Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ
Заведующий кафедрой СИИ
_____ Г. М. Цибульский
«____» _____ 2019 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Разработка модуля численной обработки данных больших объемов

09.04.02 Информационные системы и технологии

09.04.02.01 Информационно-управляющие системы

Руководитель	_____ подпись, дата	доцент, канд. техн. наук	О. А. Попова
Выпускник	_____ подпись, дата		В. Н. Юрьев
Рецензент	_____ подпись, дата	проф., д-р техн. наук	Л. А. Казаковцев
Нормоконтролер	_____ подпись, дата	доцент, канд. техн. наук	О. А. Попова

Красноярск 2019

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ
Заведующий кафедрой
_____ Г. М. Цибульский
подпись
«____» _____ 2019 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме магистерской диссертации

Студенту Юрьеву Виталию Николаевичу

Группа КИ17-02-1М, направление подготовки 09.04.02 «Информационные системы и технологии», профиль 09.04.02.01 «Информационно - управляющие системы».

Тема магистерской диссертации: «Разработка модуля численной обработки данных больших объемов».

Утверждена приказом по университету № _____ от _____г.

Руководитель ВКР Попова О. А. — доцент кафедры систем искусственного интеллекта ИКИТ СФУ, кандидат технических наук.

Перечень разделов ВКР:

- проблемный анализ темы исследования;
- анализ методов обработки и исследования больших временных рядов;
- описание программного модуля.

Перечень графического или иллюстративного материала с указанием основных чертежей, плакатов, слайдов: презентация «Разработка модуля численной обработки данных больших объемов».

Руководитель ВКР _____ О. А. Попова
подпись

Задание принял к исполнению _____ В. Н. Юрьев
подпись

« ___ » _____ 2019г.

График

выполнения выпускной квалификационной работы (ВКР) студентом направления 09.04.02 «Информационные системы и технологии», профиля 09.04.02.01 «Информационно-управляющие системы»

График выполнения выпускной квалификационной работы приведен в таблице 1.

Таблица 1 — График выполнения ВКР

Наименование этапа	Срок выполнения этапа	Результат выполнения этапа	Примечание руководителя (отметка о выполнении этапа)
Ознакомление с целью и задачами работы	15.09.18-30.09.18	Краткий обзор по теме магистерской диссертации	Выполнено
Сбор и подготовка источников	01.10.18-25.10.18	Список использованных источников (не менее 30)	Выполнено
Сбор данных и анализ методов обработки данных больших объемов	11.11.18-25.11.18	Обзор аналогов и сравнительная таблица	Выполнено
Формирование обзорной части	11.12.18-22.12.18	Обзорная часть магистерской диссертации	Выполнено
Исследование по подходам к разработке модулей обработки данных	10.02.19-01.03.19	Составление доклада	Выполнено
Исследование задач численной обработки больших данных	02.03.19-15.04.19	Составление доклада	Выполнено

Окончание таблицы 1.

Написать модельную задачу и провести тестирование с использованием методов ЧВА	25.04.19-15.05.19	Составление доклада	Выполнено
Решение экспериментальной части и практического примера	15.05.19-19.06.19	Формализация математической модели	Выполнено

Выпускник _____ В. Н. Юрьев
подпись

Руководитель _____ доцент, канд. техн. наук О. А. Попова
подпись

Реферат

Выпускная квалификационная работа в форме магистерской диссертации по теме «Разработка модуля численной обработки данных больших объемов» содержит 65 страниц текстового документа, 2 приложения, 40 источников.

БОЛЬШИЕ ДАННЫЕ, ИЗВЛЕЧЕНИЕ ЗНАНИЙ, ЧИСЛЕННО-ВЕРОЯТНОСТНЫЙ АНАЛИЗ, АГРЕГАЦИЯ, БОЛЬШИЕ ВРЕМЕННЫЕ РЯДЫ, ЧИСЛЕННОЕ МОДЕЛИРОВАНИЕ.

Цель работы — создание современного и эффективного метода численной обработки данных больших объемов.

Методы исследования – численно-вероятностный анализ, гистограммные временные ряды.

Полученные результаты – в ходе выполнения целей и задач исследования на тему численной обработки данных больших объемов рассмотрены существующие методы обработки больших временных рядов, обозначены имеющиеся проблемы и сформированы задачи исследования. На этапе моделирования применяется численный вероятностный анализ, который является новым направлением в вычислительной математике. На этапе постобработки результаты моделирования представляются в графическом виде. В результате осуществлена разработка модуля численной обработки большого временного ряда и проведено тестирование на основе практической задачи.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Магистерская диссертация посвящена актуальным вопросам численной обработки данных больших объемов. С развитием информационных технологий проблема обработки и исследования больших массивов данных становится все более и более актуальной. Существует ряд причин, которые существенно актуализируют разработку новых подходов и методов, связанных с содержательными количественными, структурными составляющими самих данных.

Данные с которыми приходится работать исследователю разнообразны по своему качественному и количественному составу, специфике способа измерения, регистрации, представления. В этом направлении важно отметить, что организации создают огромные массивы данных, которые представлены в различных образах и в различных форматах – это веб-журналы, видеозаписи, текстовые документы, машинный код или, например, геопространственные данные. В результате, пользователи, имея доступ к большому объему данных, часто нуждаются в эффективных технологиях обработки и анализа данных, с целью установить взаимосвязи между этими данными, извлечь полезную информацию и сделать на их основе выводы, имеющие большое значение.

С другой стороны, причина развития новых методов к обработке и анализу данных состоит в том, что с появлением новых информационных технологий актуальность хранения, обработки и анализа данных больших объемов с целью извлечения знаний становится важной задачей, требующей как больших вычислительных мощностей, так и новых методов анализа данных. Анализ публикаций на эту тему показал, что в последнее время повысился интерес, и увеличилась исследовательская активность в области разработки теории и практики анализа данных больших объемов.

Большой объем с одной стороны позволяет получить более точное описание объекта исследования, а с другой превращает поиск решений в сложную задачу, требующую применения современных математических методов обработки и анализа данных.

Целью исследования является разработка модуля численной обработки данных больших объемов на основе использования процедур агрегирования.

Для достижения поставленной цели в ходе исследования необходимо решить следующие основные задачи:

1) Анализ области исследования;

2) Анализ методов обработки и исследования больших временных рядов;

3) Разработка программного модуля.

1 Проблемный анализ темы исследования

1.1 Понятие больших данных

Одна из наиболее обсуждаемых тем в ИТ-изданиях в последнее время – феномен Big Data, или проблема Больших Данных. Стоит отметить, что проблемы хранения и обработки большого объема данных существовала всегда, но с развитием ИТ она стала беспокоить не только ряд крупнейших корпораций, но и гораздо более широкий круг компаний. Существует множество причин, которые послужили причиной появления Big Data.

Под термином Big Data, подразумеваются данные большого объема, технологии их обработки и хранения, проекты, рынок и даже компании, которые активно используют эту технологию.

Очевидно, что, данный термин связан с проблемой накопления и обработки очень больших массивов данных. За последние годы человечество произвело и сгенерировало информации больше, чем за всю историю своего существования.

Поток данных на сегодня действительно растет с огромной скоростью. Эти данные поступают с различных устройств. Огромные массивы данных генерируются научными учреждениями. Один только архив телескопа «Хаббл», накопленный за 15 лет, занимает около 25 Тбайт. В повседневной жизни человечество сталкивается растущим объемом данных – они поступают с различных объектов и устройств [1].

Оптимизация любых рабочих процессов требует наличия необходимой информации. В странах с развитой экономикой информатизация достигла невиданных масштабов. Нарастающие объемы данных для их использования при решении различных задач требуют некой структуризации, кластеризации, чтобы впоследствии пользователь мог бы получить нужную ему информацию. Что касается интернета и мобильных технологий, то потоки данных генерируются новыми интернет-сервисами, социальными сетями, приложениями электронной торговли, приложениями о местонахождении абонентов сетей и т. п.

В первую очередь, возросло число генераторов данных, причем весьма большого объема: социальные сети разных видов, данные электронной почты, Twitter, Wiki-проекты. Кроме того, огромные объемы данных могут генерироваться датчиками различных типов – Call Data Records (CDR) сотовых операторов, телеметрические данные, информация с камер видеонаблюдения и т.п. Во-вторых, значительное уменьшение стоимости хранения привело к тому, что многие компании могут позволить себе следовать парадигме «данные слишком ценны, чтобы их уничтожать».

При огромном объеме данных постоянно растущего числа веб-страниц это лишь вершина айсберга, большую же его часть составляют данные о данных. Интернет-маркетологам и аналитикам интересны не только содержание веб-сайтов, но и полная информация о пользователях: их привычки, история посещенных страниц, рекламные предпочтения, круг общения и знакомства. Именно такой подход позволяет продвигать рекламу и товары в Интернете. Можно лишь только предполагать, насколько подробно можно проанализировать поведение людей по истории посещений в Сети. Очевидно, что для этого нужно обрабатывать действительно гигантские объемы данных. По сути, технология Big Data не есть что-то принципиально новое. Просто на данном этапе развития технологий появились новые программные средства, обеспечивающие хранение, обработку и анализ таких объемов данных, что стало принципиально новым подходом к их анализу. Стоимость хранения информации при этом снизилась, что повлияло на возможность собирать всё больше данных и анализировать, не связанные друг с другом факторы. Человеческий мозг не может обнаружить такие закономерности, какие отмечает компьютер, выдавая совершенно неожиданные количественные взаимосвязи. Технология Big Data предоставляет огромный потенциал мегамассивов данных поиска ценных закономерностей и фактов путем объединения и анализа больших объемов данных. Необходимость обработки и хранения качественно новых объемов неструктурированных и структурированных данных показывает: традиционные подходы к их хранению и обработке стали неэффективными, а, следовательно,

необходимы новые технологии. При масштабности таких задач перед бизнесом встала задача не только выбора адекватного инструментария по анализу информации, но и построения оптимальной вычислительной инфраструктуры, которая была бы эффективной и не очень дорогой. Всё это подводит к более полному определению Big Data. Главным критерием отнесения ПО к технологии Big Data – это способность его обрабатывать большие массивы данных, поступающие из разных источников в различных форматах. При этом Big Data-приложения обеспечивают объединению данных из разных источников и разной степени структурированности, многие бизнес-задачи и научные эксперименты требуют совместной обработки данных различных форматов – это могут быть табличные данные в СУБД, иерархические данные, видео, изображения, текстовые документы, аудиофайлы и т.д.

Основные принципы работы с большими данными

Исходя из определения Big Data, можно сформулировать основные принципы работы с большими данными:

1. Горизонтальная масштабируемость. Так как данных может быть сколько угодно много – любая система, которая создана для обработки больших данных, должна быть расширяемой. Если в 2 раза вырос объём данных – необходимо в 2 раза увеличили количество железа в кластере и всё продолжит работать.

2. Отказоустойчивость. Принцип горизонтальной масштабируемости подразумевает, что машин в кластере может быть очень много. Например, Hadoop-кластер Yahoo имеет более 42000 машин. Это означает, что часть этих машин будет гарантированно выходить из строя. Методы работы с большими данными должны учитывать возможность сбоев и переживать их без каких-либо значимых последствий.

3. Локальность данных. В больших распределённых системах данные распределены по большому количеству машин. Если данные физически находятся на одном сервере, а обрабатываются на другом – расходы на передачу данных могут превысить расходы на саму обработку. Поэтому одним из

важнейших принципов проектирования BigData-решений является принцип локальности данных – по возможности обрабатываем данные на той же машине, на которой их храним.

Все современные средства работы с большими данными так или иначе следуют этим трём принципам. Для того, чтобы им следовать – необходимо придумывать какие-то методы, способы и парадигмы разработки средств разработки данных [2].

Наборы признаков больших данных

Набор признаков VVV (volume, velocity, variety) изначально выработан Meta Group в 2001 году вне контекста представлений о больших данных как об определённой серии информационно-технологических методов и инструментов, в нём, в связи с ростом популярности концепции центрального хранилища данных для организаций, отмечалась равнозначимость проблематик управления данными по всем трём аспектам. В дальнейшем появились интерпретации с «четырьмя V» (добавлялась veracity — достоверность, использовалась в рекламных материалах IBM, «пятью V» (в этом варианте прибавляли viability — жизнеспособность, и value — ценность), и даже «семью V» (кроме всего, добавляли также variability — переменчивость, и visualization). IDC интерпретирует «четвёртое V» как value с точки зрения важности экономической целесообразности обработки соответствующих объёмов в соответствующих условиях, что отражено также и в определении больших данных от IDC. Во всех случаях в этих признаках подчёркивается, что определяющей характеристикой для больших данных является не только их физический объём, но другие категории, существенные для представления о сложности задачи обработки и анализа данных [3].

Источники больших данных

Основными источниками больших данных признаются интернет вещей и социальные медиа, считается также, что большие данные могут происходить из внутренней информации предприятий и организаций (генерируемой в

информационных средах, но ранее не сохранявшейся и не анализировавшейся), из сфер медицины и биоинформатики, из астрономических наблюдений [4].

В качестве примеров источников возникновения больших данных приводятся непрерывно поступающие данные с измерительных устройств, события от радиочастотных идентификаторов, потоки сообщений из социальных сетей, метеорологические данные, данные дистанционного зондирования Земли(ДЗЗ), потоки данных о местонахождении абонентов сетей сотовой связи, устройств аудио- и видеорегистрации. Ожидается, что развитие и начало широкого использования этих источников даст старт проникновению технологий больших данных как в научно-исследовательскую деятельность, так и в коммерческий сектор и сферу государственного управления [5].

Извлечение знаний из данных больших объемов

Настоящее время стало очень важным этапом для проникновения новых информационных технологий и создаваемых на их основе высокопроизводительных компьютерных систем во все сферы человеческой деятельности - управление, производство, науку, образование и т.д. Создаваемые посредством этих технологий интеллектуальные компьютерные системы призваны усилить мыслительные способности человека, помочь ему находить эффективные решения так называемых плохо формализованных и слабоструктурированных задач, характеризующихся наличием различного типа неопределенностей и огромными поисковыми пространствами. Сложность таких задач состоит зачастую в необходимости их решения в очень ограниченных временных рамках, например, при управлении сложными техническими объектами в аномальных режимах или при оперативном разрешении конфликтных (кризисных) ситуаций. Наибольшей эффективности современные интеллектуальные системы достигают при реализации их как интегрируемых систем, объединяющих различные модели и методы представления и оперирования знаниями, а также механизмы приобретения (извлечения) знаний из различных источников [6].

Тема извлечения знаний привлекает внимание учёных как в Европе, так и во всём мире. Изучением данной темы занимаются У. Файяд, Г. Пятетский-Шапиро, Т. Гаврилова, Л. Григорьев, П. Смит, Дж. Сейферт, В. Фроли, Ц. Матеус, Е. Монк, Б. Вагнер, С.Хааг и др.

В связи с совершенствованием технологий записи и хранения данных на людей обрушились очень больше потоки информационных данных в самых различных областях. Деятельность любого предприятия (коммерческого, производственного, медицинского, научного и т.д.) теперь сопровождается регистрацией и записью всех подробностей его деятельности. В настоящее время стало понятным, что без продуктивной переработки потоки сырых данных образуют никому не нужную свалку.

В настоящее время с извлечением данных ассоциируется термин Data Mining. Data Mining значит «добыча» или «раскопка данных». Нередко рядом с Data Mining встречаются следующие слова: «обнаружение знаний в базах данных» и «интеллектуальный анализ данных». Их можно считать синонимами выражения Data Mining. Возникновение всех указанных терминов связано с новым этапом в развитии средств и методов обработки данных [7].

Основой современной технологии Data Mining является концепция шаблонов (паттернов), отражающих фрагменты многообразных взаимоотношений в данных. Эти шаблоны представляют собой закономерности, свойственные подвыборкам данных, которые могут быть компактно выражены в понятной человеку форме. Поиск шаблонов производится методами, не ограниченными рамками априорных предположений о структуре выборке и виде распределений значений анализируемых показателей.

Исторически сложилось, что у термина Data Mining есть несколько вариантов перевода:

- извлечение данных - это процесс нахождения и сбора информации, а также сохранения (конвертация) их в разных форматах;

- извлечение знаний, интеллектуальный анализ данных. Суть «извлечения знаний» состоит в следующем: у нас есть огромные массивы данных, нам необходимо получить знания. Рассмотрим пример из жизни: у нас есть очень много данных о котировках валют Forex (очень много — это порядка нескольких гигабайт текстовой информации в день). Так вот, текстовые файлы и есть данные, а вот утверждение «падение акции А приводит к падению акции В» уже является знанием, полученным на основании изучения этих данных [8].

Основные категории Data Mining:

- кластеризация данных (разделение объектов на подобные группы);

- классификация данных (отнесение объектов к заранее определенным группам);

- нейронные сети, генетические алгоритмы (универсальные оптимизаторы);

- ассоциативные правила (правила вида «если…, то...»);

- деревья решений;

- регрессионный анализ;

- анализ временных рядов.

1.2 Технологии обработки больших данных

RDBMS и NoSQL-системы

В некоторых задачах есть потребность очень высокой скорости обработки данных. Существует огромный класс задач, требующие принятия решения в реальном времени, например, обработка биометрических данных, получаемых в огромном потоке людей, которые необходимо сверить с базой данных о злоумышленниках. Для многих научных задач характерна очень большая скорость поступления данных.

Но главная проблема заключается в том, что кроме количества данных изменился и их характер. Основной объем этих данных – неструктурированная информация, поэтому ее хранение и обработка в привычных системах на основе реляционных БД, как правило, малоэффективна. И постепенно пришло

осознание того, что реляционные СУБД не являются оптимальным решением для ряда ситуаций, а это, в свою очередь, привело к появлению целого семейства решений, которые можно классифицировать одним словом – NoSQL-системы. Термин NoSQL расшифровывается как Not Only SQL (не только SQL). На сегодняшний момент существует большое количество таких систем, но все они, как правило, обладают следующими характеристиками:

- гибкость использования: у подобных систем отсутствуют требования к наличию схемы данных, а в качестве модели хранения выступает JSON1;

- встроенные возможности горизонтального масштабирования и параллельной обработки;

- возможность быстрого получения первых результатов.

Сравним типичный сценарий работы с данными в RDBMS (Relational Database Management System, реляционная СУБД) и в NoSQL-системе. В случае с RDBMS необходимо разработать схему хранения данных. Кроме того, перед загрузкой в СУБД данные должны быть очищены и преобразованы в требуемые форматы, только после этого ими можно будет воспользоваться через язык SQL-запросов. Таким образом, необходимо пройти как минимум шесть этапов (причем трансформация и загрузка данных могут быть весьма длительными и трудоемкими процессами), прежде чем появятся первые результаты (рис. 1).

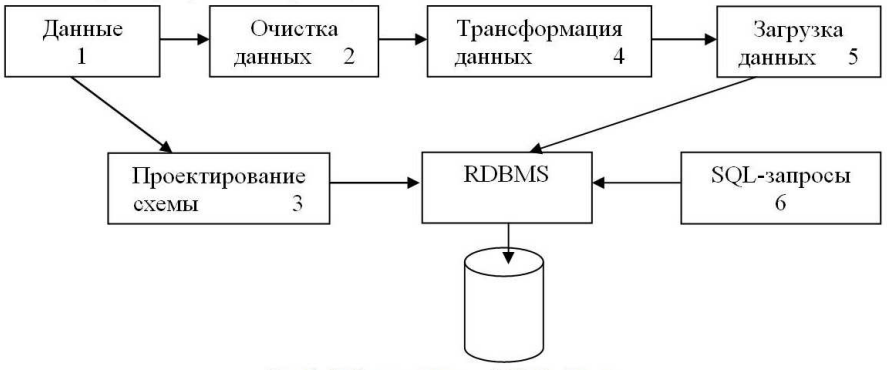

Рисунок 1 - Работа данных в rDBMS-системе

В случае с NoSQL ситуация выглядит значительно проще: после поступления данных в хранилище система уже готова к работе, конечно, при условии, что у вас есть готовая программа обработки (рис. 2).

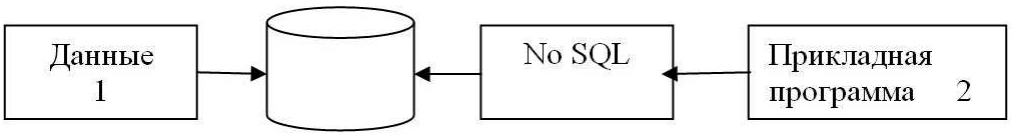

Рисунок 2 - Работа данных в NoSQL-системе.

Все NoSQL-системы при всем их многообразии можно разделить на два больших класса. Во-первых, это различные виды NoSQL Key / Value Database, или NoSQL базы данных. Типичными представителями этого класса систем являются такие проекты, как MongoDB, Cassandra или HBase. Все они представляют собой разновидность баз данных, хранящих информацию в виде пар «ключ –значение» и не имеющих жесткой схемы данных. Как правило, подобные продукты горизонтально масштабируются (объединяются в кластер из однотипных недорогих узлов) и имеют встроенные средства защиты от выхода из строя отдельных компонент кластера. Их удобно использовать в условиях постоянно изменяющейся (или вообще неопределенной) структуры данных.

Постепенно пришло осознание того, что реляционные СУБД не являются оптимальным решением для ряда ситуаций, а это привело к появлению семейства решений, которые можно классифицировать одним словом – NoSQL-системы.

Например, БД NoSQL часто используют для сбора и хранения информации в социальных сетях.

Поскольку приложения, с которыми работают пользователи, очень быстро меняются, структуру данных делают максимально простой: вместо того чтобы разрабатывать схему данных со связями между сущностями, создают структуры, содержащие основной ключ для идентификации данных и привязанное к нему содержимое. Такие оптимизированные и динамические структуры позволяют проводить изменения, не выполняя сложную и дорогую реорганизацию на уровне хранилища.

Вторым большим классом NoSQL-систем являются проекты, обеспечивающие горизонтально масштабируемую платформу для хранения и параллельной обработки данных. Они больше подходят для задач с «тяжелыми»

запросами, свойственными DWH (Data Warehouse) или бизнес-аналитике. Наиболее популярным и известным проектом является Apache Hadoop[8].

Map Reduce

MapReduce — это фреймворк для вычисления наборов распределенных задач с использованием очень большого количества компьютеров (называемых «нодами»), образующих кластер.

Для параллельных вычислений над большими наборами данных в компьютерных кластерах была разработана модель распределённых вычислений MapReduce. Рассмотрим подробнее данную модель на рисунке 3.

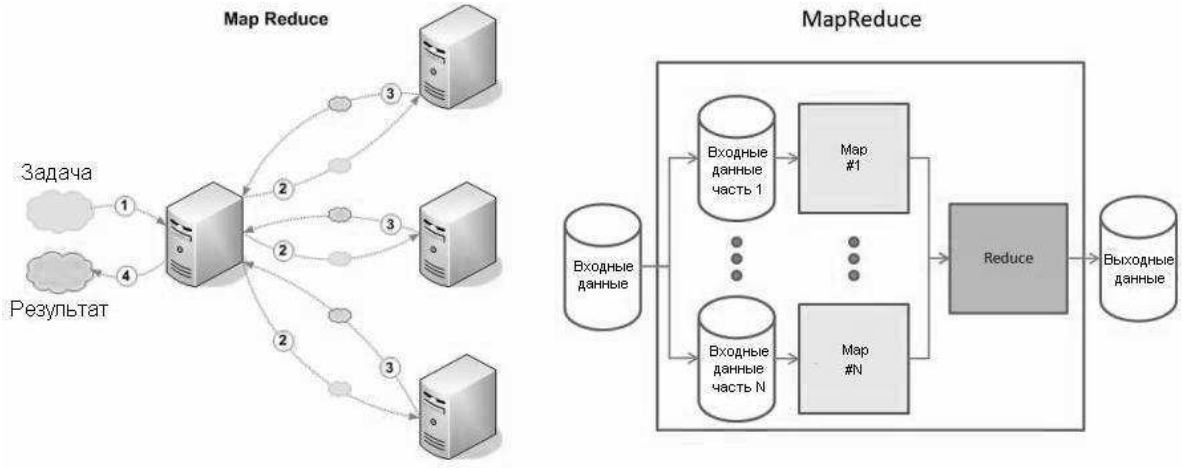

Рисунок 3 – Принцип работы модели MapReduce

На рисунке 4 представлена схема алгоритма работы MapReduce. Операция Map реализуется пользователем, и преобразует входную пару {ключ: значение} в набор промежуточных пар. Каждый узел кластера выполняет функцию Map на своей назначенной порции данных.

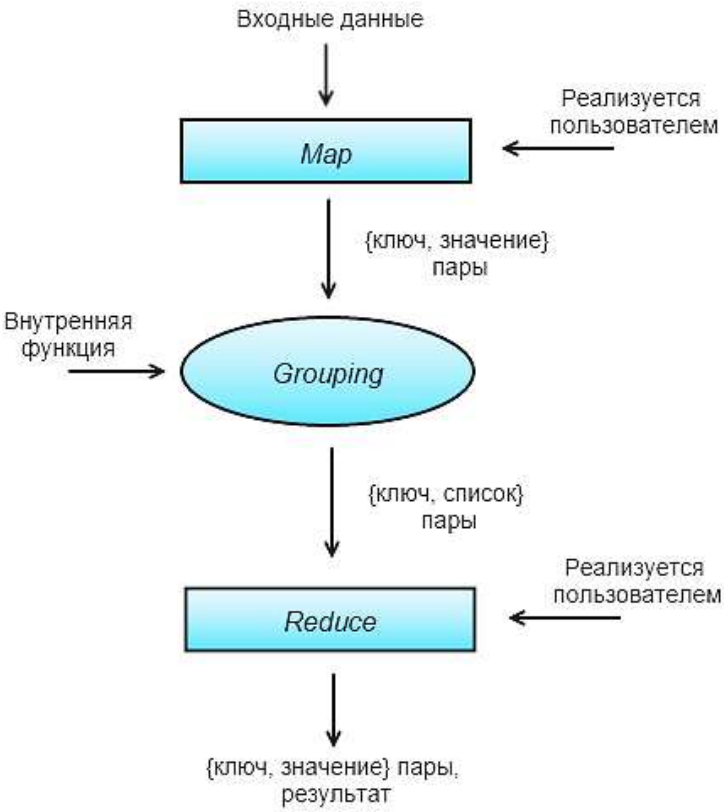

Рисунок 4 - Схема алгоритма работы MapReduce

Промежуточным этапом между Map и Reduce является этап группирования. Данная операция пользователем не задается. На этом шаге происходит объединение всех значений для одного и того же ключа и результатом является пара ключ и список значений [9].

Заключительный этап работы алгоритма выполняется функцией Reduce, также заданной пользователем. Функция проводит свертку и возвращает итоговый список значений. Таким образом формируется окончательный результат. Все данные операции являются независимыми друг от друга и легко распараллеливаются на кластерах из любого числа машин.

MapReduce позволяет решать множество задач, связанных с анализом и обработкой больших объемов данных, за приемлемое время благодаря высокому параллелизму. Кроме того, подход MapReduce устойчив к сбоям узлов и позволяет динамически распределять Map и Reduce-подзадачи по узлам кластера, принимая во внимание фактическое распределение данных по узлам кластера.

Основные принципы, заложенные в программный интерфейс Map/Reduce, можно сформулировать следующим образом:

- не данные передаются программе обработки, а программа – данным;
- данные всегда обрабатываются параллельно;
- параллельность обработки заложена архитектурно в программном интерфейсе Map/Reduce и не требует дополнительного кодирования от разработчика [10].

Hadoop

Hadoop состоит из двух основных компонент: распределенной кластерной системы Hadoop Distributed File System (HDFS) и программного интерфейса Map/Reduce. На основе HDFS и Map/Reduce разработан ряд продуктов.

Чтобы оценить достоинства, которыми обладает Hadoop, необходимо хорошо понимать принципы работы его компонентов. Рассмотрим архитектуру решения более подробно.

HDFS представляет собой распределенную, линейно-масштабируемую и устойчивую к отказам кластерную файловую систему. Фактически это кластер из множества однотипных узлов хранения данных с внутренними дисками, объединенных общей LAN. Кроме узлов хранения (Data Nodes), в кластере присутствует специальный узел, ответственный за управление и хранение метаинформации о HDFS (Name Node).

Запись в HDFS осуществляется блоками по 64 МБ. Такой большой размер обусловлен тем, что HDFS изначально спроектирована для хранения и обработки весьма значительного объема данных. Запись блоков данных по узлам кластера происходит равномерно, причем каждый блок имеет, как минимум, одну копию данных на другом узле. Таким образом, HDFS защищена от выхода из строя любого из узлов кластера (за исключением Name Node, который надо резервировать). Количество резервных копий блоков данных можно увеличивать, тем самым, добиваясь отказоустойчивости двух или более узлов кластера одновременно. При добавлении нового узла в HDFS-кластер

происходит автоматическое перераспределение блоков данных с учетом новой топологии, при этом емкость HDFS также увеличивается автоматически, не требуя каких-либо действий со стороны администратора [11].

Как следует из вышесказанного, HDFS сама по себе не представляет ничего революционного.

Самое интересное начинается при использовании связки HDFS и программного интерфейса Map/Reduce.

При разработке Map/Reduce предполагалось создать технологию, способную за ограниченное время обрабатывать огромные объемы данных.

Дело в том, что классическая парадигма серверов, имеющих совместный доступ к центральному хранилищу данных, перестает работать при их неограниченном объеме, поскольку вертикальное масштабирование серверов, массивов и каналов передачи данных ограничено. Была необходима система, масштабируемая горизонтально и не требующая передачи всех данных счетным узлам. Именно таким образом и спроектирован Map/Reduce [12].

В этой системе запрос на обработку данных представляет собой небольшую программу, написанную на языке Java, но фактически можно использовать любой язык программирования. Самое интересное, что программный интерфейс Map/Reduce самостоятельно передает и одновременно запускает эту программу на узлах кластера, хранящих обрабатываемые данные. HDFS равномерно распределяет данные по всем узлам, потому будут задействованы все доступные серверы кластера. Затем обработанные данные от всех узлов агрегируются методом Reduce и передаются пользователю [13].

Таким образом, вместо привычной концепции «база данных и сервер» здесь имеется кластер из множества недорогих узлов, каждый узел которого является и хранилищем, и обработчиком данных, а само понятие «база данных» отсутствует.

Подобные системы обладают двумя важными характеристиками. Во-первых, любой сколько угодно сложный анализ большого объема данных сводится к их обработке на локальных дисках сервера, поэтому максимально

возможное время реакции хорошо прогнозируемо. Во-вторых, система масштабируется симметрично и линейно – при добавлении новых узлов возрастают и вычислительная мощность, и дисковая емкость, поэтому время обработки данных не зависит от их объема.

Hadoop по сравнению с RDBMS (особенно с параллельными) имеет два недостатка. Первый очевиден: интерфейс Map/Reduce не совместим с SQL, поэтому приложения для работы с Hadoop придется переписывать. Второй проблемой является то, что, несмотря на все возможности параллельной обработки данных, по производительности Hadoop проигрывает существующим решениям на базе параллельных RDBMS. Этот вопрос долгое время был предметом жарких дискуссий между сторонниками и противниками RDBMS, пока группа известных специалистов по базам данных во главе с Майклом Стоунбрейкером и Дэвидом Девиттом не провела ряд испытаний, результаты которых были приведены в статье «A Comparison of Approaches to Large-Scale Data Analysis» [14].

Результаты тестов были довольно неутешительными для сторонников новых технологий. Система Hadoop сравнивалась с параллельными RDBMS Vertica и СУБД-X, которые по итогам продемонстрировали существенное превосходство в производительности над Hadoop при выполнении большого ряда тестовых задач анализа данных большого объема. В среднем для всех пяти задач на кластере из 100 узлов СУБД-X оказалась в 3,2 раза быстрее Hadoop, а Vertica – в 2,3 раза быстрее СУБД-X.

По мнению авторов тестов, преимущество в производительности, свойственное обеим системам баз данных, является результатом применения ряда технологий, разработанных в последние годы - это индексы в виде B-деревьев, ускоряющие выполнение операций выборки, новые механизмы хранения данных, эффективные методы сжатия данных с возможностью выполнять операции прямо над ними и сложные параллельные алгоритмы для выполнения запросов над большими объемами реляционных данных [15].

R (язык программирования)

Язык программирования R для статистической обработки данных и работы с графикой, а также свободная программная среда вычислений с открытым исходным кодом.

Главной особенностью языка программирования R является поддержка широкого спектр статистических и численных методов и наличием хорошей расширяемостью с помощью пакетов. Пакеты представляют собой библиотеки для работы специфических функций или специальных областей применения. Ещё одной особенностью R являются графические возможности, заключающиеся в возможности создания качественной графики, которая может включать математические символы [16].

1.3 Аппаратные решения

Существует ряд аппаратно-программных комплексов, предоставляющих предконфигурированные решения для обработки больших данных: Aster MapReduce appliance (корпорации Teradata), Oracle Big Data appliance, Greenplum appliance (корпорации EMC, на основе решений поглощённой компании Greenplum). Эти комплексы поставляются как готовые к установке в центры обработки данных телекоммуникационные шкафы, содержащие кластер серверов и содержащие управляющее программное обеспечение для массово-параллельной обработки [17].

Аппаратные решения для резидентных вычислений, прежде всего, для баз данных в оперативной памяти и аналитики в оперативной памяти, в частности, предлагаемой аппаратно-программными комплексами Hana (предконфигурированное аппаратно-программное решение компании SAP) и Exalytics (комплекс компании Oracle на основе реляционной системы Timesten (англ.) и многомерной Essbase), также иногда относят к решениям из области больших данных, несмотря на то, что такая обработка изначально не является массово-параллельной, а объёмы оперативной памяти одного узла ограничиваются несколькими терабайтами [18].

Кроме того, иногда к решениям для больших данных относят и аппаратно-программные комплексы на основе традиционных реляционных систем управления базами данных — Netezza, Teradata, Exadata, как способные эффективно обрабатывать терабайты и эксабайты структурированной информации, решая задачи быстрой поисковой и аналитической обработки огромных объёмов, структурированных данных. Отмечается, что первыми массово-параллельными аппаратно-программными решениями для обработки сверхбольших объёмов данных были машины компаний Britton Lee (англ.), впервые выпущенные в 1983 году, и Teradata (начали выпускаться в 1984 году, притом в 1990 году Teradata поглотила Britton Lee).

Аппаратные решения DAS — систем хранения данных, напрямую присоединённых к узлам — в условиях независимости узлов обработки в SN-архитектуре также иногда относят к технологиям больших данных. Именно с появлением концепции больших данных связывают всплеск интереса к DAS-решениям в начале 2010-х годов, после вытеснения их в 2000-е годы сетевыми решениями классов NAS и SAN [19].

1.4 Задачи, связанные с большими данными

Существуют три типа задач, связанных с Big Data:

1. Хранение и управление - объем данных в сотни терабайт или петабайт не позволяет легко хранить и управлять ими с помощью традиционных реляционных баз данных.

2. Неструктурированная информация - большинство всех данных Big Data являются неструктурированными.

3. Анализ Big Data - анализ информации [20].

1.5 Задачи обработки временных рядов

В настоящее время для изучения свойств сложных систем, в том числе и при экспериментальных исследованиях, широко используется подход, основанный на анализе сигналов, производимых системой. Это становится актуальным тогда, когда математически описать изучаемый процесс

практически невозможно, но в нашем распоряжении имеется некоторая характерная наблюдаемая величина. Поэтому анализ систем, особенно при экспериментальных исследованиях, часто реализуется с помощью обработки регистрируемых сигналов. Например, в аритмологии в качестве такого сигнала используется электрокардиограмма, в сейсмологии — запись колебаний земной коры, в метеорологии — данные метеонаблюдений и т.п.

Обычно такой сигнал называется наблюдаемой, а метод исследования — реконструкцией динамических систем. Этот раздел теории динамических систем называется анализом временных рядов [21].

Наблюдаемая — это последовательность значений некоторой переменной (или переменных), регистрируемых непрерывно или через некоторые промежутки времени. Очень часто вместо термина наблюдаемая используется понятие временной ряд. Понятно, что наши знания об изучаемой системе сильно ограничены наличием только лишь временного ряда вместо полного решения уравнений. Это накладывает большие ограничения на возможности метода реконструкции.

Скалярным временным рядом $\{x_i\}_{i=1}^{N}$ i=1 называется массив из N чисел, представляющих собой значения некоторой измеряемой (наблюдаемой) динамической переменной $x(t)$ с некоторым постоянным шагом τ по времени, $t_i = t_0 + (i-1)\tau$: $x_i = x(t_i)$, $i = 1, \ldots, N$. В анализе временных рядов выделяются две основные задачи: задача прогноза и задача идентификации. Задача идентификации при анализе наблюдаемых предполагает ответ на вопрос, каковы параметры системы, породившей данный временной ряд — размерность вложения, корреляционная размерность, энтропия и др.

Размерность вложения — это минимальное число динамических переменных, однозначно описывающих наблюдаемый процесс. Корреляционная размерность непосредственно является оценкой фрактальной размерности аттрактора системы и частным случаем обобщенной вероятностной размерности. Понятие энтропии тесно связано с предсказуемостью значений ряда и всей системы.

Задача прогноза имеет целью по данным наблюдений предсказать будущие значения измеряемых характеристик изучаемого объекта, т.е. составить прогноз на некоторый отрезок времени вперед. Сейчас разработано и обосновано несколько различных методов прогноза, все они подразделяются на два основных класса: локальные и глобальные [22].

Такое деление проводится по области определения параметров аппроксимирующей функции, рекуррентно устанавливающей следующее значение временного ряда по нескольким предыдущим.

Исторически первыми были разработаны глобальные методы, в которых на основе статистического анализа предлагалось использовать авторегрессию, скользящее среднее и др. Позже в рамках нелинейной динамики, были разработаны новые практические методики:

- сингулярный спектральный анализ (SSA), который является глобальным методом;
- локальная аппроксимация (LA);
- сочетание SSA–LA.

Исследование временных рядов базируется на идее, что удовлетворительную геометрическую картину странного аттрактора можно получить, если вместо переменных, входящих в исходную систему, использовать так называемые векторы задержек наблюдаемой величины $z_i = \{x_i, x_{i+1}, ..., x_{i+m-1}\}$.

Обоснован данный подход к анализу временных рядов был впервые в работе Ф.Такенса [23].

Поэтому, наиболее интригующим и заманчивым приложением теории динамических систем является прогнозирование динамики порождаемых ими временных рядов. При этом предполагается, что *a priori* характеристики систем, которые порождают этот ряд, могут быть неизвестны.

Сейчас стало ясно, что теория игр теснейшим образом переплетена с теорией динамических систем, фрактальных множеств и нелинейной динамикой,

поскольку большинство реальных временных рядов имеют самоподобную структуру. Эта особенность позволяет переосмыслить подходы к анализу временных рядов и иным (в основном, более успешным образом) подойти к их описанию. При этом выявляются различные стратегии прогноза, обосновывается невозможность использовать здравый (привычный) смысл в некоторых, казалось бы, очевидных ситуациях и т.п. Более того, если принять во внимание теорию управления хаотическими системами, то становится возможным на основе совершенно иных подходов, чем это принято в обычной теории, управлять динамической системой.

Таким образом, теоретические исследования, основанные на анализе временных рядов, могут дать мощный инструмент для понимания многих явлений, особенно когда имеющихся данных для построения модели может быть недостаточно [24].

2 Анализ методов обработки и исследования больших временных рядов

2.1 Понятие больших временных рядов и их модели

Временные ряды — это особый способ представления данных, характеризующих изменение некоторого показателя (показателей) во времени.

В экономике это ежедневные цены на акции, курсы валют, еженедельные и месячные объемы продаж, годовые объемы производства и т. п. В метеорологии типичными временными рядами являются ежедневная температура, месячные объемы осадков, в гидрологии периодически измеряемые уровни воды в реках. В технике временные ряды возникают при измерении значений приборов и параметров технологических процессов в последовательные моменты времени.

Многообразие систем и процессов, протекающих в них с течением времени, определяет различные виды временных рядов, а также методы и подходы к их исследованию.

Как правило, временной ряд - это последовательность чисел; его элементы значения некоторого процесса в определенные моменты времени t, обычно через равные промежутки; элементы временного ряда x нумеруют в соответствии с номером момента времени, к которому они относятся. Порядок следования элементов временного ряда весьма существен [25].

Понятие временного ряда часто толкуют расширительно. Например, одновременно могут регистрироваться несколько характеристик процесса. В этом случае говорят о многомерных временных рядах. Если измерения производятся непрерывно, говорят о временных рядах с непрерывным временем, или о случайных процессах. Наконец, текущая переменная может иметь не временной, а какой-нибудь иной характер, например, пространственный (тогда говорят о случайных полях).

Особенностью измерения элементов временных рядов x является присутствие случайных помех, случайных ошибок и т. д.

Временные ряды бывают двух типов: моментные и интервальные. Моментный временной ряд получается при многократном измерении некоторой величины через равные промежутки времени. Примерами моментных рядов являются ежегодно фиксируемая численность населения; курс акции на момент окончания торгов для каждого торгового дня последнего месяца; температура воздуха, вычисляемая ежедневно в полдень в течение года. Интервальные же ряды связаны с накоплением (суммой) величины за равные промежутки времени.

Примерами интервальных рядов могут служить ежедневные объемы продаж товара, ежемесячные объемы перевозок или количество осадков.

В последнее время термином ¾интервальные временные ряды стали обозначать временные ряды со значениями в виде интервалов.

Для изучения временных рядов используются различные методы и подходы, которые можно с назвать анализом временных рядов. Анализ временных рядов, в свою очередь, делится на ряд анализов, отражающих специфику как исследуемого объекта, процесса, так и способы измерения, представления данных в виде временных рядов [26].

Существует две главные цели анализа временных рядов: определение структуры ряда и прогнозирование (по настоящим и прошлым значениям предсказание будущих значений временного ряда). Для реализации этих целей необходимо, чтобы модель временного ряда была идентифицирована и описана. Когда модель будет описана, используя её можно проанализировать рассматриваемые данные (например, использовать в теории для понимания сезонного изменения осадков, если занимаются прогнозом погоды). Не обращая сильно внимания на глубину понимания теории, вы можете восстановить ряд на основе найденной модели и предугадать его будущие значения.

Временной ряд изучается с различными целями:

- управление процессом, создающим временной ряд – большие требования к математической модели, которая должна описывать, в том числе, влияние управления на временной ряд;

- предсказание будущих показаний временного ряда на основе предыдущих – это требования к математической модели определяются требованиями к точности предсказания;

- подбор статистической модели, описывающей временной ряд – Качество математических моделей в этом случае принято оценивать по количеству независимых параметров, использованных в них. Все специалисты знают, что, увеличивая размерность пространства параметров, ограниченный объем исходных данных можно подогнать под любую модель;

- описание характерных особенностей ряда – можно подумать, что математические методы в этом случае н нужны. На самом деле на этом уровне применяются очень тонкие методы, например, проверка гипотезы случайности и др. Очень часто на этом уровне оказываются излишне амбициозные исследователи после последовательного спуска с трех предыдущих уровней, чтобы найти ту самую закономерность, которая позволит им улучшить прогноз.

Иногда необходимо получить описание уникальных особенностей временного ряда, а иногда необходимо не только спрогнозировать значения временного ряда, но также и управлять поведением временного ряда. Исходя из целей анализа определяется метод анализа.

Спектральный анализ

Спектральный анализ является одним из самых мощных инструментов обработки эксперимента. В частности, он используется для анализа данных, выявление характерных частот в целях подавления шума и т.д.

Спектром данных y(x) называют некоторую функцию другой координаты (или координат, если речь идет о многомерном спектре) полученную в соответствии с определенным алгоритмом. Примерами спектров являются преобразование Фурье, спектр мощности, вейвлет-преобразование.

Спектральный анализ позволяет находить периодические составляющие временного ряда.

Корреляционный анализ

Корреляционный анализ позволяет выявить существенные свойства временных рядов. В том числе периодические зависимости и временные лаги для единичного процесса (автокорреляция) или между несколькими процессами (кросскорелляция).

Чем больше информации относительно величины *Y* содержится в исходных *x1, x2,x3* … тем более тесную связь мы можем выявить между ними.

Установив характер взаимосвязи можно получить ожидаемое значение зависимой переменной при заданных значениях объясняющих переменных, то есть построить эконометрическу модель.

Добившись разбиения зависимости переменной на случайную и объясненную, мы можем построить тренд.

Сам анализ состоит из нахождения взаимосвязей между значениями *X(t),* нахождения тренда, между отклонением значений от линии тренда. Вычитая линию тренда из функции *X(t)* мы получим некоторые остатки, анализ этих остатков позволяет выявить существование периодичности и тенденции к смене тренда [27].

Модели авторегрессии и скользящего среднего

Модели ориентированы на описание процессов, проявляющих однородные колебания, возбуждаемые случайными воздействиями. Позволяют предсказать будущие значения ряда.

Многоканальные модели авторегрессии и скользящего среднего

Модели применяются тех случаях, когда имеется несколько коррелированных между собой временных рядов. В них имеются колебания, возбуждаемые одной причиной. Позволяют предсказывать будущие значения ряда.

Модель авторегрессии и скользящего среднего. Общая модель, предложенная Боксом и Дженкинсом (1976) включает как параметры авторегрессии, так и параметры скользящего среднего. Имеется три типа параметров модели: параметры авторегрессии (p), порядок разности (d),

параметры скользящего среднего (q). В обозначениях Бокса и Дженкина модель записывается как АРПСС (p, d, q). Например, модель (0, 1, 2) содержит 0 (нуль) параметров авторегрессии (p) и 2 параметра скользящего среднего (q), которые вычисляются для ряда после взятия разности с лагом 1.

Гистограммный временной ряд

Гистограммный временной ряд (ГВР) описывает ситуации, когда в течение каждого момента времени известны гистограммы, аппроксимирующие функции плотности некоторых случайных величин. Подобные ситуации возникают, когда необходима агрегация большого числа данных в некоторые моменты времени. Во многих случаях гистограммы более информативны, чем, например, среднее значение. Области, где ГВР полезны, включают экономику, мониторинг окружающей среды.

Обычно временной ряд - это значения во времени каких-либо параметров (в простейшем случае одного) исследуемого процесса. Однако подобные ряды не описывают явления, когда реализация наблюдаемой величины доступна для каждого момента времени в виде некоторого множества.

Вот две типичных ситуации, когда это происходит:

1. Если измеряется некая переменная во времени для группы людей. Но исследование заключается не в каждом отдельном человеке, но в группе в целом. В этом случае временной ряд представляет выборочное среднее наблюдаемой величины с момента времени.

2. Когда переменная наблюдается, например, раз в секунду или в минуту, но должна быть проанализирована на более низкой частоте, скажем за день. В этом случае среднее значение и интервальный анализ много информации не учитывают.

Эти две ситуации описывают распределенная и временная агрегации соответственно. В каждом случае временной ряд функций плотности вероятности предложил бы более информативное представление, чем другие его формы.

Таким образом, чтобы использовать временные ряды функций плотностей вероятности, нужно определить, как представлять наблюдаемые распределения. Распределения могут быть оценены любым параметрическим методом или непараметрическим методом.

Далее рассмотрено представление функций плотности вероятности с использованием гистограммы.

Такие ряды, конечно, возникают во многих приложениях, включая экономику, финансы, метеорологию и так далее [28]. На рис. 5 приведен пример использования гистограммных временных рядов.

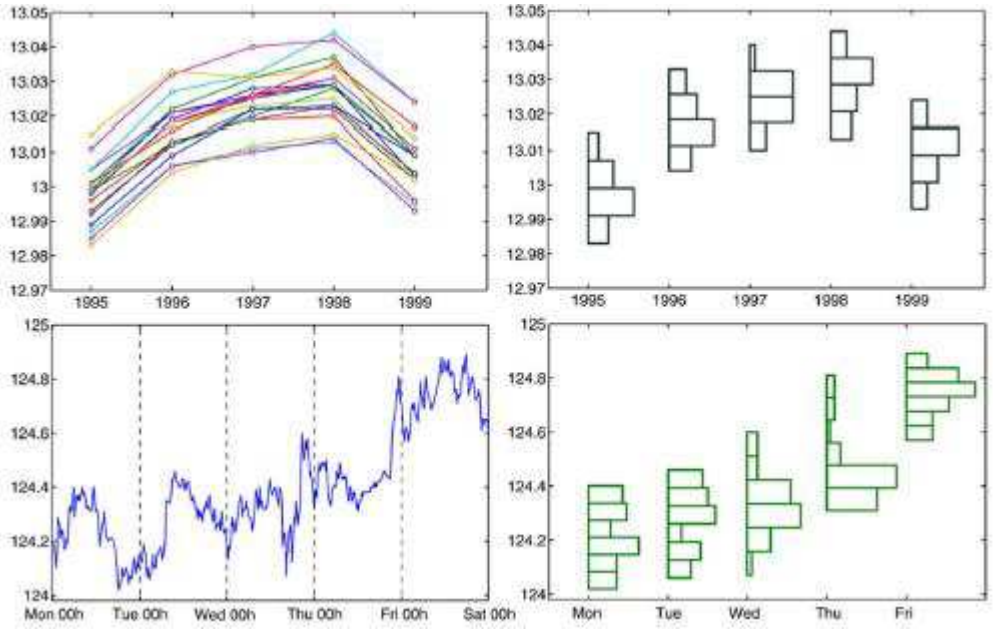

Рисунок 5 - Примеры агрегации при построении ГВР

В символическом анализе данных и Data Mining [29] гистограммы применяются для описания изменчивости количественных признаков и используются для исследования множества различных процессов.

Использование гистограмм во временных рядах функций плотности вероятности влечет за собой порождение ГВР. В анализе ГВР главной целью является предсказанием временного ряда распределений, наблюдаемых в течение времени и, следовательно, прогноз будет функций плотности вероятности, представленный гистограммой.

Гистограммный временной ряд подходит для представления агрегированных данных. Очевидно, что, если интерес заключается в исходных

данных, агрегирование не должно быть рассмотрено. Заметим, что ГВР сохраняют больше информации, чем среднее или интервал.

Для прогнозирования ГВР в ряде работ использовались адаптации известных методов. В работе [30] предложены методы сглаживания, основанные на гистограммной арифметике. В работе [31] адаптирован алгоритм k−NN для прогноза ГВР. Способность к прогнозу и простота k−NN делают свою адаптацию к ГВР подходящей. Другая сила алгоритма k –NN своя многосторонность: это может быть применено к оценке плотности, классификации, приближению функции и также к прогнозированию временного ряда.

Использование гистограмм обусловлено прежде всего тем, что они позволяют достаточно точно представлять произвольные распределения. Вторая причина развитая арифметика для работы с гистограммными переменными.

Во-первых, это важно отметить, что, несмотря на свою простоту, гистограммы охватывают все возможные интервалы оценки плотности вероятности. Наиболее популярными из них являются гистограммы с фиксированной шириной столбцов, которые в большом объеме используются на практике, дискретные плотности вероятности и полиграммы.

Во-вторых, гистограммы могут также обеспечить точное представление исходной плотности вероятности. Можно утверждать, что на основе ядерных оценок плотности вероятности можно представить более гладкие, чем гистограммы, аппроксимации плотности вероятности. Однако это значительно повышает вычислительные затраты [32].

Причины для использования гистограмм могут быть сформулированы следующим образом:

- можно использовать их для любой исходной плотности вероятности;
- гистограммы могут описывать данные с достаточной степенью точности;
- их использование упрощает простая и гибкая структура.

2.2 Методы агрегации больших временных рядов

Гистограмма

В тех случаях, когда желательно по данным эксперимента построить оценку плотности вероятностей, обращаются к построению гистограммы. Процедура ее построения состоит из следующих шагов.

В области возможных значений измеряемой величины X строится сетка $\omega = \{x_i | i = 1,2, \dots, n\}$.

Определяется, сколько выборочных значений m_i от общего числа N оказалось в каждом интервале $(x_{i-1}, x_i]$.

Над каждым из интервалов строится вертикальный прямоугольник с площадью m_i/N. Высота прямоугольника $P_i = m_i/(N(x_i - x_{i-1}))$. (2.1)

Полученная совокупность прямоугольников — гистограмма. Гистограмма — кусочно-постоянная функция, которая определяется своей сеткой ω, значениями $\{P_i\}$, принимающая на каждом интервале $(x_{i-1}, x_i]$ постоянное значение P_i [22].

Основанием для использования гистограммы $P_h(x)$ в качестве оценки неизвестной плотности вероятностей $P(x)$является кусочно-интегральная сходимость, $P_h(x)$ к $P(x)$ которая следует из того, что относительная частота $\frac{m_i}{N}$ события $X \in (x_{i-1}, x_i]$ сходится к его вероятности P_i

$$\frac{m_i}{N} \to \int_{x_{i-1}}^{x_i} p(x)dx. \tag{2.2}$$

Такой сходимости в ряде практических случаев оказывается достаточно; однако всегда желательно по возможности улучшить оценку при заданных ограничениях. Несколько факторов влияет на качество гистограммы: объем выборки N, величина интервалов группировки. Все осложняется еще и тем, что степень влияния этих факторов зависит от неизвестного экспериментатору до опыта истинного распределения вероятностей $P(x)$. Поэтому на практике гистограммы строят с некоторым учетом свойств полученной выборки.

Например, величина интервала группировки выбирается так, чтобы не сгладить существенные особенности распределения; объем выборки связывают с тем, чтобы в ячейке с наименьшим числом измерений их насчитывалось не менее пяти; размещение интервалов связывают с положением наименьшего и наибольшего выборочных значений. Отметим также, что на качество гистограммы влияет и точность измерений [33].

Теоретическая задача оптимизации гистограммы может быть сформулирована в нескольких вариантах, однако ее решение связано с трудностями, так что конкретных результатов можно добиться лишь при некоторых частных предположениях.

Применение гистограмм для исследования неопределенности рассматривается в работе О. А. Поповой: «О новых подходах к представлению неопределенности в данных для крупномасштабных систем» в которой описывается, что представление неопределенности, содержащейся в параметрах входных данных, осуществляется с использованием гистограмм второго порядка.

На основе численного вероятностного анализа [34] предлагается концептуально-гистограммный подход, который применяется для разработки процедур представления и обработки информационных потоков, а также для численного моделирования и представления характеристик экономических показателей. Показывается, что применение разработанных процедур позволяет агрегировать данные, снижает уровень информационной неопределенности в данных и существенно повышает эффективность численных расчетов.

Рассматривая гистограмму как математический объект, который определяется как кусочно-постоянная функция P, определенная сеткой $\{z_i, i = 0, 1, \dots, \}$, и на отрезке $[z_{i-1}, z_i]$ принимающая постоянное значение p_i, авторы определяют арифметические операции над гистограммами, операции вычисления максимума и минимума, возведения в степень [33-35], тем самым предлагая гистограммную арифметику над объектами, использующими как форму представления и исследования гистограмму.

Реализация арифметических операций над гистограммами основана на работе *p(x,y)* — совместной плотностью вероятности двух случайных величин *x,y*. Пусть p_z — гистограмма, приближающаяся плотность вероятности арифметической операции над двумя случайными величинами $x * y$, где $* \in \{+, -, *, /, \uparrow\}$. Тогда вероятность попадания величины z в интервал $[z_{i-1}, z_i]$ определяется по формуле [35]:

$$P(z_k < z < z_{k+1}) = \int_{\Omega_k} p(x,y)dxdy,$$

где, $\Omega_k = \{(x,y) | \, z_k \leq x * y \leq z_{k+1}\}$. (2.3)

Разберем на примере операции сложения основные принципы разработки гистограммных операций. Пусть $z = x_1 + x_2$, и носители x_1 — $[a_1 + a_2]$, x_2 — $[b_1 + b_2]$, $p\,(x_1, x_2)$ — плотность распределения вероятностей случайного вектора (x_1, x_2). Заметим, что прямоугольник $[a_1, a_2] \times [b_1, b_2]$ — носитель плотности распределения вероятностей $p\,(x_1, x_2)$, и плотность вероятности z отлична от нуля на интервале $[a_1 + b_1, a_2 + b_2]$. Обозначим $z_i, i = 0,1, \dots, n$ —точки деления этого интервала на n отрезков. Тогда вероятность попадания величины z в интервал $[z_{i-1}, z_i]$ определяется по формуле:

$$P(z_i < z < z_{i+1}) = \int_{\Omega_i} p(x_1, x_2)dx_1dx_2, \quad (2.4)$$

где $\Omega_k = \{(x_1, y_2) | \, z_i \leq x_1 * y_1 \leq z_{i+1}\}$[35]. И окончательно p_{zi} имеет вид

$$p_{zi} = \int_{\Omega_i} p(x_1, x_2)dx_1dx_2 \,/(z_{i+1} - z_i). \quad (2.5)$$

Рассмотренный выше подход обобщается на случай большого числа переменных. Пусть требуется найти гистограмму p_z суммы

$$z = a_1x_1 + a_2x_2 + \cdots + a_nx_n \quad (2.6)$$

и пусть $p(x_1, x_2, \dots, x_n)$ — плотность распределения вероятностей случайного вектора $(x_1, x_2, \dots, x_n)$. Тогда вероятность попадания z интервал (z_{i-1}, z_i) соответственно равна [35]

$$P(z_i < z < z_{i+1}) = \int_{\Omega_i} p(x_1, x_2 \dots, x_n)dx_1dx_2 \dots dx_n, \quad (2.7)$$

где $\Omega_i = \{(x_1, x_2 \dots, x_n) | \, z_i < a_1x_1 + a_2x_2 + \cdots + a_nx_n < z_{i+1}\}$, p_{zi} имеет вид

$$p_{zi} = \int_{\Omega_i} p(x_1, x_2 \dots, x_n) dx_1 dx_2 \dots dx_n / (z_{i+1} - z_i). \quad (2.8)$$

Частотный полигон

Частотный полигон — это непрерывная оценка плотности на основе гистограммы, которая представляет собой кусочно-линейную функцию [36].

Теоретические свойства одномерных и двумерных частотных полигонов показывают, что у них есть удивительные улучшения по сравнению с гистограммами [37]. Фишер не одобряет частотный полигон, по иронии судьбы, из-за причин графического отображения:

Преимуществом является наглядность, что не только форма кривой, указанной подобным образом несколько вводит в заблуждение, но с особой тщательностью всегда следует отличать бесконечно большое гипотетическое множество, из которого отображается наша выборка наблюдений, из фактических результатов наблюдений, которыми мы обладаем.

Фишер не знал о каких-либо теоретических различиях между гистограммами и частотными полигонами и думал только об одномерных гистограммах, когда писал этот раздел [37]. Его возражение использовать непрерывную непараметрическую оценку плотности больше не обосновано, но его заботу об использовании методов, которые полностью затеняют статистический шум с математической сложностью стоит подчеркнуть. Наконец, в качестве вопроса терминологии, различие между гистограммой и частотным полигоном в научной литературе стирается, гистограммная метка применяется для обоих случаев.

Одномерный частотный полигон является линейным интерполянтом середин равных промежутков областей гистограммы. Как таковой, частотный полигон выходит за пределы гистограммы в пустую область с каждого края. Частотный полигон легко проверить с помощью истиной функции плотности, то есть неотрицательная с интегралом, равным 1.

Применение частотного полигона как метода обработки и представления дистанционного мониторинга для агрегации больших объемов данных рассматривается в статье «Численный вероятностный подход к обработке и

представлению данных дистанционного мониторинга» Б. С. Добронец, О. А. Попова [38]. На основе агрегированных данных для выявления взаимосвязей между входными и выходными характеристиками изучаются теоретические и практические аспекты регрессионного моделирования. На основе численных примеров демонстрируются эффективность и надежность предлагаемых методов.

Помимо гистограмм и частотных полигонов в качестве математических моделей агрегатов можно использовать сплайны, как кусочно-полиномиальные функции. Сплайн представляет собой достаточно гладкую кусочно-полиномиальную функцию.

Сплайн-подход к агрегированию данных

Этот подход полезен по следующим причинам. Поскольку сплайн является кусочно- полиномиальной функцией, то его можно рассматривать как функцию агрегирования данных. Функция агрегирования выполняет численные обработку наборов данных и возвращает сплайн-значения. Сплайны полезны для анализа неопределённости в данных из-за того, что они адекватно представляют частотное распределение данных.

Предположим, что нам известна выборка $\Xi = (\varepsilon_1, \varepsilon_{2,\dots}\varepsilon_N)$случайной величины ε с функцией плотности вероятности $f(x)$и носителем $[a, b]$. В работе рассматривается новый подход к восстановлению функция плотности вероятности на основе эмпирических данных. Подход основан на аппроксимации функции плотности вероятности в некоторой точке с использованием прямоугольных ядер переменной ширины h. Важно отметить, что подход имеет определённое сходство с гистограммами, в частности, с методом усреднения гистограмм. С другой стороны, структура и точность построенных оценок соответствуют ядерному методу. Таким образом, используя данные Ξ можно получить оценку $\hat{f}(z)$ функции плотности *f(*z) , такую что

$$| f(z) - \hat{f}(z) | \approx h^4, \forall z \in [a,b], \tag{2.9}$$

где h — параметр сглаживания.

Рассмотрим вопрос построения сплайна s, аппроксимирующего функцию плотности $f(x)$, так чтобы выполнялась оценка

$$\| f - s \| \le Ch^4. \tag{2.10}$$

Для этих целей построим в области $[a,b]$ сетки

$$\omega_z = \{z_i = a + ih_z, i = 0,\ldots,N_z\}, \omega_x = \{x_i = a + ih_x, i = 0,\ldots,N_x\}$$

На сетке w_z вычислим значения:

$$f_i = \hat{f}(z_i) \tag{2.11}$$

Сплайн s будем строить на сетке w_x . Краевые условия выберем следующим образом $s(a) = 0, s'(a) = 0\ s(b) = 0, s'(b) = 0$

$$\sum_{i=1}^{N_z} (s(z_i) - f_i)^2 \to \min. \tag{2.12}$$

В случае кубических сплайнов, как классических так, и эрмитовых, задача (1) сводится к решению пятидиагональной системы линейных алгебраических уравнений.

Для кубических сплайнов справедлива следующая оценка

$$\| f^{\nu} - s^{\nu} \| \le Kh_x^{4-\nu} \| f^{(4)} \|, \tag{2.13}$$

где K — константа, не зависящая от h_x . Задачу можно упростить, если вычислить в узлах сетки w_x значения $\hat{f}_i$.

Для этих целей будем использовать значения $\hat{f}_{cor}(z_i)$ и процедуры сглаживания.

Например, для классических кубических сплайнов можно использовать метод скользящего среднего, метод взвешенной локальной регрессии, фильтр Савицкого–Голея. Следует стремиться, чтобы выполнялись оценки

$$| \hat{f}_i - f(x_i) | = O(h_x^4). \tag{2.14}$$

В этом случае построение сплайна сводится к решению трех диагональной системы линейных алгебраических уравнений и будет выполнена оценка (2).

Для эрмитовых кубических сплайнов в узлах сетки w_x необходимо вычислить $\hat{f}_i$ и значения производных $\hat{f}_i'$. Будем использовать фильтр Савицкого–Голея с кубическими полиномами. В этом случае для построения

эрмитовых кубических сплайнов достаточно локальных вычислений. На каждом интервале $[x_{j-1}, x_j], j = 1,...,n$ эти сплайны представимы в виде [2.15]

$$s(x) = f_{j-1}v((x - x_{j-1})/h_j) + f'_{j-1}w((x - x_{j-1})/h_j) +$$
$$+ f_j v((x - x_j)/h_j) + f'_j w((x - x_j)/h_j),$$

где $v(x) = (|x|-1)^2(2|x|+1); w(x) = x(|x|-1)^2$. (2.15)

Важно отметить, что построение регрессионных моделей с агрегированными входными данными требует использования соответствующих числовых процедур. С этой целью применяется численный вероятностный анализ (ЧВА). Отличительной особенностью ЧВА является наличие развитых арифметических операций над функциями плотности вероятности, для которых вводится понятие ФПВ – значные переменные. В рамках ЧВА имеется возможность вычисления функций от случайных аргументов с использованием процедур построения вероятностных расширений. В рамках ЧВА решаются различные задачи численного анализа, в том числе задачи интерполяции, аппроксимации и оптимизации.

2.3 Задача восстановления зависимости

Проблема восстановления зависимостей по эмпирическим данным уже не первое десятилетие широко обсуждается в научных кругах, поскольку имеет важное прикладное значение [39]. В разделе рассматриваются вопросы исследования изменчивости эмпирических данных больших объёмов при отсутствии знаний о виде восстанавливаемой зависимости. Известным методом решения данной задачи является подход, основанный на применении регрессионных моделей. Предлагается метод построения регрессионных зависимостей на агрегированных данных. Агрегирование данных применяется как метод предварительной обработки эмпирических данных для последующего численного моделирования. Известно, что агрегацию часто используют в задачах анализа данных, когда необходимо перейти от данных с высокой степенью детализации к более обобщённому представлению. Примером таких процедур является простое суммирование, вычисление среднего, медианы, диапазона

максимальных или минимальных значений т.е. интервальных данных, построение функции распределения.

Процедура агрегирования имеет свои преимущества и недостатки. С положительной стороны, мы отмечаем, что подробные данные часто являются очень неустойчивыми из-за воздействия различных случайных факторов, затрудняя обнаружение общих тенденций и шаблонов данных. Во многих случаях полезно рассматривать большие числовые данные в агрегированной форме, такой как суммирование или среднее. Важно иметь в виду, что использование таких процедур агрегирования, как усреднение, исключение экстремальных значений (эмиссия), процедура сглаживания может привести к потере важной информации. Поэтому выбор метода агрегирования является важной задачей, поскольку без предварительного исследования легко получить дополнительную неопределённость, которой нет в исходной постановке.

Для агрегирования данных используются различные математические модели. В тех случаях, когда данные могут быть представлены частотными распределениями рассматриваемых характеристик или признаков, предлагается использовать кусочно-полиномиальные модели. Частным примером кусочно-полиномиальных моделей является гистограмма, которая представляет собой кусочно-постоянную функцию агрегирования. Гистограмма, с точки зрения процесса агрегирования, во многих случаях представляет собой альтернативу операциям усреднения или построения интервальных данных. В отличие от указанных операций применение гистограмм позволяет повысить точность вычисления за счёт использования информации о частотном распределении данных вместо замены набора данных одним значением, например, значением выборочного среднего или моды. Применение гистограммы позволяет уменьшить размерность набора данных, снизить уровень неопределённости и значительно повысить эффективность численных расчётов. Важно отметить, что гистограммы являются примерами использования символьных данных, понятие которых рассматривается в Символьном анализе данных, наравне с интервалами. Billard, L., Diday, E. предложили символический тип данных, называемый

гистограммными переменными, для использования их в регрессионном моделировании [40]. Понятие гистограммно-значной переменной используется для построения гистограммных регрессионных моделей, что является новым важным направлением для теоретических исследований и решения практических задач обнаружения зависимостей в базе данных. Кроме гистограмм для агрегации полезно рассмотреть полиграммы, частотные полигоны, сплайны. Важно отметить, что, несмотря на свою простоту, кусочно-полиномиальные функции охватывают всевозможные диапазоны оценки функции плотности вероятности. Важно также отметить некоторые их свойства. Например, гистограмм, представляя собой кусочно-постоянную функцию, аппроксимирует плотность вероятности с точностью O(h). Однако уже средние точки гистограмм аппроксимируют функцию плотности вероятности с более высокой точностью $O(h^2)$. Следовательно, частотный полигон аппроксимирует функцию с точностью $O(h^2)$.

3 Описание программного модуля

Повышение качества обработки данных больших объемов позволяет при решении задач принимать эффективные решения.

3.1 Описание организации процесса численного моделирования

Для преобразования данных используется программный модуль. Этот модуль включает в себя различные математические модели, такие как гистограмма, полиграмма, частотный полигон, сплайн. В тех случаях, когда данные представляются частотными распределениями рассматриваемых характеристик или признаков, предлагается использовать кусочно-полиномальные модели. Важно отметить, что несмотря на свою простоту, кучочно-полиномальные функции охватывают всевозможные диапазоны оценки функции плотности вероятности. Использование кусочно-полиномальных функция преобразования предлагает более информативное представление существующих зависимостей в данных.

На рисунке 6 представлена общая структура системы. Данная схема содержит 4 блока.

Рисунок 6 - Общая структура системы

Первый блок (БВР) представляет собой систему наблюдаемых показателей и соответствующие им измерения, которые представляются наборами данных, например, в виде дискретных значений, непрерывных сигналов, временных рядов или оцифрованных снимков, представляющих, например, данные дистанционного зондирования земли.

Второй блок представляет собой этап предобработки данных, который наряду с такими функциями как очистка данных, исследование аномальных значений, восстановление пробелов, изучение противоречий и другие.

Предварительная обработка результатов измерения и выборов методов представления необходимы для того чтобы в дальнейшем поднять уровень достоверности полученных оценок и корректно применять методы численно вероятностного анализа для построения законов распределения и функция плотности вероятности для исследуемых показателей. В блоке осуществляется анализ исходных данных, в частности, изучается характер имеющейся неопределенности и далее определяется способ представления данных.

Третий блок представляет собой процедуру агрегации. Суть процедуры составляет методы, позволяющие первоначальный набор данных привести к наборам данных меньшего объема, сохраняя и обнаруживая при этом полезные знания. Например, гистограммный подход к агрегации полезен по следующим причинам, гистограмму можно рассматривать как математический объект, который удобен и для описания и вычисления математических процедур и операций, сохраняя суть частотного распределения данных.

Четвертый блок обозначает что на выходе мы получаем кусочно-полиномиальную модель представления данных.

3.2 Описание алгоритма построения кубического сплайна

3.2.1 Основные теоретические сведения

Для построения кубического сплайна необходимо иметь 4 * (r-1) параметров (r – число узлов сплайна). Данное построение получается в результате неопределенного интегрирования фрагмента исходного сплайнового дифференциального уравнения и называется аналогичной кусочно-полиномиальной формой (pp-формой) по аналогии с полиномиальными сплайнами. Для явного выражения коэффициентов через уже известные значения координат узловых точек, применяют разложение аналогичной кусочно-полиномиальной формы на базисные функции путем подстановки её в краевые условия Эрмита (граничные условия фрагмента сплайна, условия интерполирования и опирания на производные). В результате получается базисная форма (B-форма) сплайна. Такое представление сплайна является

значительно более компактным и записывается через базисные сплайн-функции в виде:

$$S\,(x) = \sum_{j=1}^{r} a_j B_j(x), \qquad (3.1)$$

где $B_j(x)$ — базисные сплайн-функции (как правило локальные), a_j — числовые коэффициенты, задающие вес базисных функций при формировании сплайна, физическим смыслом которых являются обобщённые (линейные и угловые) перемещения металлической линейки в узлах. Число параметров, задающих сплайн, равно числу узлов сплайна. Между параметрами функции на фрагменте и коэффициентами полинома-сплайна существует зависимость, что позволяет с одними коэффициентам находить другие, хотя формулы могут иметь достаточно сложный вид.

Преобразование аналогичной кусочно-полиномиальной формы представления сплайна в базисную форму снижает порядок системы линейных алгебраических уравнений для нахождения неизвестных коэффициентов сплайна, так как они частично выражаются через уже известные параметры – координаты заданных точек (узлов), что позволяет значительно снизить вычислительные затраты за счет возможности применить экономичные методы решения, такие как метод алгебраической прогонки или разновидности метода Гаусса для разрежённых (ленточных) матриц с выбором ведущего элемента столбца.

3.2.2 Алгоритм построения кубического сплайна

На каждом из отрезков $[x_i, x_{i+1}]$, (i = 0, 1, …, n) определим кубический многочлен:

$$S_1(x) = y_{i-1}\frac{(x-x_i)^2[2\,(x-x_{i-1})+h]}{h^3} + y_1\frac{(x-x_i)^2[2\,(x_i-x)+h]}{h^3} +$$

$$+ m_{i-1}\frac{(x-x_i)^2\,(x-x_{i-1})}{h^2} + m_1\frac{(x-x_{i-1})^2\,(x-x_i)}{h^2}, \qquad (3.2)$$

где $h = \frac{b-a}{n}$ — шаг, m_i определяется рекуррентным соотношением:

$f'(a) = m_0 = A;\ f'(b) = m_n = B;\ m_i = L_i M_{i+1} + M_i\ (i = n-1, n-2, \dots 0),$

где $L_0 = 0;\ M_0 = m_0;\ L_i = \frac{-1}{L_{i-1}+4};\ M_i = L_i(M_{i-1} - b_i)(i = 1,2, \dots, n-1);$

$b_i = \frac{3(y_{i+1} - y_{i-1})}{h}$.

А и В — заданы. $A = f'(a);\ B = f'(b)$.

Блок-схема алгоритма построения кубического сплайна представлена на рисунке 7.

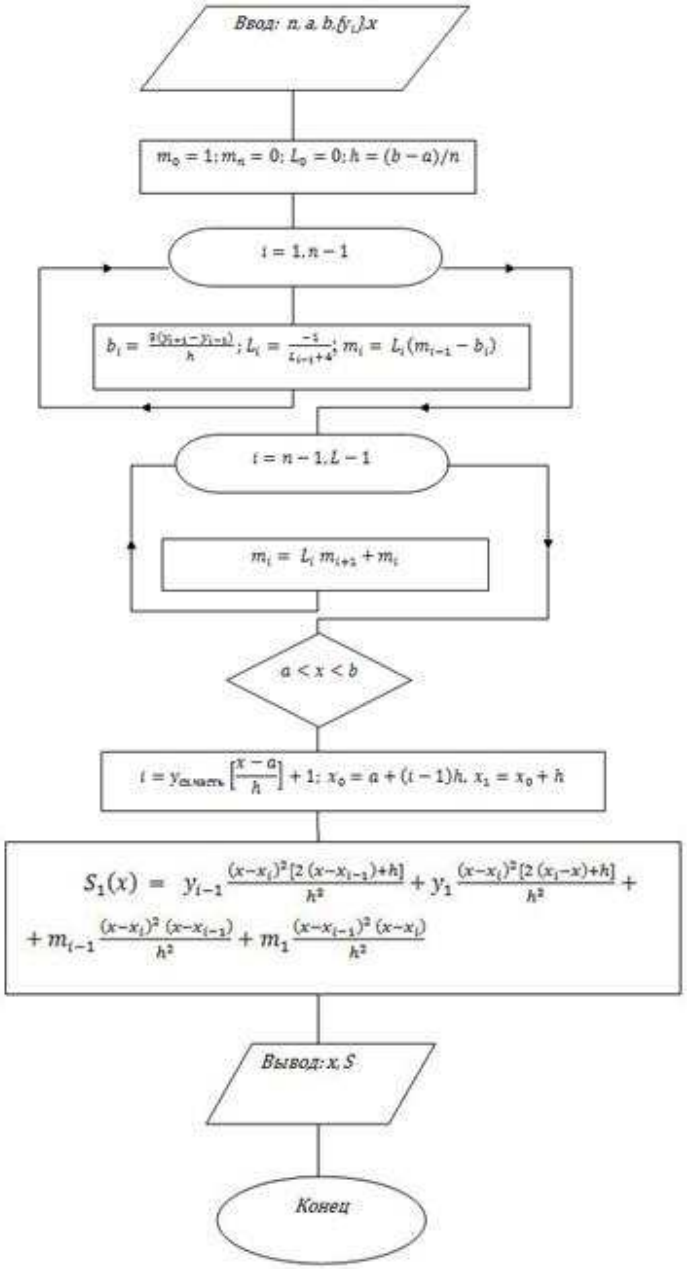

Рисунок 7 – Блок-схема алгоритма построения кубического сплайна

3.3 Описание тестового примера

Рассмотрен пример построения числовой модели для агрегированных временных рядов. Временной ряд хорошо подходит для представления многих практических ситуаций. Следует заметить, что во многих случаях временные ряды анализируются как большие данные. Для анализа взаимосвязи между временными рядами данных мы используем процедуры агрегирования.

Известно, что временные ряды хорошо описывают эмпирические данные для многих практических и теоретических ситуациях, тем не менее, существуют исследования, в которых утверждается, что временные ряды неверно представляют явления, в которых набор реализаций наблюдаемой переменной имеет определенную степень изменчивости. Существуют две типичные ситуации, когда это происходит. Первая ситуация имеет место, если переменная измеряется во времени для каждого индивида группы, и интерес исследователя относится не к индивидам в отдельности, а к группе в целом. В этом случае временной ряд выборочного среднего наблюдаемой переменной во времени будет слабым представлением. Вторая ситуация, когда переменная наблюдается на данной частоте (например, минуты), но ее необходимо анализировать с меньшей частотой (например, дней). Эти две ситуации описывают распределенное и временное агрегирование, соответственно. В каждом случае временные ряды распределений предлагают более информативное представление, чем другие формы агрегированных временных рядов [48].

Рассмотрим пример временной агрегации. На рисунке 8 изображен набор данных (x_t, y_t) расходов на питание (y_t) и дохода (x_t), число наблюдений равно 7125. Графическое представление полного набора данных визуально представляет область изменения и сосредоточения данных. Заметим, что по многим причинам работать с такими данными напрямую неудобно. Для повышения эффективности анализа данных с целью извлечения полезной информации целесообразно эти данные агрегировать в сплайны. Для этих целей

построим в области изменения дохода X сетку $\{x_0 < x_{1,\dots,} x_n\}$. Для каждого отрезка $[x_{i-1}, x_i)$ построим множество $\mathcal{Y}_i = \{y_l | x_l \in [x_{i-1}, x_i)\}$. (3.3)

По множеству $\mathcal{Y}_i$ построим частотный полигон Y_j, $i=1,\dots,n$. Частотные полигоны построенные по множествам каждого отрезка представлены на рисунке 9. Способ агрегирования исходного набора (x_L, y_L) в частотные полигоны $\{Y_i, i = 1,..,n\}$, будем называть «временной» агрегацией. Роль времени играет «доход».

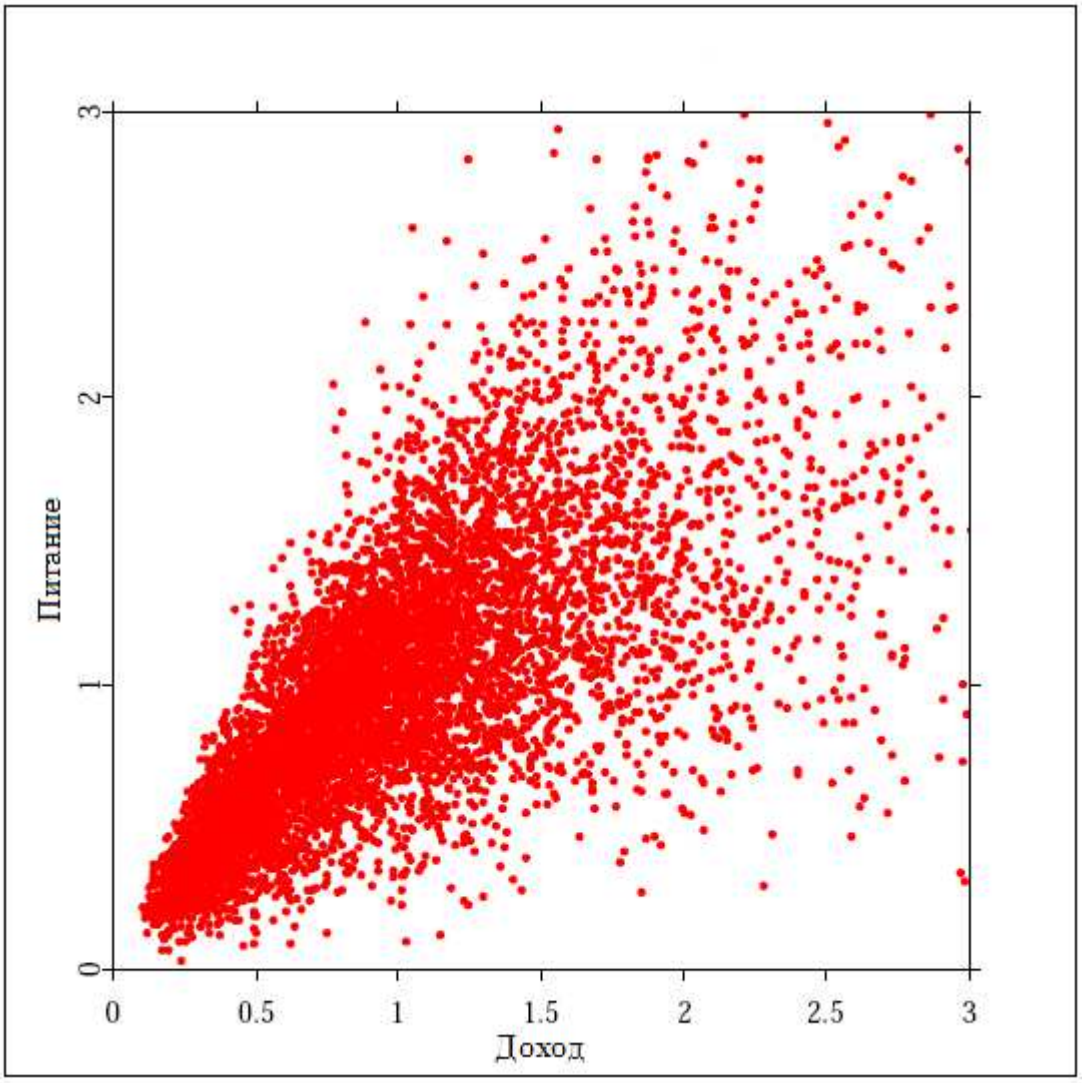

Рисунок 8 –Зависимость расходов на питание от чистого дохода

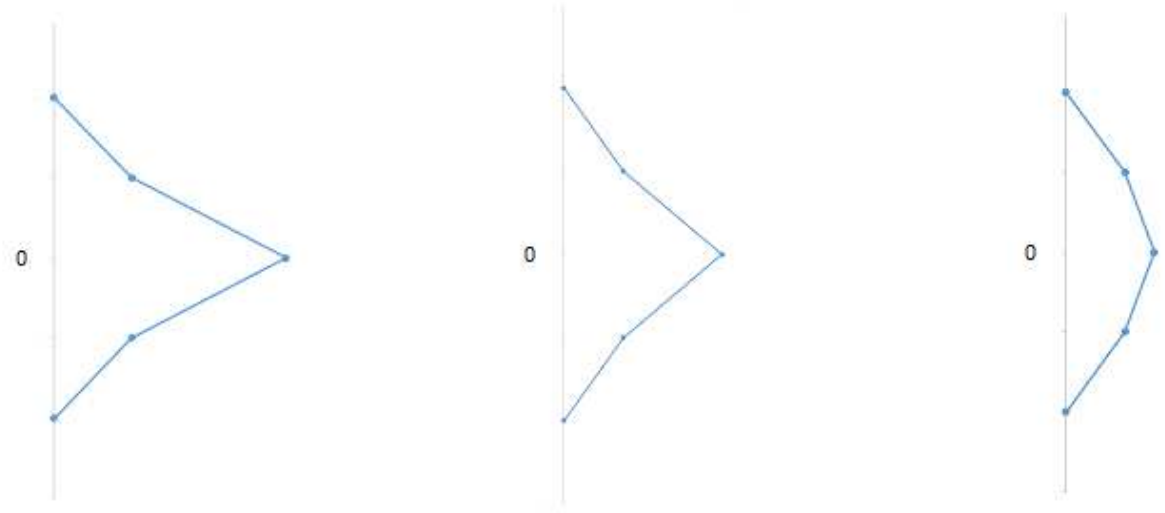

Рисунок 9 – Функции плотности вероятности A_0, A_1, A_k

В этом случае модель регрессии может быть представлена в виде

$$\sum A_k \varphi_k(t) + r(t) \tag{3.4}$$

где *A1*, *A2*,..., A_K — представляют собой функции плотности вероятности на соответственном отрезке.

Далее в качестве модели, используем эрмитовы кубические сплайны. Сплайн определяется на сетке $\{x_1^i, x_2^i, x_3^i\}$. Граничные условия имеют вид:

$s(x_1^i) = 0$, $s'(x_1^i) = 0$, $s(x_3^i) = 0$, $s'(x_3^i) = 0$

Кроме того $s'(x_2^i) = 0$ и значение $s(x_2^i)$ выберем из условия

$$\int_{x_1^i}^{x_3^i} s(\xi) d\xi = 1. \tag{3.5}$$

Для нахождения *A1*, *A2*, *A3* потребуем выполнения условия оптимальности

$$\sum_{i=1}^{7125} \rho^2(Y_i, \hat{Y}_i) \to \min. \tag{3.6}$$

На рис. 10 представлены функции плотности вероятности зависимости расходов на питание от чистого дохода.

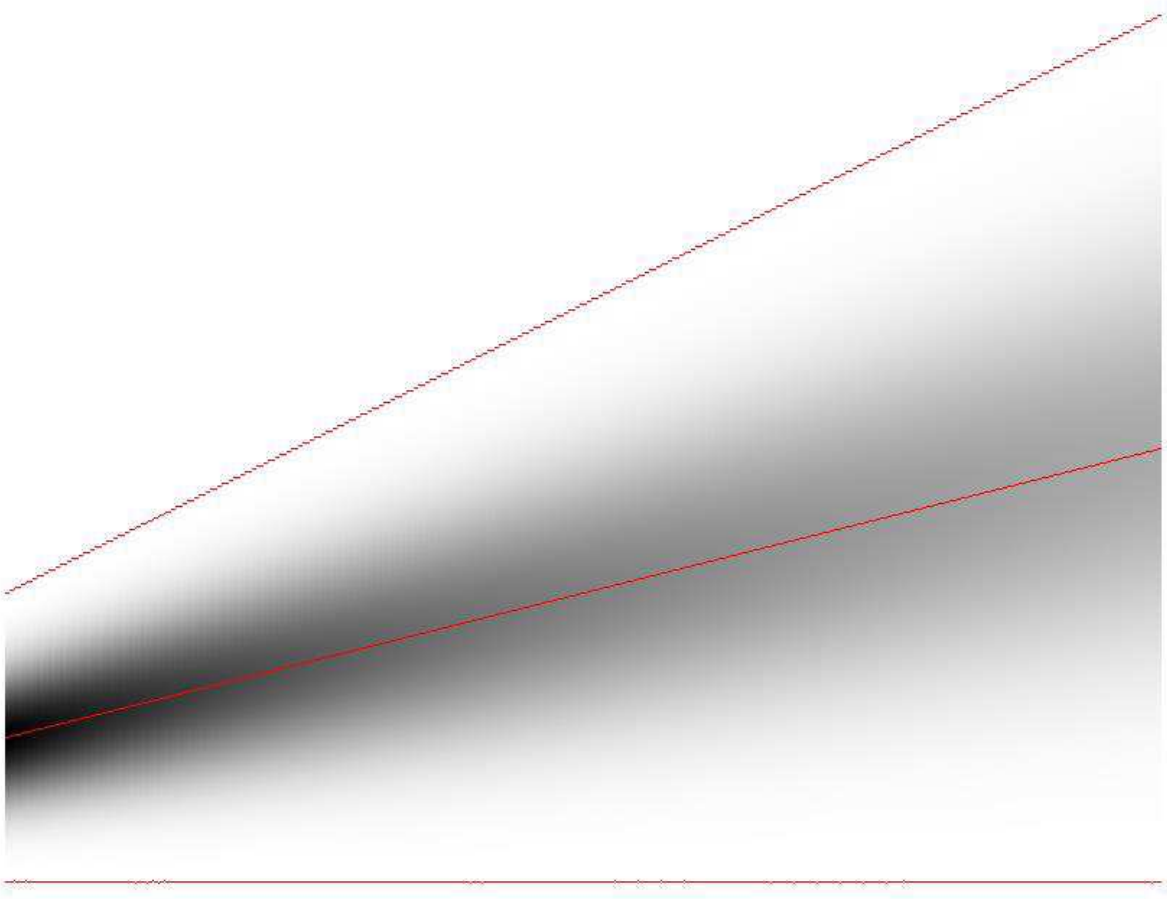

Рисунок 10 – Распределение плотности вероятности зависимости расходов на питание от чистого дохода

На первом этапе агрегирования данные представлены для каждого значения дохода в виде частотного полигона. Данные регрессии представлены в виде кубических сплайнов Эрмита. Таким образом, данные о зависимости расходов на питание от чистого дохода агрегируются с помощью кубических эрмитовых сплайнов. Визуальное представление показывает изменение максимальной, минимальной и наиболее вероятной зависимости. Оттенки серого показывают распределение плотности вероятности.

ЗАКЛЮЧЕНИЕ

В настоящее время область анализа и обработки данных больших объемов представляет большой интерес для исследователей по всему миру. В пользу данного заключения говорит внушительное количество издаваемых публикаций, книг, монографий и пособий, проводимых конференций и представляемых на нах докладов по данной тематике.

В ходе выполнения целей и задач исследования на тему численной обработки данных больших объемов рассмотрены существующие методы обработки больших временных рядов, обозначены имеющиеся проблемы и сформированы задачи исследования.

На этапе моделирования применяется численный вероятностный анализ, который является новым направлением в вычислительной математике.

На этапе постобработки результаты моделирования представляются в графическом виде, показывающем гарантированные области с внутренним распределением.

В результате осуществлена разработка модуля численной обработки большого временного ряда и проведено тестирование на основе практической задачи.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Большие данные [Электронный ресурс]: // Режим доступа: http://cyberlenika.ru/articles/n/bolshie-dannye-bolshie-problemy
2. Т. Гаврилова, Л. Григорьев. Бизнес держится на знаниях, сам того не зная. Журнал «Персонал-Микс» (№2, 2004).
3. Работа с Big Data: основные области и возможности [Электронный ресурс]: // Режим доступа: http://www.marketing.spb.ru/lib-around/stat/Big_Data.htmD
4. Big Data — хранение, обработка и анализ огромных массивов информации [Электронный ресурс]: // Режим доступа: https://web-creator.ru/articles/bigdata.
5. Десять основных тенденций 2005 года в области Business Intelligence и Хранилищ данных [Электронный ресурс]: // Режим доступа: http://citforum.ru/gazeta/3/.
6. Управление знаниями [Электронный ресурс]: // Режим доступа: http://msk.treko.ru/show_dict_ 390
7. В. Дюк. Data Mining – интеллектуальный анализ данных [Электронный ресурс]: // Режим доступа: http://www.iteam.ru/publications/it/section_92/article_1448/
8. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP / А.А. Барсегян и др. – 2-е изд., перераб. и доп. – СПб.: БХВ- Петербург, 2007. – 384 с.
9. Big Data: как не утонуть в океане информации [Электронный ресурс]: // Режим доступа: https://www.kom-dir.ru/article/1527-big-data.
10. Хранение больших объемов данных [Электронный ресурс]: // Режим доступа: http://www.qsan.su/decision/peredacha-dannyh.html.
11. Обработка больших данных [Электронный ресурс]: // Режим доступа: https://www.ibm.com/developerworks/ru/library/bd-gobigsql/.
12. Big Data: проблема, технология, рынок данных [Электронный ресурс]: // Режим доступа: http://compress.ru/article.aspx?id=22725.

13. Гистограммы второго порядка для численного моделирования в задачах с информационной неопределенностью Попова О.А. Известия ЮФУ. Технические науки. 2014. № 6 (155). С. 6-14.
14. Полигон и гистограмма [Электронный ресурс]: // Режим доступа: http://umk.portal.kemsu.ru/uch-mathematics/papers/posobie/t4-3.htm.
15. Гистограмма [Электронный ресурс]: // Режим доступа: https://basegroup.ru/community/glossary/histogramДобронец Б.С., Попова О.А.
16. Гистограммные временные ряды // Тр. X Международной конференции ФАМЭТ-2011. Красноярск: КГТЭН, СФУ, 2011. С.130–133.
17. Воробьев О.Ю. Современные теории неопределенности: эвентологический взгляд // Тр.VIII Международной конференции ФАМ. Красноярск: СФУ, 2009. С. 83–92.
18. Численный вероятностный анализ для исследования систем в условиях неопределенности Добронец Б. С., Попова О. А. Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2012. № 4 (21). С. 39-46.
19. Б. С. Добронец, Интервальная математика, Красноярск, КГУ, 2004.
20. Л.Т.Ащепков, Д.В.Давыдов, Универсальные решения интервальных задач оптимизации и управления, Институт прикладной математики ДВО РАН, М., Наука, 2006.
21. А.И.Орлов, Теория принятия решений, Учебное пособие, М., Экзамен, 2005.
22. В.А.Перепелица, Ф.Б.Тебуева, Дискретная оптимизация и моделирование в условиях неопределенности данных, М., Академия Естествознания, 2007.
23. Ferson S., Ginzburg L. Different methods are needed to propagate ignorance and Variability // Reliability Engineering and System Safety. – 1996. – № 54. – С. 133-144.
24. Neumaier A. Clouds, Fuzzy Sets and Probability Intervals // Reliable Computing. – 2004. – № 10. – С. 249-272.

25. Dempster A.P. Upper and lower probabilities induced by a multi-valued mapping // Annals of Mathematical Statistics. – 1967. – № 38. – С. 325-339.
26. Dobronets B.S., Krantsevich A.M., Krantsevich N.M. Software implementation of numerical operations on random variables // Journal of Siberian Federal University. Mathematics & Physics. – 2013. – № 6 (2). – С. 168-173.
27. Тарасенко Ф. П. Непараметрическая статистика. Томск. ТГУ, 1976. С. 294.
28. Добронец Б.С., Попова О.А. Численный вероятностный анализ неопределенных данных. Красноярск СФУ 2014г. 168с.
29. Определение среднего значения, вариации и формы распределения. Описательные статистики [Электронный ресурс]: // Режим доступа: http://baguzin.ru/wp/?p=5381.
30. Численные операции над случайными величинами и их приложения Добронец Б.С., Попова О.А. Журнал Сибирского федерального университета. Серия: Математика и физика. 2011. Т. 4. № 2. С. 229-239.
31. Элементы численного вероятностного анализа Добронец Б.С., Попова О.А. Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева. 2012. № 2 (42). С. 19-23.
32. Технология извлечения и визуализации знаний на основе численного вероятностного анализа неопределенных данных Попова О.А. Информатизация и связь. 2013. № 2. С. 63-66.
33. Fisher R. A. (1932). Statistical Methods for Research Workers, Fourth Edition. Oliver and Boyd, Edinburgh.
34. Boris S. Dobronets and Olga A. Popova. The Numerical Probabilistic Approach to the Processing and Presentation of Remote Monitoring Data // Journal of Siberian Federal University. Engineering & Technologies, 2016, 9(7), 960-971.
35. Что такое сплайны [Электронный ресурс]: // Режим доступа: http://cpu3d.com/lesson/chto-takoe-splayny.

36. Ревякин, С.А. О важности качественной информации для принятия управленческих решений. – http://www.global-katalog.ru/cncat_jump.php?13146.
37. Примеры применения сплайнов, сохраняющих интеграл, при обработке изображения О. П. Федорова Пятая Сибирская конференция по параллельным и высокопроизводительным вычислениям Томск, 2010 г.
С. 167-170.
38. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP / А.А. Барсегян и др. – 2-е изд., перераб. и доп. – СПб.: БХВ- Петербург, 2007. – 384 с.
39. Элементы численного вероятностного анализа
Добронец Б.С., Попова О.А. Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева. 2012. № 2 (42)
40. Алгоритм построения сплайна [Электронный ресурс]: // Режим доступа: http://studopedia.su/3_13686_lektsiya-.html.

ПРИЛОЖЕНИЕ А

Исходный программный код

```
Program test1;
Const n=4;
type vec = array[0..n] of real;
Var x0,x1,h,s,t,sint : real;
   i,j : integer;
   x, f, m : vec;
function sp(t: real; i :byte ) :real;
var tt :  real;
begin

   tt := t*t;
if i= 1 then sp  := 1 -3*tt+2*tt*t;  //базисные функции
if i= 2 then sp  := 3*tt-2*tt*t;
if i= 3 then sp  := t*(1-t)*(1-t);
if i= 4 then sp  := -t*t*(1-t);
end;
function spl(x,x0,h,f0,f1,m0,m1:real):real;
Var t : real;
begin
t:=(x-x0)/h;
spl := f0*sp(t,1)+f1*sp(t,2)+ m0*h*sp(t,3)+ m1*h*sp(t,4);
end;
begin
h:=13;
x[0]:=33;f[0]:=0; m[0]:=0;
x[1]:=41;f[1]:=0.366; m[1]:=0.019;
x[2]:=49;f[2]:=0.484; m[2]:=-0.008;
x[3]:=57;f[3]:=0.148; m[3]:=-0.019;
```

```
x[4]:=63;f[4]:=0; m[4]:=0;

for i:= 0 to 4 do
begin
f[i]:=f[i]/12.9740;
m[i]:=m[i]/12.9740;
end;

sint:=0;
for i:= 0 to 16 do
begin

t:= 27+ i*2;

if (t >= 33) and (t<= 41) then j:=0;
if (t > 41) and (t<= 49) then j:=1;
if (t > 49) and (t<= 57) then j:=2;
if (t > 57) and (t<= 65) then j:=3;

s:=spl(t,x[j],h,f[j],f[j+1], m[j],m[j+1]);
sint:=sint+s;
writeln(t:6:3,s:10:3);
end;

sint:=sint*2;
writeln (sint:8:4);
end.
```

ПРИЛОЖЕНИЕ Б

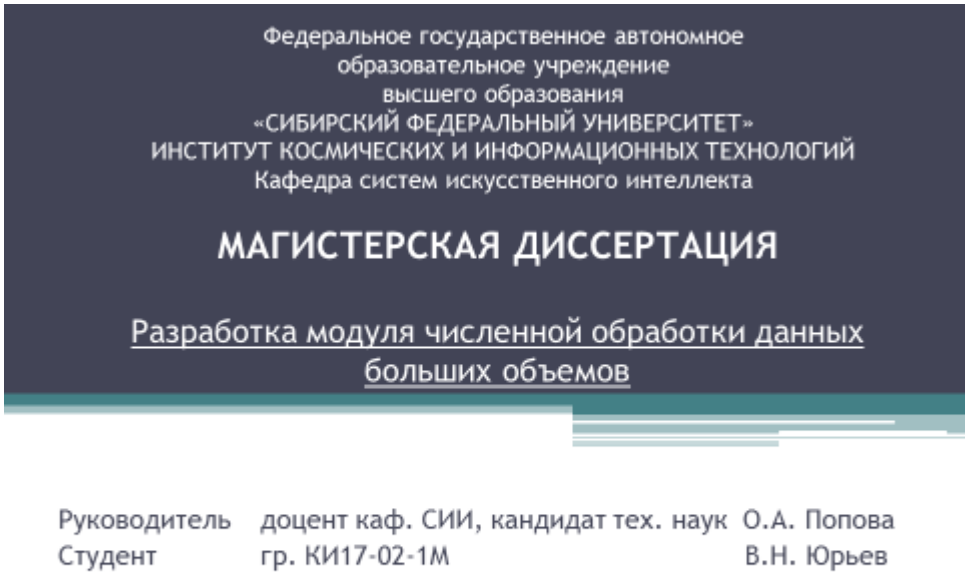

Рисунок Б.1 – Титульный лист

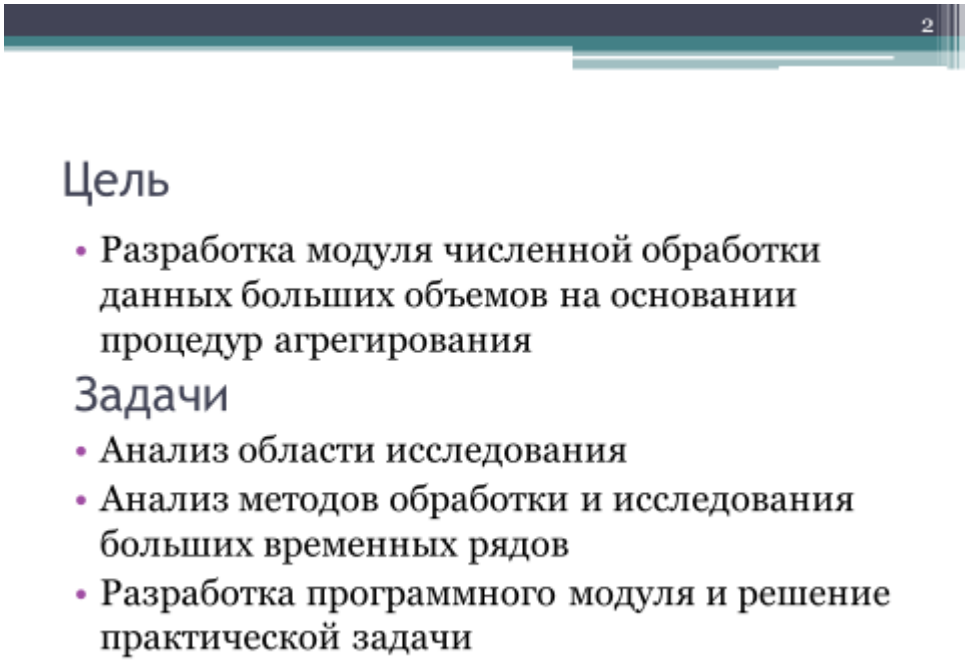

Рисунок Б.2 – Цель и задачи исследования

Понятие временного ряда

Рисунок Б.3 – Понятие временного ряда

Методы анализа временных рядов

- Спектральный анализ
- Корреляционный анализ
- Модели скользящего среднего
- Многоканальные модели скользящего среднего

Рисунок Б.4 – Методы анализа временных рядов

Рисунок Б.5 – Кусочно-полиномиальные модели

Рисунок Б.6 – Сплайн

Рисунок Б.7 – Общая структура модуля

Рисунок Б.8 – Зависимость расходов на питание от чистого дохода

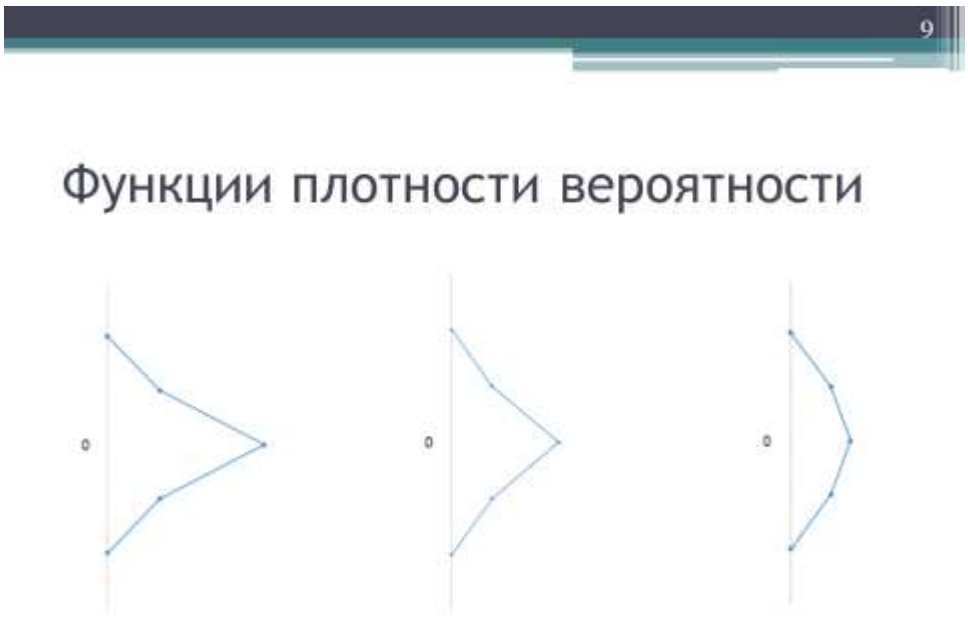

Рисунок Б.9 – Функции плотности вероятности

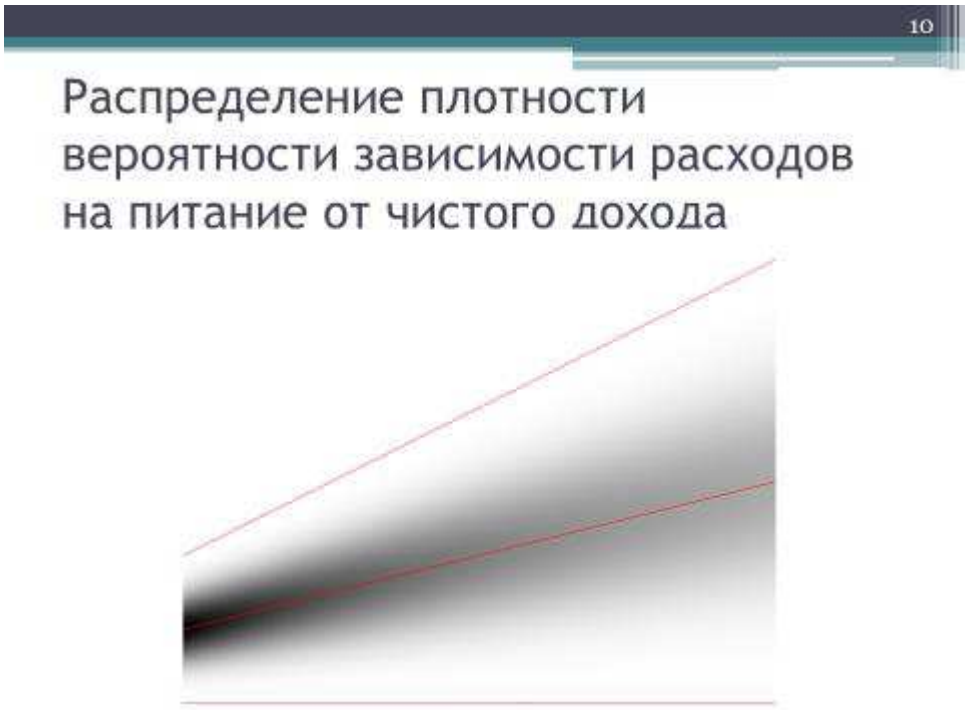

Рисунок Б.10 – Распределение плотности вероятности зависимости расходов на питание от чистого дохода

Федеральное государственное автономное
образовательное учреждение
высшего образования
"СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ"
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ
Заведующий кафедрой
_____ Г.М.Цибульский
«____» _____ 20___г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Разработка модуля численной обработки данных больших объемов

09.04.02 Информационные системы и технологии
09.04.02.01 Информационно-управляющие системы

Руководитель	_____ подпись	_____ дата	доц., канд.техн. наук должность, ученая степень	О.А.Попова
Выпускник	_____ подпись	_____ дата	КИ17-02-1М	В.Н.Юрьев
Рецензент	_____ подпись	_____ дата	проф., д-р техн. наук должность, ученая степень	Л.А.Казаковцев
Нормоконтролер	_____ подпись	_____ дата	доц., канд. техн. наук должность, ученая степень	О.А.Попова

Красноярск 2019