{a \cdot b}{H_p (a + b)}, \quad (38)$$

где a и b – ширина и длина помещения, м.

При длине рабочего помещения a = 5,4 м, ширине b = 3,6 м и высоте H = 2,8 м, потребуется следующее освещение:

Eн = 400 лк

Fл = 5200 лм – световой поток 1 лампы

Тогда i = 2,3. Следовательно, η = 0,41.

Число ламп равно N = 6, что при использовании светильников с использованием одной лампы потребует использования 6 светильников.

Расстояние между двумя светильниками составит: r = 1,8 м.

Расстояние от стены до светильников: $r_к$ = 0,9 м.

Следовательно, светильники следует расположить в два ряда по три светильника (рисунок 35).

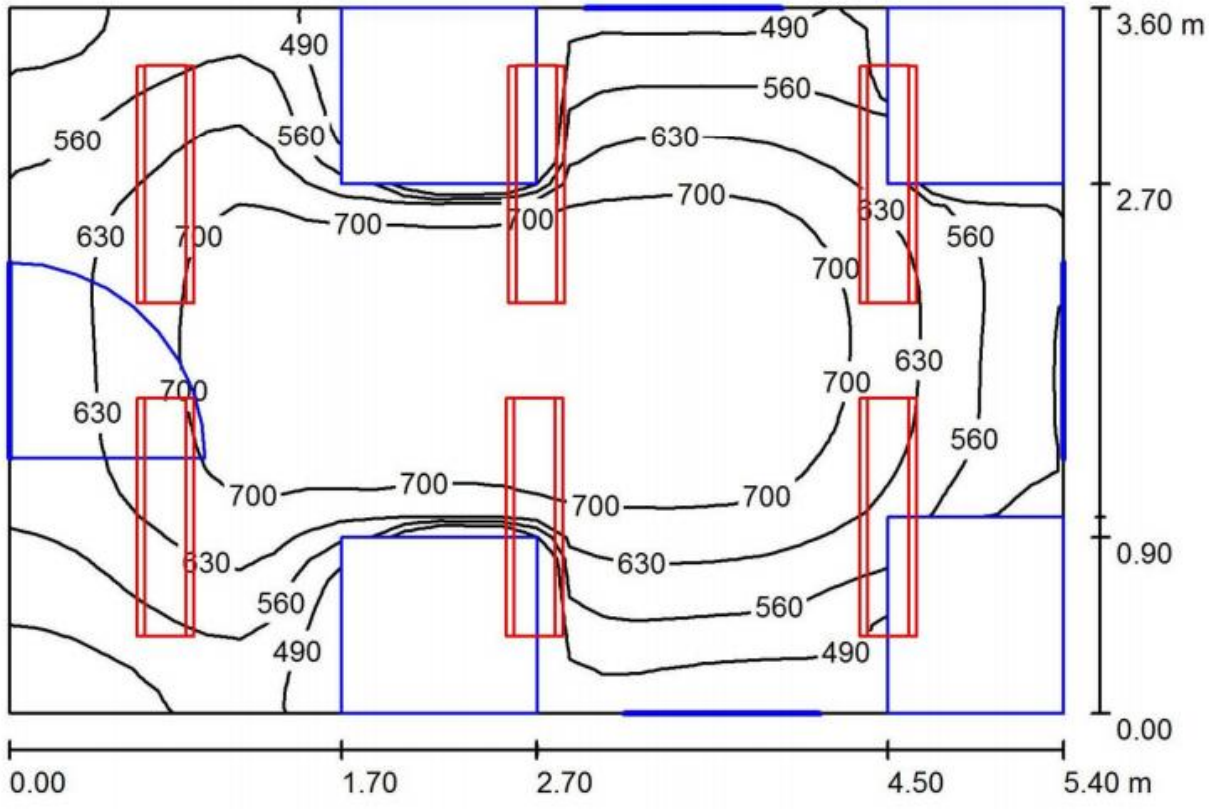


Рисунок 35 - схема освещения помещения (сделано в программе DIALUX).

Таким образом, в проекте используются 6 светильников с высотой подвеса 0,1 м и, соответственно, 6 люминесцентных ламп WILA MI1555-53 (таблица 19).

Удельная подсоединенная мощность: 19.753 Вт/м^2.

Таблица 30 - ведомость светильников.

№	Шт.	Обозначение (поправочный коэффициент)	Ф (светильник), лм	Ф (лампы), лм	Мощность, Вт
1	6	WILA MI1555-53 Micronic Deckenanbauleuchte, Mirkoprismengehäuse	3405	5200	64.0
Всего			20432	31200	384.0

Заключение

В данной работе было рассмотрено явление аномалий в статистических данных посещаемости сайта. Выявлена зависимость данных от времени и дня недели. Исходя из данной зависимости, можно строить оптимальные выборки, с точки зрения диапазона рассматриваемых значений. Также было получено, что наиболее эффективным критерием аномальности данных является скорость изменения данных.

Совместное применение алгоритмов LOF и 3G позволяет снизить количество лжеаномалий, порождаемых алгоритмом LOF, при этом уменьшается полнота работы данного алгоритма, но с учётом различных рассматриваемых типов данных, данная потеря компенсируется.

Хорошо откалиброванная связка данных алгоритмов может найти применение в любых сферах, где имеется потребность в поиске аномалий и имеется достаточно большой массив статистических данных для формирования выборки. Наибольшее применение возможно в системах реального времени, сохраняющих полученную информацию за длительный период работы. При использовании алгоритмов в других сферах также необходимо искать возможные зависимости данных.

Список использованных источников

1. Черняк, Л. / Большие Данные - новая теория и практика // Открытые системы. СУБД. — М.: Открытые системы, 2011. — № 10. — ISSN 1028-7493.

2. Социология: Энциклопедия / А. А. Грицанов [и др.]. — Мн.: Книжный Дом, 2003. — 1312 с. — (Мир энциклопедий)

3. Дюк В., Самойленко А. Data Mining: учебный курс (+CD). — СПб.: Изд. Питер, 2001. — 368 с.

4. Upton, Graham; Cook, Ian (1996). Understanding Statistics. Oxford University Press. p. 55. ISBN 0-19-914391-9.

5. Стивен Б. Акелис Боллинджера полосы (Bollinger Bands). — М.: Диаграмма, 1999. — С. 56—58. — 376 с. — ISBN 978-5-902537-13-7, 5-900082-05-09, ГРНТИ 06.73, ББК 65.526

6. Форекс учебник / Технический анализ: [Электронный ресурс]. – 7 с. URL: http://www.finforce.ru/ru/trader/forex-courses/book/technical/bollinger/ (дата обращения: 10.06.2014).

7. McLachlan, Geoffrey J. (1992); Discriminant Analysis and Statistical Pattern Recognition, Wiley Interscience, p. 12. ISBN 0-471-69115-1

8. Теория вероятностей: Учеб. для вузов. - 3-е изд., испр. / А.В. Печинкин, О.И. Тескин, Г.М. Цветкова и др.; Под ред. В.С. Зарубина, А.П. Крищенко. - М.: Изд-во МГТУ им. Н.Э.Баумана, 2004. - 456 с. (Сер. Математика в техническом университете; Вып. XVI).

9. Rorabacher, D.B. (1991) "Statistical Treatment for Rejection of Deviant Values: Critical Values of Dixon Q Parameter and Related Subrange Ratios at the 95 percent Confidence Level". Anal. Chem., 63 (2), 139–146. PDF (including larger tables of limit values)

10. Воронцов К.В. Алгоритмы кластеризации и многомерного шкалирования. Курс лекций. МГУ, 2007.

11. Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; Sander, J. (2000). "LOF: Identifying Density-based Local Outliers". Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. SIGMOD: 93–104. doi:10.1145/335191.335388. ISBN 1-58113-217-4.

12. Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; Sander, J. R. (1999). "OPTICS-OF: Identifying Local Outliers". Principles of Data Mining and Knowledge Discovery. Lecture Notes in Computer Science 1704. p. 262. doi:10.1007/978-3-540-48247-5_28. ISBN 978-3-540-66490-1.