На правах рукописи

МИНИСТЕРСТВО НАУКИ ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Волгоградский государственный технический университет»

Н.П. Садовникова

М.В. Щербаков

ТЕХНОЛОГИИ АНАЛИЗА ДАННЫХ

Учебное пособие

Волгоград – 2021

УДК 519.25

Рецензент:
профессор кафедры «Теория и методика обучения математике и информатике» ВГСПУ д-р пед. наук, профессор *Т. М. Петрова*

Печатается по решению редакционно-издательского совета
Волгоградского государственного технического университета

Технологии анализа данных: учебное пособие / Садовникова Н.П., Щербаков М.В.: ВолгГТУ, Волгоград, 2021.– с. 75

В учебном пособии представлен всесторонний анализ вопросов связанных с современными технологиями анализа данных. Рассматриваются наиболее востребованные методы. Даются основные понятия языка статистического анализа R, разбираются примеры его использования для решения практических задач.

Учебное пособие предназначено для магистров, обучающихся по программам магистратуры по профилю «искусственный интеллект» по направлениям 09.04.01 «Информатика и вычислительная техника», 09.04.03 «Прикладная информатика», 09.04.02 "Информационные системы и технологии".

Учебное пособие выполнено в рамках реализации гранта на разработку программ бакалавриата и программ магистратуры по профилю «Искусственный интеллект», а также на повышение квалификации педагогических работников образовательных организаций высшего образования в сфере искусственного интеллекта (конкурс 2021-ИИ-01 от 10.06.2021).

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Анализ данных уже многие годы является одним из наиболее популярных методов научных исследований. Прежде всего, анализ данных необходим для структурирования и анализа информации о том или ином объекте или процессе. Нет практически ни одной сферы человеческой деятельности, где бы ни применялись методы анализа данных. Будь то описательная статистика, визуализация данных, регрессионный анализ и т.д. Несмотря на то, что накоплен значительный опыт использования методов анализа данных, сегодня появилась необходимость создания новых технологий, в которых большое внимание уделяется решению задач обобщения, выявления закономерностей, нахождения ассоциаций и т.д. Связано это, прежде всего, с массовым распространением информационных сервисов и технологий, которые являются поставщиками огромных массивов разнородных данных. Для их анализа не достаточно знать только методы статистического анализа. Современная наука о данных является междисциплинарной и базируется на знаниях их области прикладной статистики, искусственного интеллекта, машинного обучения, теории оптимизации, баз данных и пр. К задачам анализа данных можно так же отнести и вспомогательные, но важные с практической точки зрения задачи сокращения размерности и анализа качества данных.

Под *анализом данных* будем подразумевать процедуру целенаправленных преобразований данных с целью извлечения полезной информации, приобретения новых знаний и принятия решений.

Анализ данных – активно развивающаяся научная дисциплина. Появились новые подходы, ориентированные на работу с большими массивами разнородных данных. Крупнейшие поставщики информационных технологий (IBM, Oracle, Microsoft, Hewlett-Packard, EMC и др.) стали использовать в своих деловых стратегиях понятие «Big data». С начала 2000 годов появилась тенденция создания нового научного

направления связанного с анализом данных. В 2001 году Уильям Кливленд опубликовал план развития технических аспектов статистических исследований и выделил науку о данных как отдельную академическую дисциплину, в которой эти технические аспекты должны быть сконцентрированы. Образовательные программы многих университетов мира вводят новые дисциплины, связанные с технологиями хранения и интеллектуального анализа данных, активизируют усилия на создание более совершенных алгоритмов и методов.

Поведение людей, настроение, характеристики любого вида деятельности, движения курсора и глаз, параметры сна и дыхания – все это сегодня может быть представлено в виде набора данных. Новый термин «датификация» определяет процесс преобразования привычных для нас форм восприятия действительности в цифровую форму, что обеспечивает возможность получения новых знаний для решения самых разных задач.

Умение работать с данными, извлекать из них нужную информацию является одной из наиболее перспективных и востребованных компетенций ближайшего времени.

ГЛАВА 1. ДАННЫЕ И «БОЛЬШИЕ ДАННЫЕ»

При изучении самых разных объектов, явлений и процессов исследователь, как правило, выделяет набор свойств, которые в наибольшей степени их характеризуют. С точки зрения математики можно сказать, что каждый объект представлен вектором переменных, которые определяют его свойства. Эти свойства должны быть каким либо образом быть измерены и зафиксированы. Результаты измерений принято называть данными.

Данные – совокупность объектов (наблюдений, случаев) и признаков (переменных), их характеризующих. В переводе с латинского слово data означает «данность» или «факт». Данные должны быть представлены в форме, пригодной для хранения, передачи и обработки. Анализ данных используется для получения новых знаний об изучаемом объекте (процессе, явлении).

Наиболее распространенный способ представления данных таблица «объект-признак», строки которой соответствуют объектам, столбцы – признакам. Если в качестве объекта определены моменты времени или номера наблюдений над объектом, то принято такие наборы данных называть временными рядами (time-series data). Если объекты разные (люди, страны, предприятия и пр.), то говорят о данных пространственного типа (cross-sectional data).

В последнее время особое значение приобретают технологии связанные с понятием «Big Data» («больших данных»), что так или иначе связано с резким ростом количества доступной для анализа информации. В новых условиях традиционные подходы к анализу данных перестали работать. Появилась необходимость в создании новых технологий анализа данных и компьютерные алгоритмы, которые считают значительно быстрее и могут приспосабливаться под задачу и самостоятельно обучаться. Кроме того необходимы достаточные мощности для обработки и хранения данных, а также простые и доступный способы сбора данных.

Определяя понятие Big Data часто говорят о четырех V: Volume, Variety, Velocity и Value (объеме, вариативности, скорости и ценности).

Big Data – это «технологии и архитектуры нового поколения для экономичного извлечения ценности из разноформатных данных большого объема путем их быстрого захвата, обработки и анализа» [1]. Понятие больших данных подразумевает работу с информацией огромного объема и разнообразного состава, весьма часто обновляемой и находящейся в разных источниках в целях увеличения эффективности работы, создания новых продуктов и повышения конкурентоспособности [2].

В качестве важнейшего стратегического ресурса технологий в сфере Big Data были признаны во многих странах мира. В ближайшее время в США планируется проведение комплексных мероприятий в целях активного использования технологий Big Data на ключевых направлениях государственной политики. В 2013 году правительство Японии опубликовало информацию о разработке национальной программы по Большим Данным.

Развитие технологий Big Data широко представлены на многочисленных специализированных конференциях и форумах, среди которых стоит выделить O'Reilly Strata Conference Making Data Work, HadoopWorld, Big Data Techcon, KDD Knowledge Discovery and Data Mining, SIGIR Information Retrieval, ICML International, Conference on Machine Learning, ICDM International Conference on Data Mining.

1.1 ИСТОЧНИКИ И ТИПЫ ДАННЫХ

Если еще двадцать лет назад в учебниках по статистики писалось, что основными источниками данных являются специально организованные наблюдения или эксперименты и различные формы отчетности, то теперь можно говорить, что данные «появляются благодаря постепенной датификации всего и вся» [3].

Датификация (data-ization) – процесс представления явлений в количественном формате для дальнейшего сведения в таблицу и анализа [3]. Датификация подразумевает перевод в анализируемую форму результатов любых взаимодействий, местоположения, сигналов активности головного мозга, движения зрачка и курсора, настроения и поведения людей.

Разнородная и разнотипная информация, которая собирается с использованием гаджетов, датчиков и благодаря сетевым технологиям передается и сохраняется, может стать источником исследований, которые помогут понять суть самых разных явлений и стимулировать развитие новых научных направлений. Вместе с тем, разнообразие данных является источником ряда проблем, которые требуют решения.

Как известно, наиболее удобны для обработки структурированные данные. В случае, когда изначально задан способ организации массивов данных, например в виде таблицы, значительно проще выполнить любые операции, связанные с обработкой и анализом. Тем не менее, большинство информации поступает к нам в неструктурированном или слабоструктурированном виде. Это разнообразные текстовые документы, изображения, звуковые файлы, электронные письма, видеофайлы, веб-страницы и пр. Для преобразования таких данных к виду, пригодному для анализа специальные методы, алгоритмы и инструменты. В качестве задач которые решаю подобные инструменты можно выделить: поиск и агрегация контента из различных источников, извлечение данных в соответствии с

заданными параметрами, семантический анализ, предоставление итоговых сведений аналитику или конечному пользователю.

Данные классифицируют по следующим признакам:

- по количеству переменных (одномерные, двумерные или многомерные наборы данных);
- по времени (текущие или исторические);
- по цели анализа (первичные, вторичные);
- по способу представления (числовые, символьные, логические, графические т.д.)

Данные, которые явно или неявно связаны с определенными датами или промежутками времени часто называют темпоральными.

Кроме того данные можно разделить на транзакционные и нетранзакционные. Транзакционные данные – это данные, каждая запись которых относится к фиксированному моменту времени и содержит сведения, фиксированные на данный момент времени, не изменяющиеся в будущем. Соответственно, нетранзакционными являются все остальные данные.

Важное значение для задач организации хранения данных имеют *метаданные* – данные о данных, которые содержат информацию о составе данных, происхождении, местонахождении, содержании, статусе, форматах и формах представления, условиях доступа и пр.,

1.2 ТЕХНОЛОГИИ СБОРА И ХРАНЕНИЯ ДАННЫХ

С появлением необходимости в хранении и обработке больших массивов разнородных данных системы извлечения, хранения и управления данными становятся одними из наиболее востребованных компонент современных информационных и вычислительных систем.

Консолидация - комплекс методов и процедур, направленных на извлечение данных из различных источников, обеспечение необходимого уровня их информативности и качества, преобразование в единый формат, в котором они могут быть загружены в хранилище данных или аналитическую систему.

Основные критерии оптимальности с точки зрения консолидации данных [4]:

- обеспечение высокой скорости доступа к данным;
- компактность хранения;
- автоматическая поддержка целостности структуры данных;
- контроль непротиворечивости данных.

Поддержка консолидации данных осуществляется с использованием технологии ETL.

ETL (extraction, transformation, loading) – комплекс методов, реализующих процесс переноса исходных данных из различных источников в аналитическое приложение или поддерживающее его хранилище данных [4]. ETL следует рассматривать не только как процесс переноса данных из одного приложения в другое, но и как инструмент подготовки данных к анализу.

Процесс ETL состоит из трех основных этапов:

- **Extraction –** непосредственно процесс извлечения информации из источника данных, а также определение состава данных, периодичности выгрузки и правил фильтрации необходимой информации

- **Transformation** – очистка и преобразование информации в формат, поддерживаемый целевой системой
- **Loading** – загрузка преобразованной информации в целевую систему.

Хранилище данных (Data Warehouse) – предметно-ориентированная информационная база данных. Строится на базе клиент-серверной архитектуры, СУБД и утилит поддержки принятия решений. Данные, поступающие в хранилище данных, становятся доступны только для чтения. Они не удаляются и не переписываются – вносятся только новые данные, это необходимо для изучения динамики изменения данных во времени. Хранилища данных, в большинстве случае, консолидированы с несколькими истониками.

Требования к хранилищам данных [5]:

- поддержка высокой скорости получения данных из хранилища;
- поддержка внутренней непротиворечивости данных;
- возможность получения и сравнения так называемых срезов данных ;
- наличие удобных утилит просмотра данных в хранилище;
- полнота и достоверность хранимых данных;
- поддержка качественного процесса пополнения данных.

Хранилище на самом верхнем уровне состоит, как правило, из трех подсистем:

- подсистемы загрузки данных (ПО, которое в соответствии с определенным регламентом извлекает данные из источников и приводит их к единому формату, определенному для хранилища. Отвечает за формализованную логическую согласованность, качество и интеграцию данных);
- подсистемы обработки запросов и представления данных (служат для извлечения данных, их аналитической обработки и представления конечным пользователям);

- подсистемы администрирования хранилища (реализуют задачи, связанные с поддерживанием системы и обеспечением ее устойчивой работы и расширения).

Ключевым компонентом организации хранилищ данных является **OLAP (On-Line Analytical Processing)** – технология комплексного многомерного анализа данных. Концепция OLAP была описана в 1993 году Эдгаром Коддом, известным исследователем баз данных и автором реляционной модели данных. Им же были сформулированы требования, которые позже преобразовали в так называемый тест FASMI (Fast Analysis of Shared Multidimensional Information), который позволяет определить технологию OLAP только пятью ключевыми словами: Быстрый Анализ Разделяемой Многомерной Информации.

Fast (Быстрый) – анализ должен производиться одинаково быстро по всем аспектам информации. Приемлемое время отклика – 5 с или менее.

Analysis (Анализ) – должна быть реализована возможность осуществлять основные типы числового и статистического анализа, предопределенного разработчиком приложения или произвольно определяемого пользователем.

Shared (Разделяемой) – множество пользователей должно иметь доступ к данным, при этом необходимо контролировать доступ к конфиденциальной информации.

Multidimensional (Многомерной) – ключевое требование OLAP реализация многомерного представление данных, включая полную поддержку для иерархий и множественных иерархий

Information (Информации) – приложение должно иметь возможность обращаться к любой нужной информации, независимо от ее объема и места хранения.

ТЕХНОЛОГИИ APACHE HADOOP

Apache Hadoop – это фреймворк, основной задачей которого является распределенная обработка больших массивов данных распределенных по вычислительным кластерам с использованием простых программных моделей [6]. Hadoop как проект с открытым исходным кодом находится под управлением Apache Software Foundation (http://www.apache.org/).

Hadoop состоит из распределенной файловой системы HDFS(Hadoop Distributed File System), основной задачей которой является хранение данных и системы MapReduce, которая предназначена для вычислений и обработки данных на кластере.

HDFS разбивает большие файлы на части, управляемые разными узлами кластера, каждая часть реплицируется между несколькими машинами так, чтобы возможные ошибки не привели к недоступности данных. Несмотря на то, что данные разбросаны по серверам данных (DataNode), они формируют одно пространство с помощью сервера имён (NameNode). Для каждого файла сервер имён хранит его путь, список блоков и их реплик.

Hadoop ограничивает число связей, которые могут создаваться процессами, поскольку каждая отдельная запись обрабатывается отдельно от других. Программы должны быть написаны в соответствии с конкретной программной моделью MapReduce. MapReduce-опрерация состоит из двух фаз:

1) *map* – выполняется параллельно и (по возможности) локально над каждым блоком данных.

2) *reduce* – дополняет map агрегирующими операциями.

Проекты Hadoop для **Machine learning**

Mahout – библиотека алгоритмов machine learning на основе MapReduce. Кластеризация, коллаборативная фильтрация, случайные деревьея, средства для факторизации матриц.

MLlib. Базовая статистика, линейная и логистическая регрессия, SVM, k-means, SVD и PCA, а также такие примитивы оптимизации как SGD и L-BFGS. Scala интерфейс использует для линейной алгебры Breeze, Python интерфейс — NumPy.

Более подробно с технологиями работы Apache Hadoop можно ознакомиться на сайте https://developer.yahoo.com/hadoop/tutorial/module1.html или пройти on-line курсы на bigdatauniversity.com/

1.3 КАЧЕСТВО ДАННЫХ

Данные не являются заведомо ошибочными или ложными, но их беспорядочность не представляет особых проблем при многократном увеличении масштаба. Она может быть даже выгодной, так как, используя лишь небольшую часть данных, мы упускали из виду широкое поле подробностей, где обнаруживается масса знаний [3]. Тем не менее, задачи анализа качества данных и улучшения их качества являются актуальными. Никакие самые мощные алгоритмы и технологии не дадут качественного результата если в исходные данные не будут соответствовать необходимым критериям качества. Полученные решения будут искажать истинную картину, определять ложные тенденции или закономерности.

Качество данных – совокупность свойств и характеристик этих данных, определяющих степень пригодности для последующего анализа [1].

Управление качеством данных – обеспечение требуемого качества данных для решения конкретной задачи.

Основными критериями качества данных являются: своевременность, точность, полнота, интерпретируемость.

При классификации проблем, связанных с качеством данных можно выделить три уровня [9]:

1) Концептуальный. В данном случае низкое качество связано с неверной стратегией сбора данных. Их может быть недостаточно для

всестороннего описания предметной области или же наоборот слишком много, но они не имеют отношения к решаемой задаче.

2) Аналитический. К этому уровню относятся шумы данных, аномальные значения, противоречивые и дублирующие записи и пропуски.

3) Технический. Снижение качества данных связано с нарушениями в структуре данных, их целостностью и полнотой, некорректностью форматов и кодировкой и т. п., что мешает интегрированию данных.

Методы повышения качества данных можно отнести к одному из этапов описанного ранее ETL-процесса [4,7,8]:

– *очистка данных* – процесс выявления и исправления ошибок в исходной информации, т. е. оценка достоверности данных, выявление ошибочных подозрительных данных: аномалий, дубликатов, противоречий и т. п.;

– *предобработка данных* – процесс подготовки данных к решению конкретной аналитической задачи и приведение их в соответствие с требованиями, определяемыми спецификой этой задачи и способами ее решения, т. е. понижение размерности исходной информации, устранение незначащих признаков и т. п.;

– *обогащение данных* – процесс насыщения данных новой информацией, позволяющей сделать их более ценной для определенной аналитической задачи, т. е. привлечение информации из дополнительных источников, заполнение пропусков в информации, выявление связей между объектами и т. п.

1.4 ЗАДАЧИ И МЕТОДЫ АНАЛИЗА ДАННЫХ

Как было сказано выше, цель анализа данных – получение новых знаний об изучаемом объекте. На верхнем уровне можно выделить задачи выявления структуры, анализа зависимостей и их использование для предсказания неизвестных значений анализируемых факторов. В

литературе можно встретить множество классификаций, которые являются детализацией этих трех задач. Так можно выделить задачи выявления значимых факторов, классификации, группировки, выявления связей, прогнозирования и т.д.

Для решения каждой задачи существует разнообразные методы, список которых все время пополняется. Многие из них заимствованы из разных областей знаний и адаптированы под конкретные задачи.

Приведенный ниже список методов не претендует на полноту, однако в нем отражены наиболее востребованные в различных отраслях подходы [10].

A/B-тестирование (A/B testing). Методика, в которой контрольная выборка поочередно сравнивается с другими. Тем самым удается выявить оптимальную комбинацию показателей для достижения, например, наилучшей ответной реакции потребителей на маркетинговое предложение. Большие данные позволяют провести огромное количество итераций и таким образом получить статистически достоверный результат.

Ассоциативные правила (Association rule learning). Набор методик для выявления взаимосвязей, т.е. ассоциативных правил, между переменными величинами в больших массивах данных. Используется в data mining.

Классификация (Classification). Набор методик, которые позволяет предсказать поведение потребителей в определенном сегменте рынка (принятие решений о покупке, отток, объем потребления и проч.). Используется в data mining.

Кластерный анализ (Cluster analysis). Статистический метод классификации объектов по группам за счет выявления наперед не известных общих признаков. Используется в data mining.

Краудсорсинг (Crowdsourcing). Методика сбора данных из большого количества источников.

Слияние и интеграция (Data fusion and data integration). Набор методик, который позволяет анализировать комментарии пользователей социальных сетей и сопоставлять с результатами продаж в режиме реального времени.

Интеллектуальный анализ данных (Data mining). Набор методик, который позволяет определить наиболее восприимчивые для продвигаемого продукта или услуги категории потребителей, выявить особенности наиболее успешных работников, предсказать поведенческую модель потребителей.

Ensemble learning. В этом методе задействуется множество предикативных моделей за счет чего повышается качество сделанных прогнозов.

Генетические алгоритмы (Genetic algorithms). В этой методике возможные решения представляют в виде `хромосом`, которые могут комбинироваться и мутировать. Как и в процессе естественной эволюции, выживает наиболее приспособленная особь.

Обработка естественного языка (Natural language processing (NLP)). Набор заимствованных из информатики и лингвистики методик распознавания естественного языка человека.

Методы анализа сетей (Network analysis). Набор методик анализа связей между узлами в сетях. Применительно к социальным сетям позволяет анализировать взаимосвязи между отдельными пользователями, компаниями, сообществами и т.п.

Прогнозное моделирование (Predictive modeling). Набор методик, которые позволяют создать математическую модель наперед заданного вероятного сценария развития событий. Например, анализ базы данных CRM-системы на предмет возможных условий, которые подтолкнут абоненты сменить провайдера.

Регрессия (Regression) Набор статистических методов для выявления закономерности между изменением зависимой переменной и одной или несколькими независимыми. Часто применяется для прогнозирования и предсказаний. Используется в data mining.

Распознавание сигнала (Signal processing). Заимствованный из радиотехники набор методик, который преследует цель распознавания сигнала на фоне шума и его дальнейшего анализа.

Анализ пространственных данных (Spatial analysis). Набор заимствованных из статистики методик анализа пространственных данных – топологии местности, географических координат, геометрии объектов. Источником больших данных в этом случае часто выступают геоинформационные системы (ГИС).

Анализ временных рядов (Time series analysis). Набор заимствованных из статистики и цифровой обработки сигналов методов анализа повторяющихся с течением времени последовательностей данных. Одни из очевидных применений – отслеживание рынка ценных бумаг или заболеваемости пациентов.

Визуализация (Visualization). Методы графического представления результатов анализа больших данных в виде диаграмм или анимированных изображений для упрощения интерпретации облегчения понимания полученных результатов.

1.5. МАШИННОЕ ОБУЧЕНИЕ

Целью машинного обучения является частичная или полная автоматизация решения сложных профессиональных задач в самых разных областях человеческой деятельности.

Машинное обучение (*Machine Learning*) – подраздел искусственного интеллекта, изучающий алгоритмы, способные к обобщению и обучению. Мы будем рассматривать методы машинного обучения основанные на анализе эмпирических данных. Часто такой тип обучения называют

обучение по прецедентам, или индуктивное обучение. Постановка задачи [11].

Дано конечное множество прецедентов (объектов, ситуаций), по каждому из которых собраны (измерены) некоторые данные. Данные о прецеденте называют также его описанием. Совокупность всех имеющихся описаний прецедентов называется обучающей выборкой. Требуется по этим частным данным выявить общие зависимости, закономерности, взаимосвязи, присущие не только этой конкретной выборке, но вообще всем прецедентам, в том числе тем, которые ещё не наблюдались.

Для решения задачи обучения по прецедентам определяется модель восстанавливаемой зависимости. Затем вводится функционал качества, значение которого показывает, насколько хорошо модель описывает наблюдаемые данные. *Алгоритм обучения* (*learning algorithm)* ищет такой набор параметров модели, при котором функционал качества на заданной обучающей выборке принимает оптимальное значение. Процесс *настройки* (fitting) модели по выборке данных в большинстве случаев сводится к применению численных методов оптимизации.

1.4 ПРИМЕРЫ ПРАКТИЧЕСКОГО ПРИМЕНЕНИЯ

1) Рассмотрим задачу, связанную с выявлением мошенничества при операциях с банковскими картами. Предположим, что у некоторого пользователя банка была украдена банковская карта, и злоумышленник снял деньги или расплатился по этой карте при приобретении товаров или услуг. В этом случае возникает вопрос: как определить, что данная транзакция была сделана мошенником, а не владельцем карты? Для решения подобной задачи используются данные о транзакциях пользователя банковской карты и формируется так называемый профиль пользователя. Профиль представляет собой набор записей о проведенных транзакций: дате, сумме, места, а также другой дополнительной информации позволяющий идентифицировать пользователя. С использованием алгоритмов машинного

обучения находится модель классификации, на основе которой может быть получена вероятностная оценка для идентификации транзакции. В случае превышения заданного порогового значения, система безопасности может заблокировать операции с картой.

2) Задача прогнозирования потребления электроэнергии. Затраты на оплату счетов за энергию составляют значительную статью бюджета как для предприятий, так и для отдельных домохозяйств. Потребление электроэнергии зависит от многих факторов: температурный режим, время суток (светлое/темное), погодные условия и т.д. Снизит затраты можно за счет прогнозирования потребления и разработки оптимального режима управления энергопотреблением. Первая задача на основе анализа и прогнозирования временных рядов и может быть решена с использованием методов машинного обучения.

3) Пример из области медицинской диагностики. На основании анализа большого массива данных содержащихся в историях болезней пациентов можно сформировать таблицу с описанием функциональных характеристик пациента и индикатором заболевания (выходные значения бинарного типа, TRUE – да, заболевание имеется (класс 1), FALSE – заболевание отсутствует(класс 2)). Решается задача классификации и формируется алгоритм для определения принадлежности значений входного вектора тому или иному классу, что позволяет оценить вероятность наличия того или иного заболевания.

4) Формирование рекомендаций товара пользователю. Задача сводится к выбору списка релевантных альтернатив и их ранжирования с целью максимизации некоторой функции “заинтересованности” пользователя товаром (например, ожидаемой прибыли). Специфика данной задачи заключается в наличии большого количества альтернатив (число рекомендуемых товаров измеряется десятками тысяч и более) и высоким уровнем неопределенности о предпочтениях пользователей (информация

либо отсутствует либо является не полной). Для решения данной задачи используют механизмы коллаборативной фильтрации и фильтрации на основе контента. Идея первого подхода заключается в том, что нужно найти похожих на текущего пользователя пользователей и рекомендовать пользователю то, что покупалось этими людьми. Идея второго подхода заключается в анализе контента и сопоставлении свойств товаров и параметров, характеризующих пользователя.

ГЛАВА 2. СРЕДСТВА АНАЛИЗА ДАННЫХ

Инструментальные средства для анализа данных являются одним из наиболее наукоемких видов программного обеспечения. По данным Международного статистического института, число наименований статистических программных продуктов приближается к тысяче. Развитие технологий работы с большими данными привело к появлению специализированных систем и новых языков программирования, ориентированных на работу с алгоритмами анализа данных и машинного обучения.

Одна из наиболее популярных в России система **STATISTICA.** Разработчиком системы является компания StatSoft Inc (http://www.statsoft.com/, Tulsa,Oklahoma, USA). В России StatSoft Inc представляет компания StatSoft Russia (http://www.statsoft.ru/company/) которая занимается локализацией программных продуктов *STATISTICA,* технической поддержкой пользователей, а также оказывает широкий спектр консалтинговых услуг. Один из последних проектов StatSoft Inc – аналитическая платформа Statistica Big Data Analytics. Подробное описание системы и многочисленные примеры представлены в книгах [12,13].

SAS (Statistical Analysis System) - интегрированная среда для построения прогнозных и описательных моделей, интеллектуального анализа данных, интеллектуального анализа текста, прогнозирования, оптимизации, имитационного моделирования, планирования экспериментов и пр. Разработчиком SAS является американская частная компания SAS institute (http://www.sas.com/en_us/company-information.html. На сегодняшний день продукты SAS это скорее приложения для Business Intelligence которые могут перенастраиваться для разных сфер и задач. В компании реализован интернет-сервис *SAS Studio*, а так же система on-line обучения и поддержки научных исследований.

Первая версия системы **IBM SPSS Statistics (Statistical Package for the Social Sciences)** появилась в 1968 году, затем этот пакет развивался в рамках Чикагского университета. В 2009 компания SPSS была приобретена IBM. IBM SPSS Statistics и продукты IBM SPSS Amos, Sample Power, VizDesigner, Data Collection, Collaboration and Deployment Services образуют модульный, полностью интегрированный программный комплекс, охватывающий все этапы процесса анализа данных. Сайт разработчика www-01.ibm.com/software/analytics/spss/. В книге А. Наследова [14] представлены разнообразные примеры реализации методов обработки и анализа данных с использованием SPSS. Кроме того, книга содержит подробную информацию о программе IBM SPSS AMOS, позволяющей использовать в исследованиях популярную и эффективную методологию моделирования структурными уравнениями (SEM - structural equation modeling).

Продукт **STADIA (Statistical Dialogue System)** является наиболее известной отечественной системой для статистического анализа. Система разработана в МГУ (http://protein.bio.msu.ru/~akula/Podr2~1.htm). По своим базовым возможностям STADIA сопоставима с большинством статистических пакетов, но отличается простотой и непретезательностью интерфейса, очень умеренными требованиями к ресурсам компьютера и денежным средствам пользователя. Пакет сопровождается учебником по прикладной статистике [15] с подробным руководством пользователя. 2009 г. STADIA включена в качестве одного из рекомендуемых программных средств в Государственный образовательный стандарт РФ.

Система **Poly Analyst** PolyAnalyst (разработчик компания Megaputer Intelligenc http://megaputer.ru/) предназначена для автоматического и полуавтоматического анализа числовых баз данных и извлечения из сырых данных практически полезных знаний. Система реализует полный цикл анализа данных, начиная с импорта и преобразования данных и заканчивая

отчетами. Модули PolyAnalyst называют Exploration engines. Они основаны на различных алгоритмах анализа данных. Версия PolyAnalyst 4.35 включает 14 Машин исследований, описание которых представлено в [16].

Система **Deductor** реализует практически все современные подходы к анализу структурированной табличной информации. Реализованные в Deductor технологии позволяют на базе единой архитектуры пройти все этапы построения аналитической системы: от создания хранилища данных до автоматического подбора моделей и визуализации полученных результатов. Первая версия продукта представлена в 2000 году, и с тех пор идет непрерывное развитие платформы на базе компании BaseGroup Labs (http://basegroup.ru/). Доступна свободно распространяемая (для некоммерческого использования) версия Deductor, функционирует электронный учебный центр (http://basegroup.ru/service/learning/lms) и современные курсы по бизнес-аналитике.

2.1 ВВЕДЕНИЕ В ЯЗЫК R

R – объектно-ориентированный язык программирования для статистической обработки данных и работы с графикой, а также свободная программная среда вычислений с открытым исходным кодом в рамках проекта GNU.

Язык **R** создавался как свободная реализация языка программирования S, который разрабатывался компанией Bell Labs с 1976 года. В августе 1993 г. Р.Джентльмен и Р.Ихак представили новую реализацию языка S, которую назвали **R.** От S-PLUS новый язык отличался некоторыми деталями, например, обращением с глобальными и локальными переменными, а также работой с памятью. В скором времени возникла распределенная система хранения и распространения пакетов к R, известная под аббревиатурой "CRAN"(Comprehensive R Archive Network – http://cran.r-project.org), основная идея организации которой – постоянное расширение,

коллективное тестирование и оперативное распространение прикладных средств обработки данных [17].

Почему стоит использовать язык **R** для анализа данных?

1) Распространяется по свободной лицензии GNU Public License;

2) **R** реализует преимуществ языка программирования высокого уровня, что позволяет одной строкой реализовать различные операции с объектами, векторами, матрицами, списками и т.д.;

3) Есть возможность сохранения всей истории вычислений для документирования.

4) Установочный пакет под Windows, Linux, MAC OS и др.

5) Наличие многочисленных исходных текстов, бинарных модулей расширений и пакетов. Базовые пакеты изначально присутствую в системе (base, *grDevices, cluster, nlme* и др.). Кроме того, можно установить любой из почти пяти тысяч доступных на CRAN пакетов. *R-пакет* представляет собой коллекцию наборов данных, функций языка R, документации и динамически загружаемых элементов на языке C или Fortran. R-пакет может быть установлен как группа, которая будет доступна в рамках сеанса R.

6) Развитая система поддержки, включающая обновление компонентов среды, интерактивную помощь и различные образовательные ресурсы, предназначенные как для начального изучения R, так и последующих консультаций [18-19].

7) Возможность работы с данными, размещенными в файловой системе Hadoop HDFS и СУБД Hbase.

R доступен в нескольких формах: исходный текст программ, написанный на C (и некоторые подпрограммы в Fortran77) и в откомпилированном виде. Для работы с **R** необходимо установить интерпретатор языка, который можно скачать на сайте проекта (http://cran.r-project.org) или русского "зеркала" http://cran.gis-lab.info. Стандартный графический пользовательский интерфейс R (RGui) включает окно

редактирования скриптов и всплывающие окна с графической информацией (рисунками, диаграммами и т.д.). Можно использовать различные интегрированные среды, например RStudio (https://www.rstudio.com/). В составе дистрибутива R имеется техническая документация с подробным описанием языка и руководством по использованию (русскоязычная версия представлена в книге [20].

Создаваемые скрипты необходимо сохранять в текстовом файле с расширением *.r . Сделать это можно, например, с помощью команды

> savehistory(file="myscript.r")

Выполнить последовательность команд скрипта можно из пункта меню "Правка > Запустить все". **R** имеет собственные встроенные редакторы скриптов, можно также использовать специализированный редактор Tinn-R (www.sciviews.org/Tinn-R/). Загрузить скрипт в R можно с помощью команды *source("myscript.r", echo=TRUE)*. Опция echo нужна для вывода команд скрипта.

Основной тип данных языка – массивы. Массивы в R выступают как контейнеры (упорядоченные наборы однородных данных), а не математические вектора (элементы векторного пространства). Второй важнейший контейнер – списки, упорядоченные наборы неоднородных данных, некоторые из которых могут быть именованными. Скаляры представляют собой векторы длины 1.

R – объектно-ориентированный язык: переменные, данные, матрицы, функции, результаты, и т.д., хранятся в оперативной памяти компьютера в форме объектов, которые имеют имя.

Выделяют два основных типа объектов [17]:

1. **Объекты, предназначенные для хранения данных** ("data objects") – это отдельные переменные, векторы, матрицы и массивы, списки, факторы, таблицы данных;

2. **Функции** ("*function objects*") – это поименованные программы, предназначенные для создания новых объектов или выполнения определенных действий над ними.

Объекты среды **R**, предназначенные для коллективного и свободного использования, комплектуются в пакеты, объединяемые сходной тематикой или методами обработки данных. Пакеты инсталлируются в определенной директории операционной системы или, в неустановленном виде, могут храниться и распространяться в архивных файлах. Полная информация о пакете (версия, основное тематическое направление, авторы, даты изменений, лицензии, другие функционально связанные пакеты, полный список функций с указанием на их назначение и проч.) может быть получена командой

library(help=<имя_пакета>)/

Для установки пакета можно зайти на сайт http://cran.gis-lab.info/web/packages, выбрать и скачать нужный пакет. В этом случае можно предварительно изучит всю информацию по пакету, в частности, описание входящих в него функций. Далее можно воспользоваться пунктом командного меню "*Пакеты > Установить пакеты из локальных zip-файлов*".

При запуске консоли RGui загружаются только некоторые базовые пакеты. Для инициализации любого другого пакета перед непосредственным использованием его функций нужно ввести команду *library* (<*имя_пакета*>).

Список наиболее востребованных пакетов приведен в книге [17].

Работая непосредственно в системе **R**, можно создать небольшие по объему объекты для хранения данных (векторы, матрицы, списки, таблицы данных). Подлежащие анализу объемные таблицы данных обычно подготавливаются при помощи сторонних приложений, и только потом загружаются в рабочую среду R из внешних файлов. Предпочтение при этом

отдается текстовым файлам, но есть возможность импортировать таблицы, сохраненные во множестве других распространенных форматов (Excel, SPSS, SAS, STATA, Access, Matlab, SQL, Oracle, и т.п.) с помощью пакета foreign.

Правила подготовки загружаемых файлов [17]:

1) В импортируемой таблице с данными не должно быть пустых ячеек. Еслинекоторые значения по тем или иным причинам отсутствуют, вместо них следует ввести NA.

2) Импортируемую таблицу с данными рекомендуется преобразовать в простой текстовый файл с одним из допустимых расширений. На практике обычно используются файлы с расширением .txt, в которых значения переменных разделены знаками табуляции (*tab-delimited files*), а также файлы с расширением .csv (*comma separated values*), в которых значения переменных разделены запятыми или другим разделяющим символом.

3) В качестве первой строки в импортируемой таблице рекомендуется ввести заголовки столбцов-переменных.

Основной функцией для импортирования данных в рабочую среду R является *read.table().*

Сохранить результаты работы можно несколькими способами:

sink(file= <имя файла>) – выводит результаты выполнения последующих команд в режиме реального времени в файл с заданным именем; для прекращения действия этой команды необходимо выполнить команду sink() без параметров;

save(file= <имя файла>, <список сохраняемых объектов>) – сохраняет указанные объекты в двоичном файле XDR-формата, с которым можно работать в любой операционной системе;

load(file= <имя файла>) – восстанавливает сохраненные объекты в текущей среде;

save.image(file= <имя файла>) – сохраняет все объекты, созданные в ходе работы, в виде специфичного для R rda-файла.

Так же можно генерировать пиксельное изображение и сохранять полученные графические окна в файлах разного формата.

Обработка данных в **R** реализуется с помощью функций. Как правило, функции возвращают результат своего выполнения в виде объекта языка **R** –переменной определенного класса: вектора, списка, таблицы и т.д. Подробное описание функций представлено на сайте проекта **R** (https://cran.r-project.org/doc/contrib/Short-refcard.pdf), с русифицированной версией обзора основных функций R снабженной ссылками на online-мануал можно ознакомиться на сайте http://aakinshin.net/ru/blog/r/functions/.

В **R** реализована возможность векторизации вычислений. Можно обрабатывать одновременно весь массив (вектор) целиком или по несколько элементов вектора в каждый момент времени, что обеспечивает значительное ускорение однотипных вычислений над большими массивами данных.

Функционал R доступен из языка программирования Python при помощи пакета RPy. В статистических пакетах SPSS (начиная с версии 16.0) и Statistica (начиная с версии 9.0) появилась поддержка функций R.

2.2 ПРИМЕРЫ ПРАКТИЧЕСКОГО ПРИМЕНЕНИЯ

Рассмотрим пример загрузки и предварительного анализа данных. R позволяет загружать данные из файла (текстового формата, CSV), физически расположенного на диске или из ресурса размещенного в интернете:

```
> upload_data_from_file  = read.table("path\\data.csv", header = T,sep = ',', dec = ",")
> upload_data_from_www  = read.table("http://resouce.com/data.txt").
```

Можно так же загружать наборы данных, встроенные в пакеты CRAN

```
> library(MASS)
> data(phones)
> upload_data_from_package = phones
```

Загрузка в первых двух случаях производится командой read.table. Для нее указывается путь к файлу и параметры, определяющие загрузку (имеют ли данные строку заголовок, какой знак используется в качестве разделителя, разделитель целой и дробной частей).

Для отображения содержимого в переменных upload_data_from_file , upload_data_from_www , upload_data_from_package можно набрать эти переменные и RStudio выдаст записи (для оптимизации R выводит ограниченное число записей). Например, переменная будет содержать набор данных о годовом числе звонков с 1950 по 1973 годы в Бельгии.

```
> upload_data_from_package
$year
[1] 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70
71 72 73
$calls
[1]   4.4   4.7   4.7   5.9   6.6   7.3   8.1   8.8  10.6  12.0  13.
5  14.9  16.1  21.2 119.0 124.0 142.0 159.0 182.0
[20] 212.0  43.0  24.0  27.0  29.0
```

Для доступа к записям набора данных, соответствующим определенному столбцу используется знак доллара ($). Например, команда

```
> upload_data_from_package$calls
```

выводит все записи для столбца с именем ‘calls’. В R команда names отображает имена столбов в наборе данных:

```
> names(upload_data_from_package)
[1] "year"  "calls"
```

Кроме этого можно использовать команду length для получения размерности набора данных (число строк и столбцов):

```
> length(upload_data_from_package)
[1] 2
> length(upload_data_from_package$calls)
[1] 24
```

Для отображения нескольких первых или нескольких последних записей в наборе данных используются команды head и tail соответственно,

```
> head(upload_data_from_package$calls, 5)
[1] 4.4 4.7 4.7 5.9 6.6
> tail(upload_data_from_package$calls, 5)
[1] 212  43  24  27  29
```

где 5 – число выводимых записей (регулируется пользователем).

Выборку, соответствующую записям по определенному столбцу можно присваивать для удобства переменной (в примере `calls`).

```
> calls = upload_data_from_package$calls
> calls
[1]   4.4   4.7   4.7   5.9   6.6   7.3   8.1   8.8  10.6  12.0  13.
5  14.9  16.1  21.2 119.0 124.0 142.0 159.0 182.0
[20] 212.0  43.0  24.0  27.0  29.0
```

В R существует возможность реализовывать проверку выполнения условий для набора данных. В результате получается бинарный вектор, где значение TRUE характеризует выполнение условия, а FALSE – нет.

```
> calls_more_150 =  calls > 150
> calls_more_150
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
[20]  TRUE FALSE FALSE FALSE FALSE
```

Для получения описательных статистик (минимум, максимум, среднее, дисперсию, медиану) используются команды над наборами данных: min, max, mean, var, median.

```
> max(calls)
[1] 212
> min(calls)
[1] 4.4
```

```
> max(calls)
[1] 212
> mean(calls)
[1] 49.99167
> var(calls)
[1] 4294.457
> median(calls)
[1] 15.5
```

Команда summary выдает таблицу с описательными статистиками

```
> summary(calls)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  4.40    7.90   15.50   49.99   62.00  212.00
```

R имеет мощный механизм формирования наборов данных (подмножеств) из основного набора.

```
> # Выбор первых восьми элементов
> first8 = calls[1:8]
> first8
[1] 4.4 4.7 4.7 5.9 6.6 7.3 8.1 8.8
> # Выбор записей в которых число звоноков превышает 100
> calls_gt100 =  subset(calls, calls > 100)
> calls_gt100
[1] 119 124 142 159 182 212
```

ГЛАВА 3. ВИЗУАЛИЗАЦИЯ ДАННЫХ

Визуализация данных – это представление данных в виде, который обеспечивает наиболее эффективную работу человека по их изучению [4].

Визуализация данных частично решает проблему сложности восприятия данных и может быть отнесена к технологии анализа данных. Визуализация традиционно рассматривалась как вспомогательное средство при анализе данных, однако сейчас все больше исследований говорит о ее самостоятельной роли.

Подсистема визуализации данных является важной составной частью качественных систем анализа данных и может использоваться на всех этапах процесса обработки данных [4]:

- визуализация исходных данных. Этот этап полезен для оценки степени соответствия ожиданиям и пригодности данных к анализу, выдвижения гипотез о закономерностях и необходимых процедурах первичной обработки;
- визуализация выборки, загруженной в систему обработки;
- визуализация результатов первичной обработки;
- визуализация промежуточных результатов;
- визуализация окончательных результатов.

С помощью визуализации данных решаются самые разные задачи:

- представление информации в наглядном виде;
- отображение закономерностей, присущих исходному набору данных;
- снижение размерности;
- анализ качества данных (шумы, выбросы, пробелы);
- иллюстрация вида модели для анализа данных (структура нейронной сети и пр.);
- интерпретация полученных результатов.

Разные типы данных требуют разных методов визуализации.

3.1 МЕТОДЫ ВИЗУАЛИЗАЦИИ

Метод визуализации – системное, основанное на правилах, динамическое и/или статическое графическое представление информации, помогающее разобраться в сложных понятиях, нацеленное на обобщение.

В 2007 году Ленглер и Эпплер разработали периодическую таблицу, классифицирующую 100 различных способов визуализации данных, стараясь применить те же принципы, что лежат в основе периодической системы химических элементов Менделеева. Элементы в таблице – способы визуализации данных, распределены по группам и периодам в зависимости от целей, для которых вы выбираете тот или иной способ визуализации данных, и в зависимости от сложности способа. Также элемента распределены по цветам в зависимости от типа визуализации, который вы хотите использовать. Интерактивная таблица размещена на сайте http://www.visual-literacy.org/periodic_table/periodic_table.html.

При наведении курсора мыши на ячейку, "всплывают" примеры для данного метода визуализации. Некоторые из представленных там методов, относящиеся непосредственно к анализу данных, перечислены ниже.

1. Линейный график (line chart, area chart)
2. График рассеивания (scatterplot)
3. Столбиковая диаграмма bar chart
4. Гистограмма histogram
5. Круговая диаграмма pie chart
6. Площадная диаграмма bubble chart
7. Дерево tree
8. Диаграмма Венна-Эйлера Venn/Euler diagram
9. Картограмма cartogram
10. Дендрограмма dendrogram

Наряду с средствами визуализации встроенными в системы анализа данных, существуют специальные сервисы, ориентированные исключительно на визуализацию Many Eyes (http://www.easel.ly), Google Chart Tool (https://developers.google.com/chart/) и др.

Дополнительного внимания заслуживают средства визуализации которые используют при анализе пространственных данных. Пространственные данные представляют собой данные о географических объектах, об их местоположении и свойствах. Визуальное представление пространственных объектов показывает их взаимное расположение и позволяет проводить анализ разнообразных пространственных отношений. Цифровое описание пространственного объекта включает в себя задание координат объекта наблюдения и описание его атрибутов.

Для отображения результатов анализа данных в картографических сервисах используют самые разнообразные способы визуализации, например:

Способ размерных символов (значков) – анализируемые характеристики объектов отображаются специальными символами, размер которых передаёт количественную информацию, а форма и цвет качественную информацию.

Способ качественного или (количественного фона) – в этом случае группируются данные с близкими значениями и созданным группам присваиваются определенные цвета, типы символов или линий.

Точечный способ – изобразительным средством является множество точек одинакового размера, каждая из которых имеет определенное значение количественного показателя.

Локализованные диаграммы – позволяют отобразить соотношение нескольких характеристик, при этом диаграммы имеют географическую

привязку (например, в точке размещения поста наблюдений показывают соотношение загрязняющих веществ).

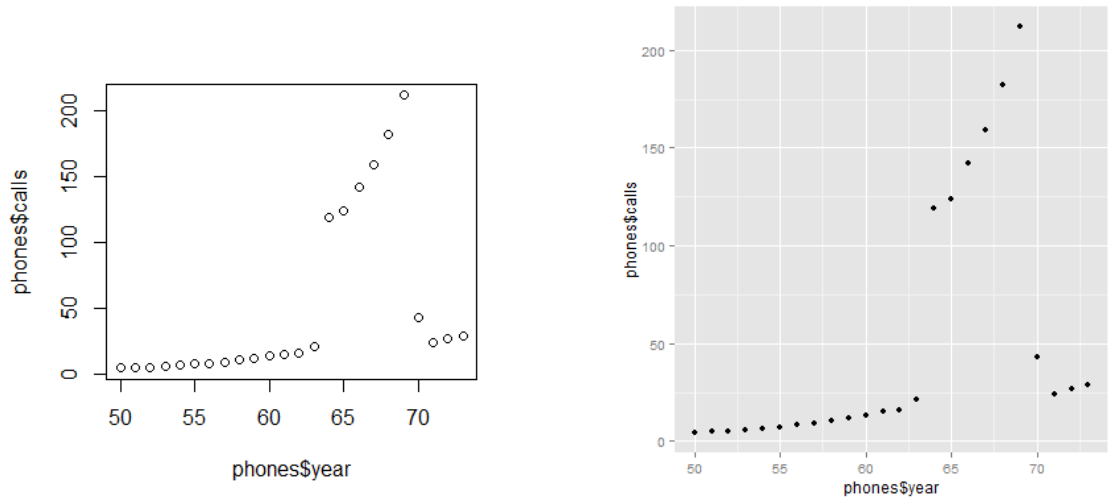
Средства реализации методов визуализации подробно описаны в книгах [17,20,21]. На сайте R Graph Gallery (http://rgraphgallery.blogspot.ru/) можно ознакомиться с различными примерами визуализации с исходным R-кодом.

3.2 ПРИМЕРЫ ПРАКТИЧЕСКОГО ПРИМЕНЕНИЯ

1) Построение точечной диаграммы осуществляется командой `plot`.

```
> library(ggplot2)
> plot(phones$year, phones$calls)
> qplot(phones$year, phones$calls)
```

Результаты применения команд представлены на рисунке 1 а) и б).



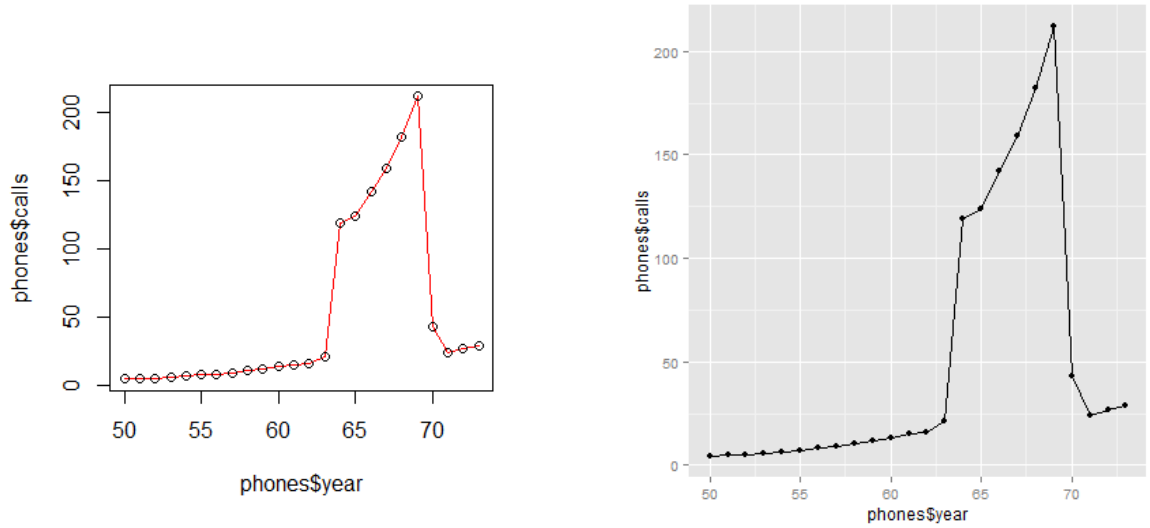
а) б)

Рис 1. Результаты визуализации с использованием команды а) plot; б) qplot

Для построения линейного графика, требуется указать параметр `type="l"`. Параметр `col` задает цвет линии.

```
> plot(phones$year, phones$calls, type="l", col="red")
> points(phones$year, phones$calls)
> qplot(phones$year, phones$calls, geom=c("line","point"))
```

Следует обратить внимание, что тип графика в команде `qplot` задается с помощью параметра `geom`. Результаты визуализации представлены на рисунке 2.



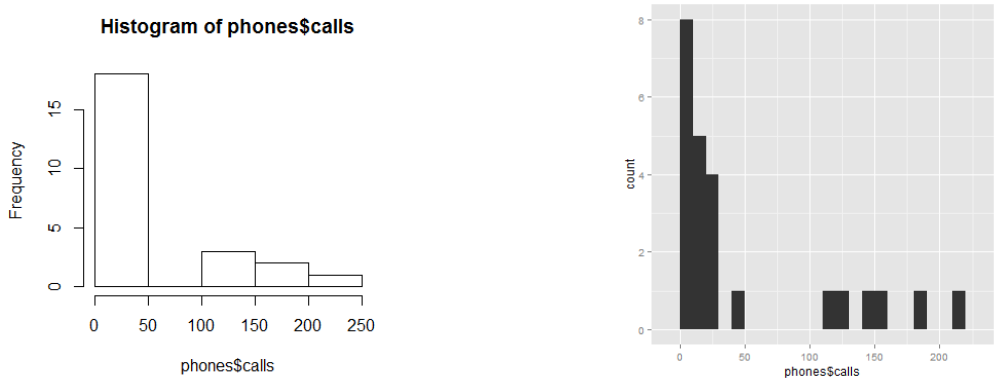
а) б)

Рис 2. Результаты визуализации линейного графика с использованием команды а) plot; б) qplot

2) В R можно построить гистрограмму с использованием команды `hist`. Параметр `breaks` определяет число столбцов в гистограмме.

```
> hist(phones$calls, breaks = 5)
> qplot(phones$calls, binwidth=10)
```

Результаты представлены на рисунке 3.



а) б)

Рис 3. Результаты построения гистограммы

3) Способ визуализации “ящик с усами” (box plot) очень распространен для получения общей информации о переменной: о среднем, медиане, минимальном и максимальном значении, а также о квантилях.

```
> boxplot(phones$calls)
> qplot(phones$year,phones$calls, geom="boxplot")
```

Результат визуализации “ящик с усами” представлен на рисунке 4

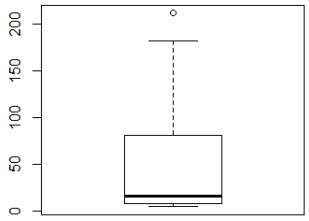


Рис 4. Результат визуализации «ящик с усами»

Следует отметить, что R позволяет сохранить полученные рисунке во внешний файл. Например команда ‘`pdf’` сохраняет файл в формате pdf по указанному пути.

```
> pdf("path\\exp1_res.pdf",width=5,height=5)
> plot(exp1_data$Strategy, type="n", xlab="time", ylab = "Strategy")
> dev.off()
```

4. КЛАСТЕРИЗАЦИЯ ДАННЫХ

Кластерный анализ – это метод классификационного анализа; его основное назначение – разбиение множества исследуемых объектов и признаков на однородные в некотором смысле группы, или кластеры. Это многомерный статистический метод, поэтому предполагается, что исходные данные могут быть значительного объема. Это относится как к количеству объектов исследования, так и к признакам их характеризующих. Впервые термин кластерный анализ, появился в работе Триона (Tryon) в 1939 году. Активный интерес к данной теме пришёлся на период 60-80 гг. Импульсом для разработки многих кластерных методов послужила книга «Начала численной таксономии», опубликованная в 1963 году биологами Р. Сокэломи и П. Снитом.

Кластерный анализ в отличие от большинства математико-статистических методов не накладывает никаких ограничений на вид рассматриваемых объектов и позволяет исследовать множество исходных данных практически произвольной природы.

Задача кластеризации заключается в следующем [22].

Имеется обучающая выборка $X_\ell = \{x_1, \ldots, x_\ell\} \subset X$ и функция расстояния между объектами $\rho(x, x')$. Требуется разбить выборку на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из объектов, близких по метрике ρ, а объекты разных кластеров существенно отличались. При этом каждому объекту $x_i \in X_\ell$ приписывается метка (номер) кластера y_i.

Кластер – это совокупность элементов, которые являются однородными или похожими в данной системе признаков.

Цели кластерного анализа [23]:

а) структуризация (представление общей структуры данных);

б) описание кластеров в терминах тех или иных признаков;

в) установление взаимосвязи между различными аспектами явлений;

г) формирование обобщающих утверждений о свойствах данных и явлений;
д) визуализация данных в процессах принятия решений.

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

Решение задачи кластеризации принципиально неоднозначно в силу следующих причин:

- не существует однозначно наилучшего критерия качества кластеризации;
- число кластеров, как правило, неизвестно заранее и устанавливается в соответствии с некоторым субъективным критерием;
- результат кластеризации существенно зависит от метрики, выбор которой также субъективен и определяется экспертом.

Этапы кластерного анализа:

- подготовка данных;
- определение обучающей выборки;
- выбор алгоритма
- выбор метрики;
- реализация алгоритма;
- оценка качества кластеризации.

Характеристики кластера.

Центр кластера (*cluster centroid)* – это среднее геометрическое место точек в пространстве переменных.

Радиус кластера (*cluster radius)* – максимальное расстояние точек от центра кластера.

Размер кластера (*cluster size*) – оценочная характеристика. Может быть определен либо по радиусу кластера, либо по среднеквадратичному отклонению объектов для этого кластера.

Объект относится к кластеру, если расстояние от объекта до центра кластера меньше радиуса кластера. Если это условие выполняется для двух и более кластеров, объект является спорным (может быть отнесен к нескольким кластерам)

Метрики

Понятие метрики (расстояние между объектами) является интегральной мерой сходства объектов между собой. Выбор метрики зависит от условий задачи и влияет на результат решения.

Метрикой будем называть величину d_{ij} , которая удовлетворяет следующим аксиомам:

1. $d_{ij} > 0$ (неотрицательность расстояния)

2. $d_{ij} = d_{ji}$ (симметрия)

3. $d_{ij} + d_{jk} > d_{ik}$ (неравенство треугольника)

4. Если d_{ij} не равно 0, то i не равно j (различимость нетождественных объектов)

5. Если $d_{ij} = 0$, то i = j (неразличимость тождественных объектов)

Меру близости (сходства) объектов удобно представить как обратную величину от расстояния между объектами. В многочисленных изданиях посвященных кластерному анализу описано более 50 различных способов вычисления расстояния между объектами. Рассмотрим некоторые из них.

Евклидово расстояние – наиболее распространенная метрика, является геометрическим расстоянием в многомерном пространстве.

Квадрат евклидова расстояния. Используется, когда необходимо придать большие веса более отдаленным друг от друга объектам.

Манхэттенское расстояние – сумма модулей разностей координат

точек. Для этой меры влияние отдельных больших разностей (выбросов) уменьшается (так как они не возводятся в квадрат).

Расстояние Чебышева – максимум модуля разностей координат точек. Это расстояние используют когда необходимо определить два объекта как «различные», если они различаются по какой-либо одной координате.

Метрика Хэмминга (процент несогласия) используется для кластеризации категориальных данных. Определяется как число различных позиций у рассматриваемых объектов.

4.1.ОБЗОР МЕТОДОВ КЛАСТЕРИЗАЦИИ

Иерархический подход

Объединяет методы кластеризации, характеризующиеся построением иерархической или древовидной структуры. В качестве условия остановки в иерархических методах используют пороговое число кластеров, которое необходимо получить или пороговое значение расстояния между кластерами. Основная проблема иерархических методов заключается в сложности определения условия остановки таким образом, чтобы выделить «естественные» кластеры и в то же время не допустить их разбиения. Еще одна проблема иерархических методов кластеризации заключается в неоднозначности выбора точки разделения или слияния кластеров.

Агломеративные методы (Agglomerative (boon up)).

Каждый объект первоначально находится в отдельном кластере. Кластеры формируют, группируя объекты каждый раз во все более и более крупные кластеры.

Метод ближнего соседа (Nearest neighbor, NN). Здесь расстояние между двумя кластерами определяется расстоянием между двумя наиболее близкими объектами (ближайшими соседями) в различных кластерах. Этот метод позволяет выделять кластеры сколь угодно сложной формы при

условии, что различные части таких кластеров соединены цепочками близких друг к другу элементов.

Алгоритм:

1. Составление матрицы попарных расстояний между объектами. Каждому объекту назначается свой кластер;
2. Нахождение в матрице наименьшего элемента (то есть наименьшего расстояния между соседями);
3. Объединение кластеров, в которые входят объекты, имеющие наименьшее расстояние.
4. Проверка: сколько осталось кластеров. Если один, то завершить алгоритм. Если два и более, то перейти к шагу 1.

Обобщение метода является подход, основанный на выборе k ближайших соседей (k-nearest neighbors algorithm, kNN). В этом случае находится k самых близких объектов и определяется расстояние до каждой из них. Существуют многочисленные модификации этого алгоритма, которые позволяют его подстроить под конкретную задачу за счет использования наиболее подходящих функций сочетания и метрик.

Метод Уорда (Ward's method). Данный метод построен таким образом, чтобы оптимизировать минимальную дисперсию внутри кластеров. Эта целевая функция известна как внутригрупповая сумма квадратов или сумма квадратов отклонений. В качестве расстояния между кластерами берется прирост суммы квадратов расстояний объектов до центров кластеров, получаемый в результате их объединения. На каждом шаге алгоритма объединяются такие два кластера, которые приводят к минимальному увеличению целевой функции. Этот метод направлен на объединение близко расположенных кластеров и "стремится" создавать кластеры малого размера.

Алгоритм

1. Создать первичный набор кластеров, содержащих по одному элементу из подлежащего кластеризации набора.

2. Для всех возможных пар кластеров посчитать целевую функцию, взять лучшее значение и запомнить, для какой пары кластеров лучшее значение получено.

3. Объединить пару кластеров с лучшим значением целевой функции.

4. Если число кластеров больше двух перейти к шагу 2. Иначе завершить алгоритм.

Метод CURE (Clustering Using REpresentatives). Выполняет иерархическую кластеризацию с использованием набора определяющих точек для помещения объекта в кластер.

Алгоритм:

1. Построение дерева кластеров, состоящего из каждой строки входного набора данных.

2. Формирование «кучи» в оперативной памяти, расчет расстояния до ближайшего кластера (строки данных) для каждого кластера. При формировании кучи кластеры сортируются по возрастанию дистанции от кластера до ближайшего кластера. Расстояние между кластерами определяется по двум ближайшим элементам из соседних кластеров. Для определения расстояния между кластерами используются «манхеттенская», «евклидова» метрики или похожие на них функции.

3. Слияние ближних кластеров в один кластер. Новый кластер получает все точки входящих в него входных данных. Расчет расстояния до остальных кластеров для новообразованного кластера. Для расчета расстояния кластеры делятся на две группы: первая группа – кластеры, у которых ближайшими кластерами считаются кластеры, входящие в новообразованный кластер, остальные кластеры – вторая группа. И при этом для кластеров из первой группы, если расстояние до новообразованного

кластера меньше чем до предыдущего ближайшего кластера, то ближайший кластер меняется на новообразованный кластер. В противном случае ищется новый ближайший кластер, но при этом не берутся кластеры, расстояния до которых больше, чем до новообразованного кластера. Для кластеров второй группы выполняется следующее: если расстояние до новообразованного кластера ближе, чем предыдущий ближайший кластер, то ближайший кластер меняется. В противном случае ничего не происходит.

4. Переход на шаг 3, если не получено требуемое количество кластеров.

Дивизимные (Divisive (top down).

Изначально все объекты принадлежат одному кластеру, который на последующих шагах делится на меньшие кластеры, в результате образуется последовательность расщепляющих групп.

Один из первых дивизимных алгоритмов был предложен С.Макнаотоном в 1965 году.

Алгоритм

1. Все элементы помещаются в один кластер C_1

2. Выбирается элемент, у которого среднее значение расстояния от других элементов в этом кластере наибольшее.

3. Выбранный элемент удаляется из кластера C_1и формирует первый член второго кластера C_2

4. На каждом последующем шаге элемент в кластере C_1, для которого разница между средним расстоянием до элементов, находящихся в C_2, и средним расстоянием до элементов, остающихся в C_1, наибольшая, переносится в C_2

5. Выполнять 4. До тех пор, пока соответствующие разницы средних не станут отрицательными, т.е. пока существуют элементы, расположенные к элементам кластера C_2 ближе, чем к элементам кластера C_1.

Неиерархический подход

Объединяет методы, которые определяют центр кластера и группируют все объекты в пределах заданного от центра порогового значения.

Неиерархические методы менее чувствительны к шумам и выбросам, некорректному выбору метрики, включению незначимых переменных в набор, участвующий в кластеризации и пр. Недостатком этих методов является необходимость заранее определять параметры кластеризации (количество кластеров, итераций, правило остановки и пр.). Однако в некоторых случаях, например при анализе данных большого объема эти методы наиболее предпочтительны.

Большинство алгоритмов этого типа реализуют следующую последовательность действий:

1. Начать с исходного разбиения данных на некоторое заданное число кластеров; вычислить центры тяжести этих кластеров.

2. Поместить каждую точку данных в кластер с ближайшим центром тяжести.

3. Вычислить новые центры тяжести кластеров; кластеры не заменяются новыми до тех пор, пока не будут просмотрены полностью все данные.

4. Шаги 2 и 3 повторяются до тех пор, пока не перестанут меняться кластеры.

Метод k-средних (k-means) [24]. Предложен Г. Штейнгаузом в 1956 г. Пожалуй, самый известный из методов кластеризации. Используется для анализа данных большого объема. Метод очень чувствителен к шуму и обособленным точкам пространства, поскольку даже малое количество таких точек может существенно влиять на вычисление центра масс кластера.

Алгоритм:

1. Задание числа кластеров k, на которые надо разбить входные данные.

2. Выбор k точек в исходном пространстве, именуемые как центры кластеров. На практике в основном выбирают случайные точек исходных данных.

3. Для каждой точки входных данных находится ближайший центр кластера, называемые точками соответствующего центра кластера. Группа точек одного центра кластера называются соответствующим кластером.

4. Вычисление для каждого кластера С его центра масс, и назначение центра кластера.

5. Повтор процедуры с 3 шага в том случае, если не выполняется критерий остановки, например, неизменяющиеся кластеры за несколько последних итераций.

Практическое применение k-means связано с рядом проблем, обусловленных, прежде всего с необходимостью заранее задавать количество кластеров. Кроме того, алгоритм очень чувствителен к выбору начальных центров кластеров и не может идентифицировать объекты, находящиеся на границе кластеров.

Модификации k-means

Существуют многочисленные модификации алгоритма k-means, например, для работы с категорийными атрибутами – ***k-modes*** [26, 27] и смешанными атрибутами (*k-prototypes* [28]). Еще одна модификация k-means алгоритм *Farthest First* [29]. Его особенностью его является изначальный выбор центроидов по принципу удаленности от остальных.

Замена локальной функции оптимизации на глобальную обеспечивает возможность добавления новой точки в кластер без пересчета расстояний. В этом случае рассчитываются только так называемые кластерные характеристики (clusters features). Это существенно снижает вычислительные затраты. Эта идея позволила сосздать целый класс

модификаций Масштабируемые аналоги scalable k-means [30], BIRCH [31], LargeItem [32, 34], CLOPE [33, 34] и др.

Нечеткая кластеризация (Fuzzy Classifier Means (FCM) или c-means) [25]. Данный метод так же является модификацией k-means для случая, когда векторы данных можно отнести к нескольким кластерам одновременно с некоторой степенью принадлежности. Степень принадлежности определяется расстоянием от объекта до соответствующих кластерных центров. Данный алгоритм итерационно вычисляет центры кластеров и новые степени принадлежности объектов.

Алгоритм.

1. Выбрать начальное нечеткое разбиение n объектов на k кластеров путем выбора матрицы принадлежности U размера n x k.

2. Используя матрицу U, найти значение критерия нечеткой ошибки:

$$E^2(X,U) = \sum_{i=1}^{N}\sum_{k=1}^{K} U_{ik} \| x_i^{(k)} - c_k \|^2 , \quad (4.1)$$

где c_k – «центр масс» нечеткого кластера k:

$$c_k = \sum_{i=1}^{N} U_k x_i . \quad (4.2)$$

3. Перегруппировать объекты с целью уменьшения этого значения критерия нечеткой ошибки.

4. Возвращаться в п. 2 до тех пор, пока изменения матрицы U не станут незначительными.

4.2 ОЦЕНКА КАЧЕСТВА КЛАСТЕРИЗАЦИИ

Надежности и достоверность полученных в результате кластеризации решений должна быть оценена. Для этого существуют формальные и неформальные методы. Формальные методы зависят от того метода, который использовался для кластеризации, как правило, их применение

связано с реализацией сложных и трудоемких процедур. На практике можно воспользоваться следующими приемами:

- выполните процедуру кластерного анализа несколько раз с использованием различных способов измерения расстояния. Сравните полученные результаты и оцените степень совпадения результатов; используйте разные методы кластерного анализа и сравните полученные результаты;
- разбейте данные на две равные части случайным образом. Выполните кластерный анализ отдельно для каждой половины. Сравните кластерные центроиды двух подвыборок;
- случайным образом удалите некоторые переменные. Выполните кластерный анализ по сокращенному набору переменных. Сравните результаты с теми, которые получены на основе полного набора переменных.

4.3 ПРИМЕРЫ ПРАКТИЧЕСКОГО ПРИМЕНЕНИЯ

Рассмотрим пример решения задачи кластеризации для типового набора данных: ирисы Фишера. Набор данных доступен по адресу https://archive.ics.uci.edu/ml/machine-learning-databases/iris/.

Набор данных содержит 150 различных цветов ириса, для которых измерены в сантиметрах: длина чашелистника (англ. sepal length), ширина чашелистника(англ. sepal width), длина лепестка (англ. petal length) и ширина лепестка (англ. petal width). Для каждого экземпляра определен класс: ирис щетинистый (Iris setosa), ирис виргинский (Iris virginica) и ирис разноцветный (Iris versicolor). Задача сводится к кластеризации записей (отнесения к тому или иному кластеру). Загрузим данные из файла, полученные из репозитория и зададим названия столбцов:

```
> iris_data = read.csv("c:\\DataLake\\uci\\iris.data", header=FALSE)
```

```
> colnames(iris_data) <- c("sepal.length", "sepal.width",
"petal.length", "petal.width", "class")
> names(iris_data)
[1] "sepal.length" "sepal.width" "petal.length"
"petal.width" "class"
```

Создадим набор данных из которого исключим столбец, характеризующий класс (class).

```
> iris_data.input <- iris_data
> iris_data.input$class <- NULL
> head(iris_data.input)
 sepal.length sepal.width petal.length petal.width
1          5.1         3.5          1.4         0.2
2          4.9         3.0          1.4         0.2
3          4.7         3.2          1.3         0.2
4          4.6         3.1          1.5         0.2
5          5.0         3.6          1.4         0.2
6          5.4         3.9          1.7         0.4
```

Выполним кластеризацию данных с использованием алгоритма k-средних (k-means). Для алгоритма k-средних требуется определить число кластеров, в данном случае нам априорно оно известно и равно 3.

```
clustered_data <- kmeans(iris_data.input, 3)
> clustered_data
K-means clustering with 3 clusters of sizes 50, 62, 38
```

Для анализа результатов необходимо воспользоваться доступными свойствами переменной clustered_data.

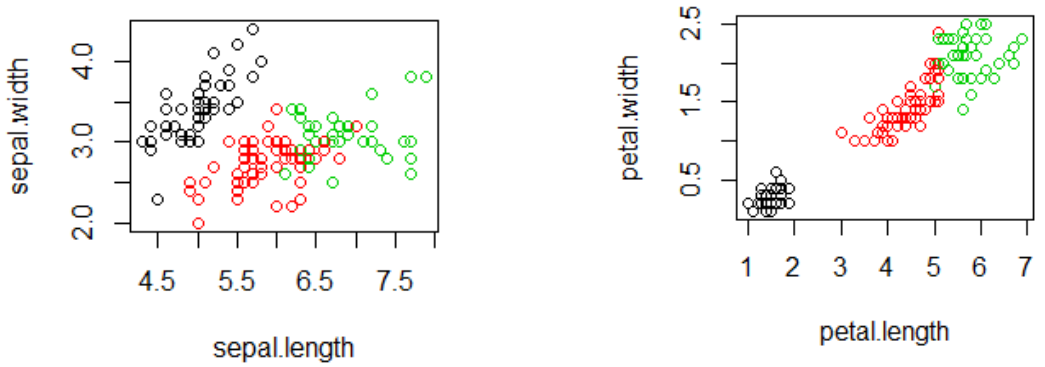
```
Cluster means:
 sepal.length sepal.width petal.length petal.width
1     5.006000    3.418000     1.464000    0.244000
2     5.901613    2.748387     4.393548    1.433871
3     6.850000    3.073684     5.742105    2.071053
Clustering vector:
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 3 2 2 2
[57] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 3 3 3 2 3 3 3 3
[113] 3 2 2 3 3 3 3 2 3 2 3 2 3 3 2 2 3 3 3 3 3 2 3 3 3 3 2 3 3 3 2 3 3 3 2 3 3 2
Within cluster sum of squares by cluster:
[1] 15.24040 39.82097 23.87947
(between_SS / total_SS =  88.4 %)
Available components:
```

```
[1]
"cluster"      "centers"      "totss"        "withinss"     "tot.wit
hinss" "betweenss"    "size"
[8] "iter"         "ifault"
```

В результате можно визуально оценить результаты кластеризации, построив следующие графики (см рис 5) на которых цветом обозначены различные кластеры.

```
> table(iris_data$class, clustered_data$cluster)

                 1  2  3
 Iris-setosa    50  0  0
 Iris-versicolor  0 48  2
 Iris-virginica   0 14 36
>     plot(iris_data[c("petal.length",     "petal.width")],     col     =
clustered_data$cluster)
>     plot(iris_data[c("sepal.length",     "sepal.width")],     col     =
clustered_data$cluster)
```



а) б)

Рис .5. Визуализация результатов кластеризации а) длины и ширины чашелистника; а) длины и ширины лепестка.

Аналогичным образом можно получить результаты с использованием алгоритма MeanShift (библиотека library(LPCM))

```
fit <- ms(iris_data.input, h=0.1)
> fit$cluster.label
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2
[57] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 2 3 3 2 3 2 3 2 2
[113] 3 2 2 3 2 4 3 2 3 2 3 2 3 3 2 2 2 2 3 4 2 2 2 3 3 2 2 3 3 3 2 3 3 3 2 2 3 2
>
```

Результаты кластеризации представлены на рисунке 6.

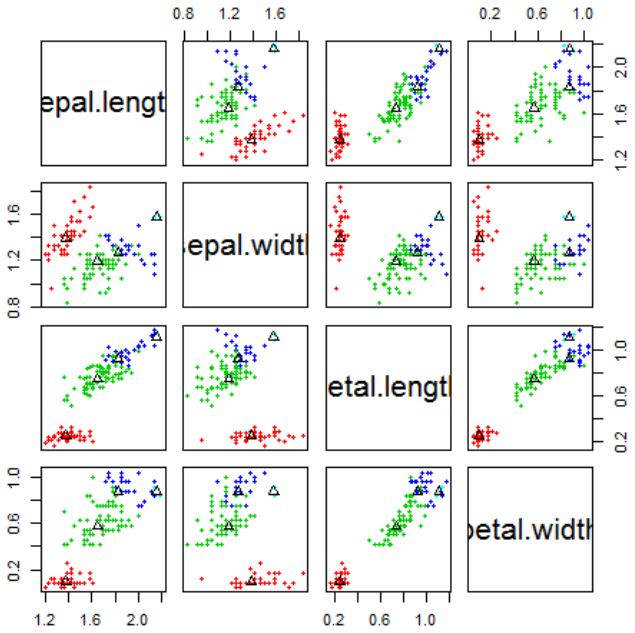


Рис. 6. Результаты кластеризации с использованием алгоритма MeanShift

ГЛАВА 5. РЕГРЕССИОННЫЙ АНАЛИЗ

Задачи, связанные с установлением зависимостей активно изучаются уже более 200 лет, с момента разработки К. Гауссом в 1794 г. метода наименьших квадратов. В математической статистике с этого времени было разработано огромное количество методов и инструментов для решения этих задач. Рассмотрим один из наиболее значимых и распространенных подходов – регрессионный анализ.

В общем случае, регрессионный анализ представляет собой задачу о выявлении внутренних свойств объекта по имеющимся данным о входах и выходах. Термин "регрессия" был введён Ф. Гальтоном (1886) и сначала он использовался в биологическом смысле. В статистике термин стал использоваться после опубликования работ К Пирсона (Pearson) в 1908 г.

Регрессионная модель $f(w,x)$ – это параметрическое семейство функций, задающее отображение

$$f: W \times X \longrightarrow Y,$$

где $w \in W$ – пространство параметров, $x \in X$ – пространство свободных переменных, Y – пространство зависимых переменных.

Так как регрессионный анализ предполагает поиск зависимости математического ожидания случайной величины от свободных переменных $E(y|x) = f(x)$, то в её состав входит аддитивная случайная величина ε:

$$y = f(w,x) + \varepsilon. \quad (5.1)$$

Параметры модели настраиваются таким образом, что модель наилучшим образом приближать данные. Критерием качества приближения (целевой функцией) обычно является среднеквадратичная ошибка. Предполагается, что зависимая переменная есть сумма значений некоторой модели и случайной величины. Относительно характера распределения случайной величины *y* делаются предположения, называемые гипотезой порождения данных. Для подтверждения или опровержения этой гипотезы

выполняются статистические тесты, такие как, например, F-тест, или критерий Фишера.

С помощью регрессионного анализа исследуются внутренние механизмы рассматриваемого явления, и оценивается роль отдельных факторов.

Основные задачи регрессионного анализа следующие:

- определения вида и формы зависимости;
- оценка параметров уравнения регрессии;
- проверка значимости уравнения регрессии;
- проверка значимости отдельных коэффициентов уравнения;
- построение интервальных оценок коэффициентов;
- исследование характеристик точности модели;
- построение точечных и интервальных прогнозов результирующей переменной.

Задача нахождения регрессионной модели ставится следующим образом [35]. Задана выборка – множество $\{x_1, x_2, \dots, x_N | x \in R\}$ значений свободных переменных и множество $\{y_1, y_2, \dots, y_N | y \in R\}$ соответствующих им значений зависимой переменной. Эти множества обозначаются как *D*, множество исходных данных $\{(x, y)_i\}$. Задана регрессионная модель (5.1) зависящая от параметров $w \in R$ и свободных переменных x. Требуется найти наиболее вероятные параметры w.

Методы регрессионного анализа делят на *многомерные* и *одномерные* в зависимости от числа независимых переменных, *линейные* и *нелинейные.* Для нахождения параметров нелинейных регрессионных моделей используются методы оптимизации, например метод сопряжённых градиентов, метод Ньютона-Гаусса и т.д.

Для реализации множественной регрессии существует множество подходов. Например, шаговый и ступенчатый метод [36], метод группового учета аргументов (МГУА) [37].

Модель линейной регрессии

Линейная регрессия предполагает, что функция f зависит от параметров w линейно. При этом линейная зависимость от свободной переменной x необязательна.

$$y = f(w, x) + \nu = \sum_{i=1}^{N} \quad w_i g_i(x) + \nu \tag{5.1}$$

В случае, когда функция $g_i(x) = x_i$, линейная регрессия имеет вид

$$y = \sum_{i=1}^{N} \quad w_i x_i + \nu = \langle w, x \rangle + \nu \tag{5.2}$$

здесь x_i — компоненты вектора x.

Значения параметров в случае линейной регрессии находят с помощью метода наименьших квадратов.

Разности $y_i - f(x_i)$ между фактическими значениями зависимой переменной и восстановленными называются *регрессионными остатками*. В литературе используются также синонимы: *невязки* и *ошибки*. Одной из важных оценок критерия качества полученной зависимости является сумма квадратов остатков:

$$SSE = \sum_{i=1}^{N} \quad (y_i - f(w, x_i))^2 \tag{5.3}$$

Здесь SSE — сумма среднеквадратичных ошибок.

Оценка качества модели регрессии

Проверка качества уравнения регрессии включает проверку ее общей точности, а также точности оценок параметров. Необходимо проверить а) является ли модель и ее параметры статистически значимыми (позволяет ли имеющаяся выборка судить о генеральной совокупности); б) какова практическая ценность модели (насколько информация, полученная по модели, снимает неопределенность в отношении регрессора Y).

Снижение качества модели может быть вызвано:

- наличием выбросов;

- наличием влиятельных наблюдений;
- нарушением основных предположений регрессионного анализа;
- наличие мультиколлинеарности.

5.1 КРИТЕРИИ ТОЧНОСТИ МОДЕЛИ

Наиболее простыми критериями можно считать *средние абсолютное и относительное отклонения* (mean absolute error, mean absolute percentage error):

$$MAE = \frac{1}{n}\sum_{k=1}^{n}\left|Y_k - Y_k^*\right|, \tag{5.4}$$

$$MAPE = \frac{1}{n}\sum_{k=1}^{n}\left|\frac{Y_k - Y_k^*}{Y_k}\right| \cdot 100\% \tag{5.5}$$

Первый критерий выражается в тех же единицах измерения, что и моделируемый ряд динамики, поэтому он может быть использован только для сравнения моделей одного и того же ряда динамики. С другой стороны, он позволяет оценить «физическое содержимое» погрешности моделирования.

Среднее относительное отклонение является безразмерной величиной и позволяет судить о точности модели и сравнивать их между собой. Однако ее значение также во многом зависит от уровней ряда динамики. Если значения Y_k велики по сравнению со своим разбросом, то MAPE-оценка будет малой, а если Y_k близки к нулю, то MAPE-оценка будет большой вне зависимости от точности модели. Если же имеются наблюдения, строго равные нулю, то использовать относительные величины вообще невозможно.

Одной из важных оценок критерия качества полученной зависимости является *сумма квадратов остатков*:

$$SSE = \sum_{i=1}^{N} \left(y_i - f(w, x_i)\right)^2$$

(5.6)

Коэффициент корреляции (выборочный) определяет силу линейной зависимости между показателями:

$$r_{YX} = r_{XY} = \frac{m_{YX} - m_Y m_X}{s_Y s_X}, \qquad (5.7)$$

$-1 \le r_{YX} \le 1$.

Чем ближе модуль $|r_{YX}|$ к 1, тем точнее модель. При $|r_{YX}| = 1$ между X и Y существует функциональная зависимость, при $|r_{YX}| = 0$ линейная связь полностью отсутствует.

Знак r_{XY} определяет направление зависимости (положительная или отрицательная).

Для нелинейных моделей коэффициент корреляции рассчитывается для их линеаризованной формы, например для логарифмической модели рассчитывается

$$r_{Y \ln X} = \frac{m_{Y \ln X} - m_Y m_{\ln X}}{s_Y s_{\ln X}}. \qquad (5.8)$$

Обычно считают, что если $|r_{YX}| < 0{,}5$, то линейная зависимость отсутствует, если $|r_{YX}| < 0{,}7$ – зависимость слабая, если $|r_{YX}| \ge 0{,}9$ – присутствует сильная линейная зависимость.

Коэффициент детерминации R^2:

$$R^2 = 1 - \frac{\sum_{k=1}^{n} \left(Y_k^* - Y_k\right)^2}{\sum_{k=1}^{n} \left(Y_k - m_Y\right)^2} = 1 - \frac{\sum_{k=1}^{n} e_k^2}{\sum_{k=1}^{n} \left(Y_k - m_Y\right)^2}. \qquad (5.9)$$

$\sum_{k=1}^{n}\left(Y_k^* - Y_k\right)^2 = \sum_{k=1}^{n}\varepsilon_k^2$ (сумма квадратов отклонений) – мера остаточного, не объясненного моделью разброса исходных данных.

$\sum_{k=1}^{n}\left(Y_k - m_Y\right)^2$ (общая сумма квадратов) – мера общего рассеивания Y_k относительно линии математического ожидания m_Y.

Смысл R^2: какая доля зависимого показателя не является случайной, т.е. описывается моделью.

Для линейных моделей $0 \le R^2 \le 1$. Чем ближе коэффициент детерминации к единице, тем точнее модель. При $R^2 = 1$ модель проходит точно через исходные данные.

Для нелинейных моделей коэффициент детерминации может быть отрицательным $R^2 \in (-\infty; 1]$, если модель совершенно не объясняет показатель (даже хуже, чем просто горизонтальная прямая). Причиной может быть, например, вычислительная ошибка, неверный выбор функционального вида модели.

Для линейных моделей $R^2 = r_{XY}^2$.

Основным недостатком R^2 является то, что при усложнении модели он возрастает и поэтому не может служить достоверным критерием выбора одной модели из нескольких.

Критерии Фишера (F-тест)

***F*-**критерий Фишера заключается в проверке гипотезы H_o *о статистической незначимости уравнения регрессии.* Для этого выполняется сравнение фактического $F_{факт}$ и критического (табличного) $F_{табл}$ значений *F*-критерия Фишера. $\boldsymbol{F}_{факт}$ определяется из соотношения значений факторной и остаточной дисперсий, рассчитанных на одну степень свободы. Выполняется сравнение $F_{факт}$ и критического (табличного)

$F_{табл}$ значений F-критерия Фишера. Если табличное значение меньше фактического, то признается статистическая значимость и надежность характеристик, если наоборот, то признается статистическая незначимость, ненадежность уравнения регрессии.

5.2 ПРИМЕРЫ ПРАКТИЧЕСКОГО ПРИМЕНЕНИЯ

Рассмотрим решение задачи множественной регрессии для задачи определения эффективности рекламы. Имеется набор данных в котором зафиксированы параметры проведенных рекламных кампаний. Для каждой кампании определены затраты на рекламу по телевидению (TV), радио (Radio) и в газетах (“Newspaper”) и определен показатель эффективности, выражаемый в количестве продаж (“Sales”). Результаты визуализации представлены на рисунке 7.

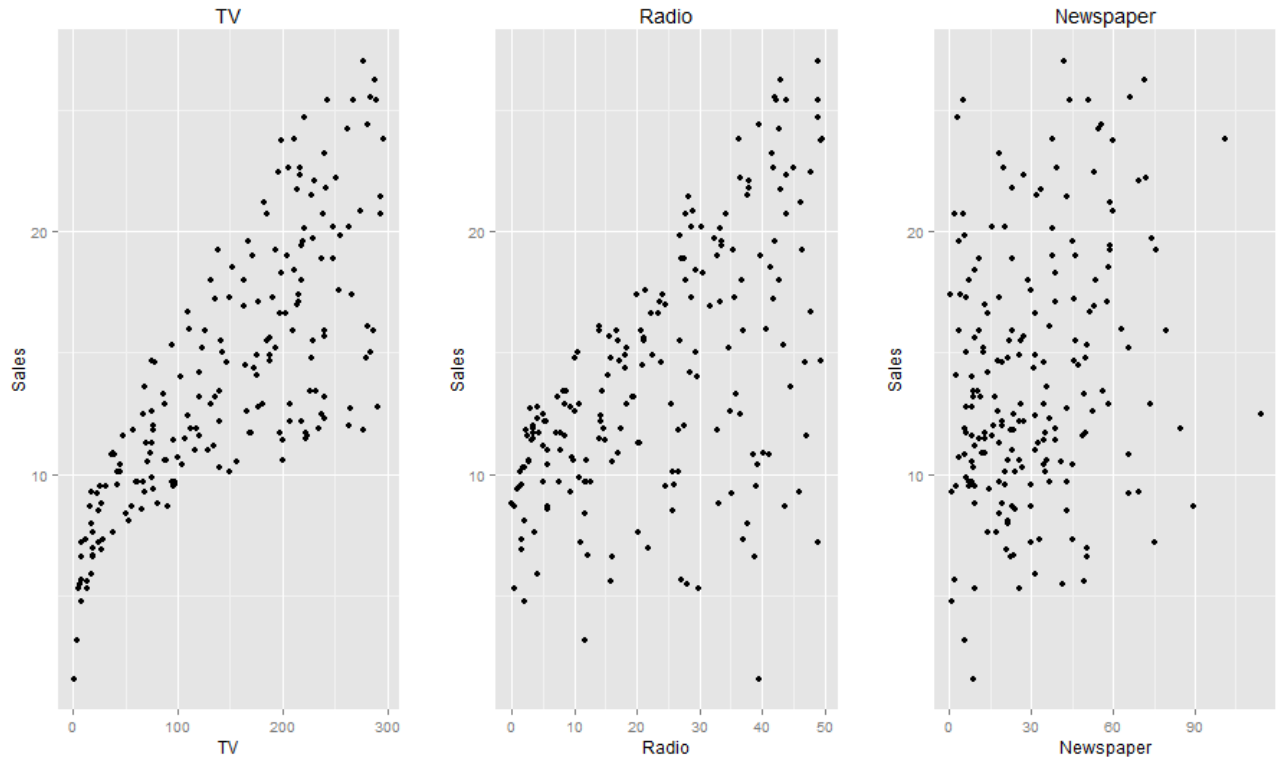


Рис.7. Результаты визуализации зависимости количества продаж от объемов рекламной кампании на телевидении (TV), радио (Radio) и в газетах (“Newspaper”)

Для реализации множественной регрессии используется команда lm, в которой задается формула вида y ~ x1 + x2 + …, где y – зависимая переменная, а x1, x2, … – независимые переменные (регрессоры)

```
> lr_fit <- lm(Sales ~ TV + Radio + Newspaper, data=adv)
```

Полученные результаты можно проанализировать с помощью команды `summary`.

```
> summary(lr_fit) # show results
Call:
lm(formula = Sales ~ TV + Radio + Newspaper, data = adv)
Residuals:
   Min      1Q  Median      3Q     Max
-8.8277 -0.8908  0.2418  1.1893  2.8292

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.938889   0.311908   9.422   <2e-16 ***
TV           0.045765   0.001395  32.809   <2e-16 ***
Radio        0.188530   0.008611  21.893   <2e-16 ***
Newspaper   -0.001037   0.005871  -0.177     0.86
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.686 on 196 degrees of freedom
Multiple R-squared:  0.8972,     Adjusted R-squared:  0.8956
F-statistic: 570.3 on 3 and 196 DF,  p-value: < 2.2e-16
```

Обратим внимание, что в результате получаются коэффициенты, значения которых отображаются с помощью команды

```
> coefficients(lr_fit) # model coefficients
(Intercept)           TV        Radio    Newspaper
2.938889369  0.045764645  0.188530017 -0.001037493
>
```

6. ЛАБОРАТОРЫНЙ ПРАКТИКУМ

Лабораторная работа №1. «Введение в R»

Цель: Получить практические навыки работы с языком статистического анализа R.

Порядок выполнения работы

1. Изучение основных типов данных в R и операций с ними.
2. Создание таблиц данных.
3. Основные операторы, условия и циклы.
4. Объявление и вызов функций.
5. Работа с пакетами R.
6. Выполнение индивидуального задания
7. Составление отчета

2. Контрольные вопросы

1) Формы представления, типы и виды анализируемых данных.
2) Общая схема интеллектуального анализа данных.
3) Способы получения и предобработки данных для извлечения знаний.
4) Какие особенности языка R вы можете выделить?
5) Какие типы данных есть в R?
6) Как создать таблицу в ?
7) Как задать функцию в R?
8) Как работать с встроенными и внешними пакетами в R?

3. Практические задания

Создайте фрейм данных из N записей со следующими полями: Nrow - номер записи, Name – имя сотрудника, BirthYear – год рождения, EmployYear – год приема на работу, Salary – зарплата. Заполните данный фрейм данными так, что Nrow изменяется от 1 до N. Name задается произвольно, BirthYear распределено равномерно (случайно) на отрезке

[1965, 1990], EmployYear распределен равномерно на отрезке [BirthYear +18, 2006], Salary задается произвольно в интервале от 10000 до 30000.

Подсчитайте число сотрудников с зарплатой, больше 15000. Добавьте в в таблицу поле, соответствующее суммарному подоходному налогу (ставка 13%), выплаченному сотрудником за время работы в организации.

Лабораторная работа №2. «Классические методы статистики и визуализация в R»

Цель: получить навыки решения базовых статистических задач в R и использования методов визуализации данных

. **Порядок выполнения работы**

1. Ознакомится с видами статистических распределений и способами их представлений.
2. Изучить методы описательной статистики.
3. Изучить способы проведения статистических тестов и проверки гипотез в R.
4. Ознакомиться с способом проведения дисперсионного анализа в R.
5. Изучить способы визуализации в R.
6. Выполнить индивидуальное задание
7. Составить отчет

2. Контрольные вопросы

1) Какие законы распределения случайных величин вы знаете?
2) Как в R вывести график плотности распределения случайной величины?
3) Для чего нужна описательная статистика?
4) Какие характеристики выборки вы знаете?
5) Как проверяются статистические гипотезы?
6) В чем суть дисперсионного анализа?

7) Какие способы представления графической информации для анализа данных вы знаете.
8) Общая схема анализа данных. Требования к алгоритмам анализа данных.
9) Способы получения и предобработки данных.
10) Примеры задач анализа данных для построения моделей искусственного интеллекта.

3. Практические задания

1. Подготовьте данные и используя критерий x^2, проверьте нуль-гипотезу, состоящую в том, что количество киберпреступлений не зависит от количества использующих интернет. Постройте мозайчатую диаграмму (mosaicplot).

2. Подготовьте данные зависимости убытков от кибератак от числа зарегистрированных кибератак. Постройте статистическую таблицу. Найдите среднее количество кибератак, дисперсию и стандартное отклонение; найдите линейную зависимость между убытками и количеством кибератак.

3. Найдите в интернете данные связанные с информационной безопасностью. Скачайте эти данные в виде таблицы CSV и постройте по ним столбчатую диаграмму, круговую диаграмму и диаграмму размаха (ящик с усами) Подберите цвета, попробуйте изменить форматирование рисунка.

Лабораторная работа №3. «Регрессия в R»

Цель: изучить методы регрессионного анализа и способы их реализации в R.

Порядок выполнения работы

1. Изучить предлагаемый теоретический материал.
2. Построить уравнение регрессии.
3. Оценить качество построенной модели.
4. Выполнить прогнозирование, используя построенную модель.
5. Подготовить отчет.

2. Контрольные вопросы

1. Для чего используют регрессионный анализ?
2. Какой вид имеет уравнение регрессии в общем случае?
3. Перечислите этапы регрессионного анализа.
4. С помощью какого метода находят коэффициенты регрессии?
5. Data Mining
6. Процесс ETL
7. Оценка качества моделей искусственного интеллекта
8. Недостатки систем искусственного интеллекта

3. Практические задания

Решите задачу множественной линейной регрессии для произвольного набора данных из репозитория UC Irvine Machine Learning Repository. Проанализируйте полученные результаты и сделайте выводы.

Лабораторная работа № 4. «Классификация и кластеризация в R»

Цель: изучить методы классификации и кластеризации и способы их реализации в R.

Порядок выполнения работы

1. Изучить предлагаемый теоретический материал.
2. Решить задачу кластеризации.
3. Решить задачу классификации.
4. Подготовить отчет.

2. Контрольные вопросы

1. Чем отличается классификация от кластеризации?
2. Постановка задачи классификации?
3. Что является результатом решения задачи классификации?
4. Постановка задачи кластеризации.
5. Что является результатом решения задачи кластеризации?
6. Что значит обучение с учителем?
7. Основные этапы кластерного анализа?
8. Какие методы классификации вы знаете?
9. Виды нейронных сетей
10. Структура интеллектуальной системы поддержки принятия решений на основе моделей искусственных нейронных сетей
11. Методы обучения нейронных сетей
12. Задачи компьютерного зрения
13. Где применяют технологии “Компьютерного зрения”
14. В чем состоит проблема понимания изображений?
15. Библиотеки для построения моделей распознавания образов.
16. В чем компромисс между точностью и объяснимостью в ИИ?

3. Практические задания

1) Кластеризация.

1. Выберите данные для задачи классификации.
2. Примените метод k средних для решения задачи кластеризации.
3. Визуализируйте данные (постройте облако точек, "раскрасив" точки в цвета, соответствующие номерам кластеров) и сравните полученные кластеры с облаками точек в соответствующей задаче классификации (за визуализацию Вы получите дополнительно 5 баллов).

4. Исследуйте работу метода, варьируя количество кластеров.

2) Классификация

1. Выберите данные для задачи классификации.

2. Решите задачу с помощью наивного Байесовского классификатора; если число признаков = 2, то визуализируйте данные.

3. Постройте кросс-валидационную таблицу, сделайте вывод о точности решения задачи классификации.

4. Задайте несколько новых данных, покажите соответствующие точки на графике (выделите их другим цветом).

5. Определите класс для новых данных.

7. ЗАДАНИЕ ДЛЯ КОНТРОЛЬНОЙ РАБОТЫ

В рамках контрольной работы студент выполняет индивидуальный исследовательский проект, связанный с анализом данных.

Задачами контрольной работы является:

- получение навыков формализации предметной области и постановки задач анализа данных;

- применение технологий и методов анализа данных для решения практических задач.

- получение навыков самостоятельной исследовательской деятельности.

Структура работы:

Титульный лист

Оглавление

1. Постановка задачи следует описать какие данные используются для анализа, структура и источники (1 стр)

2. Описание используемого метода (1 стр)

3. Практическая часть.

3.1 Описать инструмент анализа (пакет, язык, библиотека) (1 стр)

3.2 Описать процесс получения решения (основные экранные формы, тест программы, протокол) (3 стр)

3.3 Представить результаты полученного решения, сделать содержательные выводы (1 стр)

Заключение

Сделать практические выводы, рекомендации по результатам анализа данных

Литература

Правила оформления

1) Шрифт Times New Roman, 14, интервал – 1,5. Текст работы выравнивают «по ширине».
2) Текст оформляют с соблюдением следующих размеров полей: левое – 30 мм, правое – 10 мм, верхнее – 15 мм нижнее – 20 мм. Абзацы в тексте начинают отступом, равным 15 мм.
3) Листы должны иметь сквозную нумерацию. Номер страниц проставляют арабскими цифрами внизу по центру, без точки. На титульном листе Титульные листы, аннотации, разделы «Оглавление», «Введение», «Заключение», «Литература» («Источники» — при наличии ссылок на интернет-ресурсы) не нумеруются, но включаются общую нумерацию страниц.
4) Разделы, подразделы основной части пояснительной записки должны иметь заголовки. Заголовки следует писать с прописной буквы без точки в конце, не подчеркивая.
5) Список используемой литературы оформляется в соответствии с действующими правилами составления библиографии. Ссылки на источники приводятся по тексту в квадратных скобках.
6) Формулы и уравнения следует выделять из текста, в отдельную строку и располагаться по центру страницы. Пояснения символов и числовых коэффициентов, входящих в формулу, должны быть приведены непосредственно под формулой. Пояснения каждого символа следует давать с новой строки в той последовательности, в которой символы приведены в формуле. Первая строка пояснения должна начинаться со слова «где» без двоеточия. Формулы, следующие одна за другой и не разделенные текстом, разделяют запятой. Формулы нумеруются сквозной нумерацией арабскими цифрами, которые записывают на уровне формулы справа в круглых скобках. Одну формулу обозначают – (1). Ссылки в тексте на порядковые номера формул дают в скобках, например, в формуле (1).

7) Иллюстрации (рисунки, схемы, диаграммы, чертежи и т.п.) располагают как по тексту документа, так и в конце его. Иллюстрации, за исключением иллюстраций приложений, следует нумеровать арабскими цифрами сквозной нумерацией. Рисунок подписывается снизу по центру (Рисунок 1 – Название рисунка).
8) Ссылки на рисунки, таблицы, формулы, источники ОБЯЗАТЕЛЬНЫ!

Темы письменных работ (контрольной работы)

1. Анализ данных об образовательных ресурсах.
2. Анализ данных об объектах недвижимости.
3. Анализ городских данных.
4. Анализ транспортных потоков.
5. Анализ экологических данных.
6. Анализ уровня образования россиян: тенденции и дифференциация.
7. Анализ тенденции преступности.
8. Анализ медицинских данных
9. Анализ данных о состоянии здоровья
10. Анализ демографических данных.
11. Анализ данных для поддержки решений о размере страховых взносов
12. Анализ данных для задач оценки рисков (инвестиционных, экологических и пр.)
13. Анализ данных для определения потенциальных покупателей продукта
14. Анализ данных для определения целевых аудиторий.
15. Анализ инвестиционной привлекательности регионов России
16. Анализ пространственных данных
17. Анализ и прогнозирование востребованности профессий
18. Анализ влияния инновационной активности на уровень жизни
19. Анализ данных о научных исследованиях

Web-ресурсы (примеры и источники данных)

http://statsoft.ru/

https://basegroup.ru/

http://www.algorithmist.ru/2011/05/clustering-with-example-in-r.html

http://opendata.volganet.ru/transport.sort

https://habrahabr.ru/post/340698/

https://habrahabr.ru/company/cloud4y/blog/334234/

https://knoema.ru

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Найдич, А. Большие данные: насколько они большие? [Электронный ресурс] / А. Найдич ; Компьютер Пресс. – Москва, [2012]. – Режим доступа : http://compress.ru/article.aspx?id=23469

2. Тиндал Сьюзен. Большие данные: все, что вам необходимо знать [Электронный ресурс] / Тиндал Сьюзен : PC Week/RE. - Москва, [2012] – Режим доступа : http://www.pcweek.ru/idea/article/detail.php?ID=141962

3. Майер-Шенбергер, В. Большие данные. Революция, которая изменит то, как мы живем, работаем и мыслим / Виктор Майер-Шенбергер, Кеннет Кукьер ; пер. с англ. Инны Гайдюк. — Москва : Манн, Иванов и Фербер, 2014. – 240 с.

4. Паклин, Н. Б. Бизнес-аналитика от данных к знаниям / Н. Б. Паклин, В. И. Орешков. – 2-изд., испр. – Санкт-Петербург : Питер, 2013. – 706 с.

5. Ralph Kimball, Margy Ross The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, John Wiley and Sons, Ltd, 2013. – 421 p.

6. Yahoo!Apache Hadoop [Электронный ресурс] : Режим доступа: https://developer.yahoo.com/hadoop/tutorial/module1.html

7. Ханк, Д. Э. Бизнес-прогнозирование: пер. с англ. / Д. Э. Ханк, Д. У. Уичерн, А. Д. Райтс. – 7-е изд. – Москва : Издат. дом «Вильямс», 2003.

– 651 с.

8. Анализ данных и процессов: учеб. пособие / А. А. Барсегян [и др.]. – 3-е изд., перераб. и доп. – Санкт-Петербург : БХВ-Петербург, 2009. – 512 с.

9. Любицын, В. Н. Повышение качества данных в контексте современных аналитических технологий / В. Н. Любицын // Вестник ЮУрГУ. - № 23. – 2012. - С. 83-86

10. Большие данные [Электронный ресурс] / TADVISER. - Режим доступа : http://www.tadviser.ru/index.php

11. Воронцов, К. В. Курс лекций [Электронный ресурс] / К. В. Воронцов. - Режим доступа : http://www.machinelearning.ru

12. Боровиков, В. П. Популярное введение в современный анализ данных в системе STATISTICA / В. П. Боровиков – Москва : Горячая линия – Телеком, 2013. – 288 с.

13. Халафян, А. А. STATISTICA 6. Статистический анализ данных / А. А. Халафян – Москва : Бином-Пресс, 2010. - 528 с.

14. Наследов, А. IBM SPSS Statistics 20 и Amos: Профессиональный статистический анализ данных : Практическое руководство / А. Наследов. - Санкт-Петербург : Питер, 2013, - 416 с.

15. Data Mining (Megaputer Intelligence, PolyAnalyst) [Электронный ресурс] : Режим доступа: http://www.exponenta.ru/soft/Others/polyanalyst/polyanalyst.asp

16. Кулаичев, А. П. Методы и средства комплексного анализа данных : учебное пособие для вузов по дисциплинам "Прикладная статистика", "Информатика" / А. П. Кулаичев . – 4-е изд., перераб. и доп. – Москва : ФОРУМ: ИНФРА-М, 2006. – 512 с.

17. Мастицкий, С. Э. Статистический анализ и визуализация данных с помощью R [Электронный ресурс] / С. Э. Мастицкий, В. К. Шитиков : R: Анализ и визуализация данных. – 2014. - Режим доступа:

http://r-analytics.blogspot.com

18. Бесплатные on-line курсы на CodeSchool по R [Электронный ресурс] : Режим доступа: https://www.codeschool.com/courses/try-r

19. Курсы на Coursera по анализу данных [Электронный ресурс] : Режим доступа: https://www.coursera.org/specializations/jhudatascience

20. Венэбльз, У. Н. Введение в R. Версия 3.1.0 / У. Н. Венэбльз, Д. М. Смит; перевод и редакция Фоменко А. А. – Москва, 2014. – 109 с.

21. Chang, W. R Graphics Cookbook / W.Chang, - O'Reilly Media, 2013. - 416 p.

22. Воронцов, К. В. Алгоритмы кластеризации и многомерного шкалирования: Курс лекций. МГУ / К. В. Воронцов – Москва, 2007. – 18 с.

23. Миркин, Б. Г. Методы кластер-анализа для поддержки принятия решений: обзор / Б. Г. Миркин; Национальный исследовательский университет «Высшая школа экономики». – Москва : Изд. дом Национального исследовательского университета «Высшая школа экономики», 2011. – 88 с. – 150 экз.

24. MacQueen J.B. Some methods for classification and analysis of multivariate observations // Proc. of the 5th Berkeley Symposium on Mathematical Statistics and Probability. Vol. 1. Berkeley: Univ. of Calif. Press, pp. 281–297, 1967.

25. Bezdek J., Hathaway R., Sobin M., Tucker W.Convergence theory for fuzzy c-means: counterexamples and repairs // IEEE Trans. on Systems, Man, and Cybernetics. 1987. N 17. – p. 873–877.

26. Huang, Z.: Clustering large data sets with mixed numeric and categorical values, Proceedings of the First Pacific Asia Knowledge Discovery and Data Mining Conference, Singapore, pp. 21-34, 1997.

27. Huang, Z.: Extensions to the k-modes algorithm for clustering large data sets with categorical values, Data Mining and Knowledge Discovery 2(3), pp. 283-304, 1998.

28. Cao, F., Liang, J, Bai, L.: A new initialization method for categorical data clustering, Expert Systems with Applications 36(7), pp. 10223-10228., 2009.

29. Sharmila, Kumar, M.: An optimized farthest first clustering algorithm, Engineering (NUiCONE), Nirma University International Conference, India, pp. 1-5, 2013.

30. Bahmani, B., Moseley, B., Vattani, A., Kumar, R., Vassilvitskii, S.: Scalable k-means++, Journal Proceedings of the VLDB Endowment, Vol. 5 issue 7, pp. 622-633, 2012.

31. Zhang, T.; Ramakrishnan, R.; Livny, M.: BIRCH: an efficient data clustering method for very large databases, Proceedings of the 1996 ACM SIGMOD international conference on Management of data - SIGMOD '96, , New York, USA, pp. 103-114, 1996.

32. Ke Wang, Chu Xu, Bing Liu. Clustering Transactions Using Large Items, In Proc. CIKM'99, Kansas, Missouri, 1999.

33. Yang, Y., Guan, X., You, J.: CLOPE: a fast and effective clustering algorithm for transactional data, Proceeding KDD '02 Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, New York, USA, pp. 682-687, 2002.

34. Guojun Gan, Chaoqun Ma, Jianhong Wu. Data Clustering: Theory, Algorithms, and Applications, ASA-SIAM Series on Statistics and Applied Probability, SIAM, p. 466, 2007.

35. Дрейпер, Н. Прикладной регрессионный анализ : в 2-х т. / Н. Дрейпер, Г.Смит. – Москва : Финансы и статистика, т.1 - 1986, т.2 – 1987.

36. Петрович, М. Л. Регрессионный анализ и его математическое обеспечение на ЕС ЭВМ: Практическое руководство / М. Л. Петрович. – Москва : Финансы и статистика, 1982. – 199 с.

37. Ивахненко, А. Г. Индуктивный метод самоорганизации моделей сложных систем / А. Г. Ивахненко. – К.: Наук. думка, 1982. – 360 с.

Учебное издание

Наталья Петровна **Садовникова,**

Максим Владимирович **Щербаков**

ТЕХНОЛОГИИ АНАЛИЗА ДАННЫХ

Редактор

Компьютерная верстка

Темплан заказных изданий 2021 г. Поз. №

Подписано в печать …. Формат 60х84 1/16. Бумага офсетная.

Гарнитура Times. Печать офсетная Усл. Печ. Л. __. Уч.изд. л. ___.

Тираж 100 экз. Заказ

Волгоградский государственный технический университет

400005, г. Волгоград, пр. им. В. И. Ленина, 28, корп.1

Отпечатано в типографии ИУНЛ ВолгГТУ.

400005, г. Волгоград, пр. им. В. И. Ленина, 28, корп.7