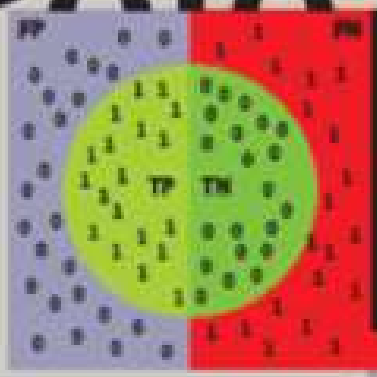
ИТМО

А.С. Ватьян, Н.Ф. Гусарова, Н.В. Добренко

DATA SCIENCE: ПРОБЛЕМЫ И РЕШЕНИЯ



Санкт-Петербург 2025

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

УНИВЕРСИТЕТ ИТМО

А.С. Ватьян, Н.Ф. Гусарова, Н.В. Добренко

DATA SCIENCE: проблемы и решения

РЕКОМЕНДОВАНО К ИСПОЛЬЗОВАНИЮ В УНИВЕРСИТЕТЕ ИТМО
по направлениям подготовки
45.03.04 - Интеллектуальные системы в гуманитарной сфере,
11.03.02 - Инфокоммуникационные технологии и системы связи
09.03.02 Информационные системы и технологии
в качестве учебного пособия для реализации основных профессиональных образовательных программ высшего образования бакалавриата

Санкт-Петербург

2025

УДК 004.89
ББК 32.813.5
В21

В21 Ватьян А.С., Гусарова Н.Ф., Добренко Н.В. DATA SCIENCE: проблемы и решения. – СПб: Университет ИТМО, 2025. – 219 с.

ISBN 978-5-7577-0731-0

Рецензент: Шалыто Анатолий Абрамович, профессор, ведущий научный сотрудник, ОАО «НПО «Аврора», главный научный сотрудник, национальный центр когнитивных разработок

Наука о данных (Data Science, DS) на сегодняшний день формирует группу самых востребованных приложений ИТ в бизнесе.
Пособие не только описывает методологии DS проекта, представляющие собою симбиоз предметных знаний разных областей и возможностей машинного обучения и искусственного интеллекта и поддерживающими их технологическими решениями, но и формирует базовое понимание научных основ этих решений. Большое внимание уделяется источникам ошибок при реализации этих проектов и мерам по их предупреждению. В пособии сделана попытка отразить пересечение всех этих тенденций. Насколько известно авторам, материал такого охвата предлагается русскоязычным читателям впервые.

ISBN 978-5-7577-0731-0

Содержание

ВВЕДЕНИЕ

Среди обширного спектра квалификаций, востребованных на рынке труда в области информационных технологий, «специалист по данным» (data scientist) занимает сегодня одну из лидирующих позиций. Возникнув в конце 1990-х гг. как тренд в математической статистике, наука о данных (Data Science, DS) выросла в междисциплинарный подход, который использует математику, статистику, искусственный интеллект и машинное обучение в сочетании с опытом экспертов в конкретной предметной области для извлечения или экстраполяции знаний и из потенциально зашумленных, структурированных или неструктурированных данных [Dhar, 2013].

Пособие построено в соответствии с программой курса «Введение в науку о данных», читаемого на мегафакультете трансляционных информационных технологий Университета ИТМО с 2023 года. Большое внимание в пособии уделяется источникам ошибок при реализации DS проектов и мерам по их предупреждению.

Изучение материалов пособия, в рамках данной дисциплины, иллюстрированных примерами конкретных технологических решений в различных сферах приложения DS позволит сформировать у студентов следующие знания: основные методы, способы и средства получения, хранения, обработки и защиты информации; общие принципы обработки данных с целью устранения шумов, пустот и прочих элементов; основные методы анализа информации с помощью специализированного прикладного программного обеспечения; методы математической статистики для обработки первичных данных;

умения: осуществлять выбор специализированного прикладного программного обеспечения для решения профессиональных задач; применять методы математической статистики для обработки первичных данных для решения решать задач профессиональной деятельности; использовать средства хранения и передачи информации для работы с цифровым следом;

навыки: применять методы математической статистики для обработки первичных данных для решения решать задач профессиональной деятельности; анализ потребностей и целей пользователей (людей, групп людей и ИКС); использования существующих информационных технологий при решении задач профессиональной деятельности.

Подробно описанные в пособии научные основы, методологии DS проекта и возможные ошибки при реализации этих проектов, а также практическая часть курса, содержащая в себе основной спектр задач специалиста по данным, совместно сформируют у студента не только компетенции, включающие в себя способность подготавливать, обрабатывать и анализировать большие, сложные и слабо структурированные данные, которые генерирует современный бизнес и социум, но и личные качества, такие как креативность и мотивация, а также способность находить оптимальные решения с широких позиций в области DS проектов.

Преподаватели и студенты могут использовать пособие в качестве дополнительного и расширяющего материала к лекционному курсу, а также при проведении практических занятий.

1. ОБЩАЯ ХАРАКТЕРИСТИКА НАУКИ О ДАННЫХ

1.1. Предмет науки о данных

1.1.1. Определения науки о данных

В литературе можно найти разные определения DS. Ограничимся несколькими:

- Наука о данных – это концепция объединения статистики, анализа данных, информатики и связанных с ними методов для понимания и анализа фактических явлений с данными [Hayashi, 1998].
- Наука о данных – это междисциплинарная академическая область [Donoho, 2017; Emmert-Streib, 2018], которая использует статистику, научные вычисления, научные методы, процессы, научную визуализацию, алгоритмы и системы для извлечения или экстраполяции знаний и идей из потенциально шумных, структурированных или неструктурированных данных [Dhar, 2013].
- Наука о данных – это расширение статистики, способное справляться с огромными объемами данных, производимыми в наши дни; DS добавляет методы из Computer science в репертуар статистики [Дэви, 2017].
- Наука о данных использует методы и теории, взятые из многих областей в контексте математики, статистики, компьютерных наук, информационной науки и доменных знаний [Cao, 2017]. Она также интегрирует доменные знания из базовой прикладной области (например, естественные науки, информационные технологии и медицина) [Danyluk, 2021].
- Наука о данных многогранна и может быть описана как наука, исследовательская парадигма, метод исследования, дисциплина, рабочий процесс и профессия [Mike, 2023].

Кроме того, лауреат премии Тьюринга Джим Грей представил науку о данных как «четвертую парадигму» науки (эмпирическую, теоретическую, вычислительную и теперь основанную на данных) и утверждал, что все в науке меняется из-за влияния информационных технологий и потока данных [Hey, 2009: Bell, 2009].

1.1.2. Виды анализа данных

Все приведенные определения согласны в том, что DS – деятельность, связанная с анализом данных и поиском лучших решений на их основе. Раньше подобными задачами занимались специалисты по математике и статистике. В книге [Тьюки, 1981] известный американский специалист по математической статистике, Д.У. Тьюки, подразделяет статистический анализ на два этапа: разведочный и подтверждающий.

Под разведочным анализом данных понимается первичная обработка результатов наблюдений, осуществляемая посредством простейших средств – карандаша, бумаги и логарифмической линейки (сейчас диапазон применяемых средств, конечно, расширился). Этот этап включает преобразование данных наблюдений и способы их наглядного представления, позволяющие выявить

внутренние закономерности, проявляющиеся в данных. Цели разведочного анализа:

- максимальное «проникновение» в данные,
- выявление основных структур,
- выбор наиболее важных переменных,
- обнаружение отклонений и аномалий,
- проверка основных гипотез,
- разработка начальных моделей.

Основные средства разведочного анализа – изучение вероятностных распределений переменных, построение и анализ корреляционных матриц, факторный анализ, дискриминантный анализ, многомерное шкалирование.

На втором этапе – подтверждающем анализе – применяются традиционные статистические методы оценки параметров и проверки гипотез.

Таким образом, корни DS лежат в статистике, она опирается на математику и информатику. Однако современная DS сделала большие шаги от разведочного анализа Тьюки. Включение в DS методов информатики, машинного обучения, искусственного интеллекта позволило эффективно работать с большими, сложными и слабо структурированными данными, которые генерирует современный бизнес и социум.

Специалисты компании Amazon, одного из пионеров использования DS для оптимизации бизнеса, выделяют четыре вида анализа данных, который реализуется в рамках DS [Amazon, 2024]:

1. *Описательный анализ* (Descriptive analysis) изучает данные, чтобы получить представление о том, что произошло или происходит в среде данных. Он широко использует визуализацию данных, такую как круговые диаграммы, столбчатые диаграммы, линейные графики, таблицы или сгенерированные повествования. Например, служба бронирования авиабилетов может регистрировать такие данные, как количество билетов, забронированных каждый день. Описательный анализ выявит всплески бронирования, спады бронирования и высокоэффективные месяцы для этой службы.
2. *Диагностический анализ* (Diagnostic analysis) предполагает глубокое погружение или подробное изучение данных для понимания того, почему что-то произошло. Он характеризуется такими методами, как детализация, обнаружение данных, интеллектуальный анализ данных и корреляции. При этом над заданным набором данных могут быть выполнены множественные преобразования, чтобы обнаружить уникальные закономерности в каждом из этих методов. Например, служба бронирования авиабилетов может выделить месяц, в котором произошел всплеск бронирования. Это может привести к выявлению закономерности: многие клиенты приезжают в определенный город, чтобы посетить ежемесячное спортивное мероприятие.
3. *Предиктивный анализ* (Predictive analysis) использует исторические данные для составления точных прогнозов относительно закономерностей данных, которые могут возникнуть в будущем. Для этого применяются такие методы, как машинное обучение, прогнозирование, сопоставление с образцом и пре-

дикативное моделирование. В каждом из этих методов компьютеры обучаются обратному проектированию причинно-следственных связей в данных. Например, служба полетов может использовать DS для прогнозирования закономерностей бронирования рейсов на следующий год в начале каждого года. Компьютерная программа или алгоритм могут просматривать прошлые данные и прогнозировать всплески бронирования для определенных направлений в мае. Предвидя будущие потребности своих клиентов в поездках, компания может начать целевую рекламу для этих городов с февраля.

4. *Предписывающая аналитика* (Prescriptive analysis) выводит предиктивные данные на новый уровень. Она не только предсказывает, что, скорее всего, произойдет, но и предлагает оптимальный ответ на это. Она может анализировать потенциальные последствия различных вариантов и рекомендовать наилучший курс действий. Он использует анализ графов, моделирование, сложную обработку событий, нейронные сети и рекомендательные механизмы из машинного обучения. В примере с бронированием авиабилетов предписывающий анализ может рассмотреть предыдущие маркетинговые кампании, чтобы максимально использовать преимущества предстоящего всплеска бронирований. Специалист по данным может прогнозировать результаты бронирования для разных уровней маркетинговых расходов на разных маркетинговых каналах. Эти прогнозы данных дадут компании по бронированию авиабилетов большую уверенность в своих маркетинговых решениях.

1.1.3. Типы задач DS

Как отмечает [Келлехер], одним из важнейших навыков специалиста по данным является способность сформулировать решаемую проблему как стандартную задачу науки о данных. Типизация задач DS связана со способом формирования модели проблемы, которая, в свою очередь, определяется соотношением знаний и данных о проблеме, которыми располагает дата-сайентист. Здесь существует два подхода: дедуктивное обучение и индуктивное обучение (обучение по прецедентам) [Айвазян, 1983; Айвазян, 1985].

Дедуктивное обучение. Имеется набор данных (датасет), представляющий собой выборку из генеральной совокупности и описывающий ситуацию в виде прецедентов (пар «объект–ответ»). Известен набор моделей, которые могут в явной форме описать связь между объектами и ответами. На основании теоретических знаний и экспертного опыта специалист выбирает наиболее подходящую, на его взгляд, модель и формулирует ее в виде гипотезы – утверждения, которое может быть проверено формально. Проверка гипотезы осуществляется статистическими средствами – определяется вероятность, с которой данная гипотеза соответствует всем имеющимся прецедентам. Если эта вероятность достаточно высока, то гипотеза признается верной и используется в качестве прогноза для дальнейшего развития ситуации по всей генеральной совокупности; если нет, то выбирается следующая модель, и т.д.

Тип задач DS, которые решаются в рамках этого подхода, – *проверка статистических гипотез*. Примеры таких задач: является ли новая вакцина от

COVID-19 эффективной? какова связь между внедрением нового дизайна сайта и изменением уровня продаж в интернет-магазине?

Машинное обучение (обучение по прецедентам). Как и в первом случае, имеется набор данных (датасет), представляющий собой выборку из генеральной совокупности и описывающий ситуацию в виде прецедентов (пар «объект–ответ»). Предполагается, что существует зависимость между ответами и объектами, но она неизвестна и, как правило, не выражается аналитически. В этом случае выбирается модельная среда (нейросеть), которая обучается по прецедентам – т.е. подстраивает свои весовые коэффициенты так, чтобы для максимально большого количества объектов формировать ожидаемые ответы. Так как гипотеза в явном виде здесь отсутствует, то для проверки правильности обучения выделяется дополнительный датасет, на котором нейросеть не обучалась. Обученная таким образом нейросеть с весами признается моделью и используется для дальнейших прогнозов развития ситуации по всей генеральной совокупности.

В рамках этого подхода выделяется несколько типов задач DS, которые принципиально соответствуют типизации задач машинного обучения, в том числе:

- кластеризация (или сегментация) – нужна ли адресная реклама для клиентов нашего интернет-магазина? есть ли типичные группы проблем в интернет-трафике?
- обнаружение аномалий – есть ли такие дни/часы, когда уровень продаж интернет-магазина сильно падает/сильно растет?
- поиск ассоциативных правил – какие товары чаще всего покупают вместе?
- прогнозирование (включая подзадачи классификации – как прогнозировать отток клиентов? – и регрессии – какую цену можно будет поставить на товар, если производитель внесет в него определенные изменения?).

Единый и исчерпывающий список задач DS, скорее всего, построить невозможно и не нужно: во-первых, они содержательно пересекаются, а во-вторых, постоянно появляются новые задачи, а старые теряют актуальность. Тем не менее, базовая «тройка» задач DS в классе обучения по прецедентам остается незыблемой: это кластеризация, классификация и регрессия [Amazon].

Классификация – это сортировка данных по определенным группам или категориям, например:

- разделение продуктов на популярные или непопулярные;
- разделение заявок на страхование на высокорисковые или низкорисковые;
- сортировка комментариев в социальных сетях на положительные, отрицательные или нейтральные.

Регрессия – это метод поиска взаимосвязи между двумя точками данных или процессами. Связь может представляться математической формулой или графиком. Когда значение одной точки данных известно, регрессия используется для прогнозирования другой точки данных, например:

- взаимосвязь между удовлетворенностью клиентов и количеством сотрудников;

– взаимосвязь между количеством пожарных станций и количеством травм в результате пожара в определенном месте.

Кластеризация – это метод группировки тесно связанных данных для поиска закономерностей и аномалий. В отличие от классификации, при кластеризации данные нельзя точно классифицировать по фиксированным категориям, поэтому они группируются в наиболее вероятные взаимосвязи. С помощью кластеризации можно обнаружить новые закономерности и взаимосвязи, например:

– группировка клиентов со схожим поведением при покупке для улучшения обслуживания клиентов;
– группировка сетевого трафика для определения ежедневных закономерностей использования и более быстрого выявления сетевой атаки.
– объединение статей в несколько различных новостных категорий для поиска фейкового новостного контента.

1.1.4. Акторы процесса DS

Многогранный процесс DS реализуется во взаимодействии различных специалистов, среди которых сегодня принято выделять несколько основных ролей – бизнес-аналитик, эксперт предметной области, специалист по данным, аналитик данных, инженер по данным [Amazon, 2024; Зыков, 2021; Кузнецов, 2022].

Аналитик данных (data analyst) в основном фокусируется на ретроанализе данных и выведении из него закономерностей, используя для этого статистику, математику, статистический анализ и средства визуализации. Результатом его работы могут быть регулярные отчеты.

Специалист по данным (data scientist) призван строить по тем же данным более широкую картину – например, специалист по данным может проектировать способ хранения, обработки и анализа данных. Проще говоря, аналитик данных обрабатывает существующие данные, извлекая из них полезную информацию, тогда как специалист по данным создает новые методы и инструменты для работы аналитиков.

Специалист по данным часто рассматривается как краеугольный камень проекта по науке о данных, но лишь немногие обладают полным спектром навыков, традиционно связанных с этой ролью. К ним относятся:

– расширенные знания статистики и визуализации данных,
– знание нескольких языков программирования,
– экспертиза в области машинного обучения и глубокого обучения,
– сильная деловая хватка,
– отличные навыки общения и презентации.

В небольших командах специалисты по данным часто выполняют несколько функций, включая задачи, которые обычно выполняют инженеры по данным или архитекторы. Роль специалиста по данным также может меняться в зависимости от среды проекта — требуя большего или меньшего участия в других технических, машинном обучении или предметно-ориентированных областях.

Фактически сегодня понятие «специалист по данным» стало настолько широким, что вызвало настоящие дебаты о том, как определить его роль и требуемые опыт и навыки. Тем не менее, можно перечислить их, опираясь на мнение большинства экспертов DS, как это сделано на рис. 1.1.

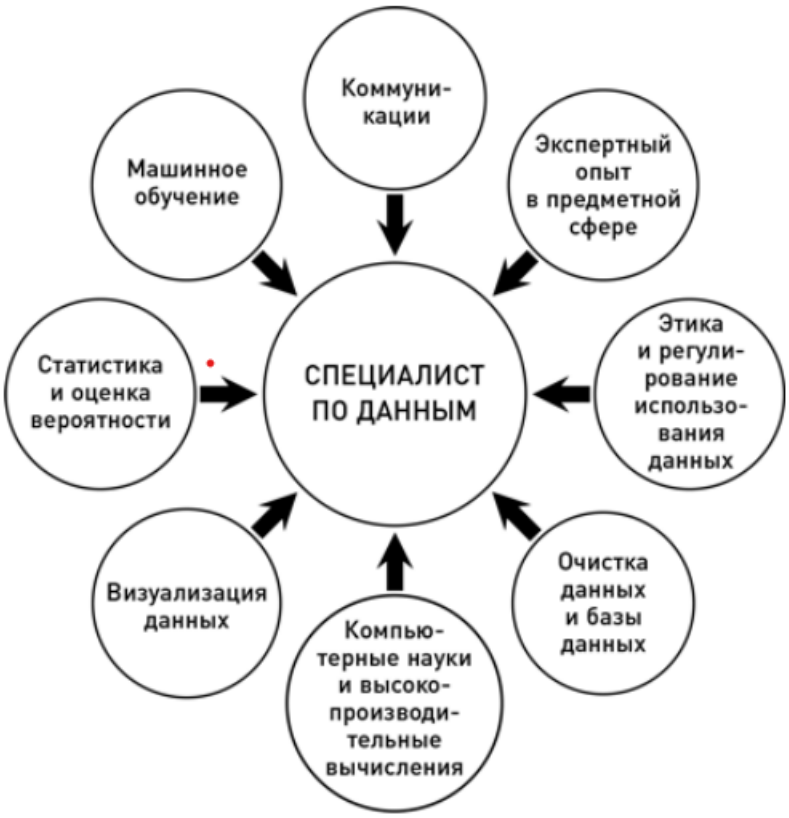


Рис. 1.1. Номенклатура навыков специалиста по данным [Келлехер]

Бизнес-аналитик (business-analyst) устраняет разрыв между бизнесом и ИТ. Он определяет бизнес-кейсы, собирает информацию от заинтересованных сторон или проверяет решения. Бизнес-аналитиков иногда путают с аналитиками данных, поскольку и те, и другие активно работают с данными. Однако бизнес-аналитики обычно больше сосредоточены на извлечении действенных идей из хранилищ данных с помощью таких инструментов, как Tableau, и они могут не так часто погружаться в предиктивное моделирование, как аналитики данных. Границы между этими двумя ролями могут быть размытыми, особенно в небольших компаниях, но бизнес-аналитики, как правило, больше ориентированы на бизнес и меньше вовлечены в кодирование.

Инженер по данным (data engineer) создает и поддерживает системы, которые позволяют специалистам по данным получать доступ к данным и интерпретировать их. Роль обычно заключается в создании моделей данных, построении конвейеров данных и надзоре за извлечением, преобразованием, загрузкой (Extract, Transform, Load, ETL). В зависимости от настроек и размера организации инженер по данным может также управлять связанной инфраструктурой, такой как хранилища больших данных, потоковые передачи и платформы обработки, такие как Amazon S3. Специалисты по данным используют данные, которые специалисты по данным обработали, для построения и обучения прогно-

стических моделей. Затем специалисты по данным могут передавать результаты аналитикам для дальнейшего принятия решений.

Эксперт предметной области (domain expert) – очень важная фигура в DS-проекте: исходя из своего опыта работы и интуиции (инсайтов), он может предложить специалисту по данным самые первые гипотезы для DS-проекта, а также проверить получившиеся результаты DS-проекта с точки зрения «здравого смысла». Конечно, специалист по данным тоже должен иметь экспертный опыт в предметной сфере. Большинство проектов начинаются с реальной проблемы и необходимости разработать ее решения. Специалист по данным должен понимать и проблему, и то, как ее решение могло бы вписаться в организационные процессы. Этот экспертный опыт направляет специалиста при поиске оптимального решения, а также позволяет конструктивно взаимодействовать с экспертами предметной области. Кроме того, специалист по данным может использовать его в работе над аналогичными проектами в той же или смежной областях и быстро определять их фокус и охват.

В целом успех DS проектов зависит не только от создания мощных моделей, но и от коллективного опыта разнообразной команды. Любой DS проект требует ряда навыков, включая инженерию, понимание бизнеса и прикладные знания, в зависимости от поставленной задачи. В то время как крупные организации могут позволить себе роскошь нанимать экспертов для каждой роли, небольшим командам часто приходится сбалансировать различные обязанности, требуя от участников выходить за рамки своих основных компетенций.

1.2. Данные и стандарты для их описания

1.2.1. Типы данных

В [Ou, 2024] предлагается хотя и не совсем стандартная, но достаточно обобщенная типизация данных:

1. Таблица – наиболее распространенный и удобный для обработки средствами информационных технологий тип данных. Каждая строка соответствует образцу (сэмплу), а каждый столбец – атрибуту или признаку, описывающему этот сэмпл.

2. Набор точек (датасет) – в этом случае данные можно рассматривать как набор точек в определенном пространстве, не обязательно эвклидовом.

3. Временной ряд. Текст, разговор и последовательности ДНК могут быть включены в эту категорию, которую также можно рассматривать как функцию переменной времени.

4. Изображение. Его можно рассматривать как функцию двух переменных.

5. Видео – это функция как переменной времени, так и пространственных переменных.

6. Веб-страница и газета. Хотя каждая статья на веб-странице или в газете является частью временной последовательности, они также имеют пространственную структуру.

7. Сетевые данные – граф, состоящий из вершин, в которых содержатся данные, и ребер, соединяющих их.

Основные типы данных могут объединяться в другие данные, более высокого порядка, такие как набор изображений, набор временных последовательностей, последовательности таблиц и т.д.

При таком высоком уровне обобщенности описания данных можно выделить два положения, принципиально важных для организации процесса DS в целом:

- поскольку в процессе сбора данных неизбежны шумы, все эти модели обычно являются стохастическими моделями. Обычно нам также необходимо аппроксимировать стохастические модели. Один подход заключается в аппроксимации стохастической модели детерминированной моделью. К этому типу относятся модели регрессии и обработки изображений, основанные на вариационном принципе. Другой подход заключается в моделировании распределения данных, например, путем предположения, что данные следуют определенному распределению или временная последовательность удовлетворяет предположениям цепи Маркова;
- базовое предположение анализа данных заключается в том, что наблюдаемые данные генерируются моделью. Основная проблема анализа данных – найти эту модель. Конечно, в большинстве случаев нас не интересует вся модель. Вместо этого мы просто хотим узнать некоторое важное содержимое. Например, мы используем корреляционный анализ, чтобы определить, коррелируют ли два набора данных или нет, и используем методы классификации и кластеризации, чтобы разделить данные на группы.

1.2.2. Типы моделей данных

В [Ou, 2024] выделены базовые математические структуры, используемые для построения моделей данных:

1. *Метрическая структура*. Для встраивания набора данных в метрическое пространство можно ввести некоторые метрики (например, расстояние). Функция косинусного расстояния является типичным примером в обработке текста.
2. *Сетевая структура*. Некоторые данные, включая социальные сети, имеют органичную сетевую структуру, в то время как некоторые другие данные без сетевой структуры могут быть организованы в сеть на основании определенных оценок. Например, для набора точек в метрическом пространстве мы можем решить, существует ли ребро между двумя вершинами или нет, на основе расстояния между ними, из которого мы можем получить сетевую структуру. Алгоритм PageRank является типичным примером сетевой структуры.
3. *Алгебраическая структура*. Данные можно рассматривать как векторы, матрицы или тензоры более высокого порядка. Некоторые датасеты со скрытой симметрией также могут быть выражены алгебраическими структурами.
4. *Топологическая структура.* Термин «топология» употребляется в двух взаимосвязанных значениях: топология как раздел математики, изучающий свойств объектов, которые не изменяются при его непрерывной (т.е. без разрывов и

склеиваний) деформации, и топология как математическая структура, отражающая эти свойства. Например, шарик из пластилина можно непрерывными деформациями перевести в кубик, но нельзя – в бублик; бублик, в свою очередь, можно перевести в чашку с ручкой. Это означает, что первая пара объектов имеет единую топологическую структуру (сфера), а вторая пара – другую (двумерный тор). В данном случае неизменным остается количество «дырок» в объекте – 0 в первом случае и 1 во втором. Как оказывается, выявление таких «дырок» в датасете может дать много полезной информации о его свойствах.

5. *Функциональная структура.* Фундаментальная статистическая проблема заключается в обнаружении функциональной структуры заданного набора точек. Это могут быть линейные функции для линейной регрессии, кусочно-постоянные функции для кластеризации или классификации, кусочно-полиномиальные функции, такие как сплайн-функция, и другие функции, такие как вейвлет-расширение.

1.2.3. Формат данных

Данные, используемые в DS-проекте, разделяются по степени формализованности на три базовых типа (рис. 1.2).

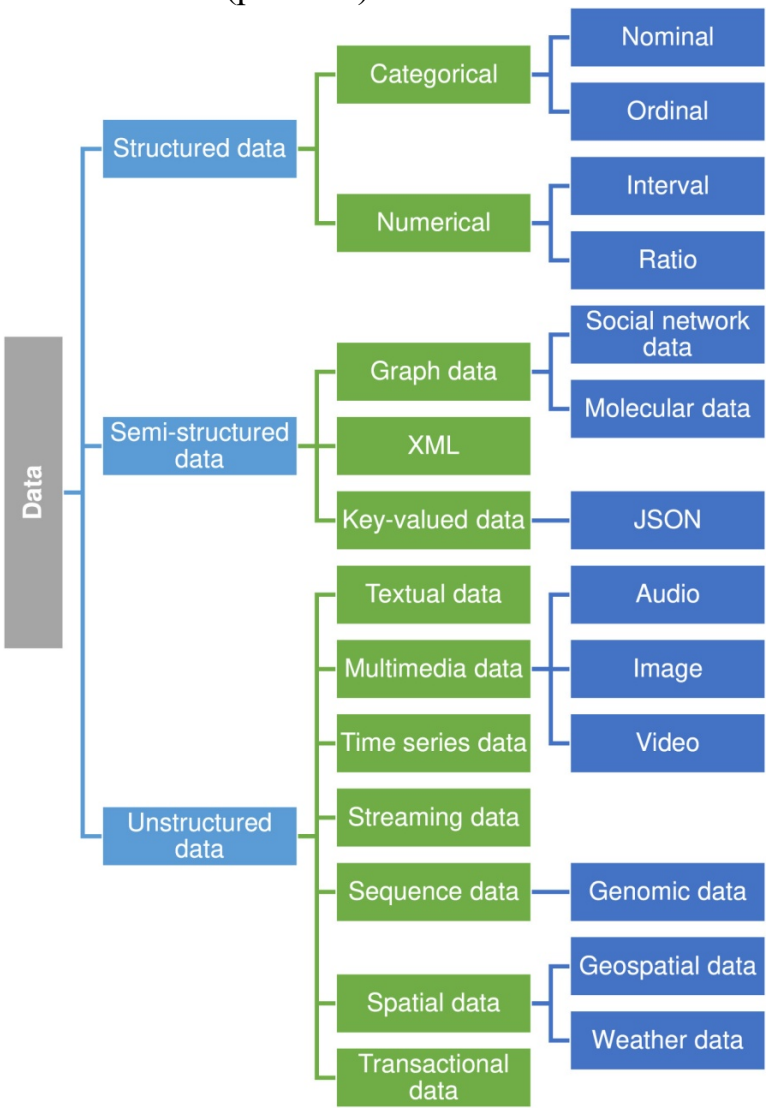


Рис. 1.2. Классификация типов данных [Cunha]

– *Структурированные данные*. Данные, которые могут храниться, быть доступными и обработанными в форме с фиксированным форматом, называются структурированными. Они имеют заранее определенный формат.
– *Полуструктурированные,* или *слабоструктурированные данные* – это данные, зачастую собранные из различных источников. Структура данных документирована, но в зависимости от источника данных конкретный формат представления информации может быть разным. Слабоструктурированные данные требуют обязательной обработки и последующей валидации перед использованием. Например, данные о координатах и скоростях молекул, в которых некоторые координаты пропущены или некоторые записи повторяются, являются полуструктурированными. Нужно понять, почему так произошло, и перед использованием либо исключить такие данные (что может привести к систематической ошибке), либо, исходя из модели данных, восстановить пропущенные значения.
– *Неструктурированные данные* – это данные, в которых координаты измеряются в разных единицах, числа иногда записаны словами, иногда латинскими цифрами, а иногда в виде сканированного изображения почерка лаборанта. Типичный пример неструктурированных данных — гетерогенный источник, содержащий комбинацию простых текстовых файлов, картинок и видео.

Как видно на рис. 1.2, данные сначала делятся на структурированные, полуструктурированные и неструктурированные. Внутри каждого типа структурированные данные делятся на категориальные и числовые данные, полуструктурированные данные делятся на графические данные, XML и данные с ключевыми значениями, а неструктурированные данные делятся на текстовые, мультимедийные, временные ряды, потоковые, последовательные, пространственные и транзакционные данные.

Выделяют также специальные типы данных:
– *Данные на естественном языке* составляют отдельную разновидность неструктурированных данных, для которых ставятся и решаются специальные задачи DS;
– *Машинные данные* – информация, автоматически генерируемая компьютером или другим устройством без вмешательства человека, например, данные Интернета вещей (IoT), журналы веб-серверов, журналы сетевых событий и телеметрии и пр.;
– *Потоковые данные* поступают в систему при возникновении некоторых событий, а не загружаются в хранилище данных большими массивами, при этом они могут принимать почти любую из вышеперечисленных форм – например, прямые трансляции спортивных мероприятий, данные биржевых котировок;
– *Транзакционные данные* отображают результат выполнения каких-либо операций – например, данные о взаимодействии молекул между собой (а именно о пересечении границ рассматриваемой области пространства), о траектории конкретной молекулы, об испарении капель дождя;

– *Ссылочные данные* (или: мастер-данные, основные данные, нормативно-справочная информация) – это базовые неизменяемые данные, заранее известные из внешних источников, такие как нормативы, сокращения, акронимы, словари, стандарты.

Некоторые примеры специальных типов неструктурированных данных представлены на рис. 1.3.

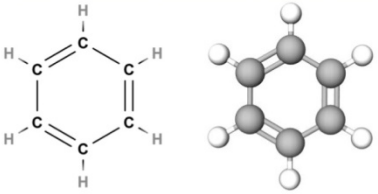
<table>
<tr><th>Data Type</th><th colspan="2">Example</th></tr>
<tr><td>Genomic data</td><td colspan="2">CGTAGGACTGAGGTTAAACCCCGG
AACAACCTGGTTACCGTACGCCCC
TCATGCGTAGATCGATCCAGACTA
GTACTACGTACGGACTGTACCGAT
TGGACCGTTTAACATTGGACCTAC
CTTGGCCAATTAACCGGTTAACCG
AACCGGTTACGTGTACGTACGATA</td></tr>
<tr><td rowspan="5">Transactional data</td><td>Name</td><td>Items</td></tr>
<tr><td>John</td><td>Milk, Bread, Viagra</td></tr>
<tr><td>Mary</td><td>Bread, Pregnancy test</td></tr>
<tr><td>Bob</td><td>Wine, Cream, Viagra</td></tr>
<tr><td>Alice</td><td>Wine, Pregnancy test</td></tr>
<tr><td>Geospatial data</td><td colspan="2">09:54; 37.76045; -122.4898
09:52; 37.75994; -122.48978
09:51; 37.75951; -122.48974
09:50; 37.75925; -122.49034</td></tr>
<tr><td>Molecular data</td><td colspan="2">  </td></tr>
</table>

Рис. 1.3. Примеры неструктурированных данных [Cunha]

1.2.4. Большие данные и специфика их анализа в DS

Большие данные (Big Data) – массивы информации, которые невозможно обработать или проанализировать при помощи традиционных методов с использованием человеческого труда и настольных компьютеров. Особенность Big Data – еще и в том, что массив данных со временем продолжает экспоненциально расти, поэтому для оперативного анализа собранных материалов необходимы вычислительные мощности суперкомпьютеров. Соответственно, для обработки Big Data необходимы экономичные, инновационные методы обработки информации и предоставления выводов. Таким образом, специфика данной сферы – это инструменты и системы, способные выдерживать высокие нагрузки.

Проблематика Big Data выделилась как самостоятельное направление в рамках ИТ в связи с тем, что появилась потребность в обработке слабо формализуемых и неформализуемых данных.

В эпоху до машинного обучения и ИИ информационной базой для выполнения компьютерных программ были БД. В них информация жестко подчиняется правилам логики первого порядка, соответственно, требуется целостность данных и непротиворечивость содержимого БД. То, что не вставляется в БД, при таком подходе не подлежало обработке.

Однако хотелось выйти за рамки этих жестких требований. Пример – создание систем беспилотного управления автомобилем. Чтобы программа взяла управление автомобилем полностью на себя, требовалось заложить в нее знание обо всех ситуациях на дороге. Это невозможно: мы не можем заранее предсказать, как поведут себя все автомобили на дорогах, как будут выглядеть все пешеходы и т.д. Следовательно, мы не можем прописать четкие сценарии поведения для любых ситуаций на дороге и записать их в формальную БД. С другой стороны, водителей-людей готовят самих принимать решения на дороге, опираясь на целый набор информации: правила дорожного движения, дополнительные факторы, которые есть на дороге, и свой прошлый, накопленный в поездках уникальный опыт.

Таким образом, для работы с неформализуемыми данными требуется решить две проблемы:

- применить data-driven подход, т.е. научить программу саму выводить правила и закономерности, опираясь на данные, знания и опыт из прошлого. Эта проблема решается в рамках машинного обучения и, в более продвинутом варианте, искусственного интеллекта;
- предоставить компьютеру такие данные, из которых он может сформировать контекст, необходимый для решения задачи. Лобовое решение – сохранять как можно больше данных, а потом из них выбирать необходимое. Это и есть подход Big Data. (Заметим, что активно разрабатываются и нелобовые решения, когда объективно данных не хватает, а контекст нужен).

Таким образом, под Big Data понимают:

- всю цифровую информацию, которая накапливается или генерируется в ходе анализируемого процесса;
- подходы к работе с массивами таких неструктурированных данных.

Набор признаков для Big Data был впервые предложен в 2001 году с целью указать на равную значимость управления данными по трем аспектам: экстремальный объем (Volume), разнообразие типов (Variety) и скорость обработки данных (Velocity). В дальнейшем появились интерпретации с четырьмя V (veracity – достоверность), пятью V (viability – жизнеспособность и value – ценность), семью V (variability – переменчивость и visualization – визуализация):

- Объем (Volume) – количество сгенерированных и хранящихся данных. Размер данных определяет значимость и потенциал данных, а также то, могут ли они быть рассмотрены как большие данные. Подчеркнем, что рост объема данных может быть связан не только с количеством экземпляров в датасете (например, количеством страниц в сети Интернет), но и с их размерностью. Например, типичными для данных по экспрессии генов являются датасеты с наблюдениями не более чем по нескольким сотням независимых

выборок (субъектов) и с информацией о десятках или сотнях тысяч генов на каждую выборку.

- Разнообразие (Variety) – тип данных. Разнообразие больших данных проявляется в их форматах: структурированные цифры из клиентских баз, неструктурированные текстовые, видео- и аудиофайлы, а также полуструктурированная информация из нескольких источников. Если раньше данные можно было собирать только из электронных таблиц, то сегодня данные поступают в разном виде: от электронных писем до голосовых сообщений. Большие данные при сопоставлении друг с другом могут дополнять отсутствующие данные.
- Скорость (Velocity). Здесь подразумевается скорость, с которой данные генерируются и обрабатываются. Очень часто большие данные используются в режиме реального времени.
- Изменчивость (Variability) – противоречивость, непостоянство наборов данных, что может препятствовать их обработке и управлению.
- Достоверность (Veracity) – качество данных напрямую влияет на точность проведения анализа данных.

Сравнение подходов традиционной аналитики и аналитики больших данных представлено в табл. 1.1.

Таблица 1.1.

Традиционная аналитика	**Big data аналитика**
Постепенный анализ небольших пакетов данных	Обработка сразу всего массива доступных данных
Редакция и сортировка данных перед обработкой	Данные обрабатываются в их исходном виде
Старт с гипотезы и ее тестирования относительно данных	Поиск корреляций по всем данным до получения искомой информации
Данные собираются, обрабатываются, хранятся и лишь затем анализируются	Анализ и обработка больших данных в реальном времени, по мере поступления

1.2.5. Стандарты качества данных

Качество данных – комплексная характеристика, которую присваивают источникам или наборам данных для их сравнения и использования в конкретных целях. Существует много определений качества данных, но, в общем, качество данных – это оценка того, насколько данные пригодны для использования и соответствуют его контексту обслуживания. Нельзя говорить о качестве данных в отрыве от цели их использования – эта характеристика строится с учетом множества параметров, начиная от таких простых, как объем данных, и заканчивая такими сложными, как стилистика текста на естественном языке.

Чтобы получить ясные и одинаково интерпретируемые критерии качества данных, была разработана и продолжает развиваться серия международных стандартов качества данных ISO 8000. На сегодняшний день опубликованы и применяются следующие части стандарта ISO 8000:

ISO 8000-1:2011 Data quality – Part 1: Overview

ISO 8000-100:2016 Data quality – Part 100: Master data: Exchange of characteristic data: Overview

ISO 8000-110:2009 Data quality – Part 110: Master data: Exchange of characteristic data: Syntax, Semantic encoding, and conformance to data specification

ISO 8000-115:2018 Data quality – Part 115: Master data: Exchange of quality identifiers: Syntactic, semantic and resolution requirements

ISO 8000-120:2016 Data quality – Part 120: Master data: Exchange of characteristic data: Provenance

ISO 8000-130:2016 Data quality – Part 130: Master data: Exchange of characteristic data: Accuracy

ISO 8000-140:2016 Data quality – Part 140: Master data: Exchange of characteristic data: Completeness

ISO 8000-150:2011 Data quality – Part 150: Master data: Quality management framework

ISO 8000-2:2017 Data quality – Part 2: Vocabulary

ISO 8000-311:2012 Data quality -- Part 311: Guidance for the application of product data quality for shape (PDQ-S)

ISO 8000-60:2017 Data quality – Part 60: Data quality management: Overview

ISO 8000-61:2016 Data quality – Part 61: Data quality management: Process reference model

ISO 8000-8:2015 Data quality – Part 8: Information and data quality: Concepts and Measuring

В настоящий момент продолжается работа над следующими частями стандарта:

ISO 8000-62 Data quality -- Part 62: Data quality management: Organizational process maturity assessment: Application of the ISO/IEC 330xx family of standards

ISO 8000-63 Data quality -- Part 63: Data quality management: Process measurement

ISO 8000-64 Data quality -- Part 64: Data quality management: Organizational process maturity assessment: Application of the Test Process Improvement method

ISO 8000-65 Data quality -- Part 65: Data quality management: Process measurement questionnaire.

Стандарты группы ISO 8000 базируются на следующих общих принципах:

- качество применимо к данным, имеющим определенное назначение, учитывающимся при принятии какого-либо решения;
- качество данных затрагивает нужные и подходящие данные, уместные в подходящем месте в подходящее время;
- качество данных отвечает требованиям потребителя;
- качество данных предотвращает повторение дефектов данных и сокращает избыточные расходы.

Из группы ISO/TS 8000 в Российской Федерации локализованы ГОСТ Р 56214-2014/ISO/TS 8000-1:2011 «Качество данных. Часть 1. Обзор»[1] и ГОСТ Р ИСО 8000-2-2019 «Качество данных. Часть 2. Словарь»[2].

ГОСТ Р 56214-2014/ISO/TS 8000-1:2011 содержит более 20 спецификаций, к которым сейчас активно добавляются новые:

a. части 1–99: «Качество общих данных»;

b. части 100–199: «Качество основных данных»;

c. части 200–299: «Качество данных в транзакциях»;

d. части 300–399: «Качество данных о продукции».

В качестве примера приведем фрагмент стандарта ГОСТ Р ИСО 8000-110-2011. «КАЧЕСТВО ДАННЫХ. Часть 110. Основные данные. Обмен данными характеристик. Синтаксис, семантическое кодирование и соответствие спецификации данных», который иллюстрирует правильное и неправильное описание данных.

Если бит является основным стандартным блоком запоминаемых или хранимых электронных данных, то значение свойства является основным стандартным блоком хранимых электронных данных, относящихся к характеристикам. Значение данных само по себе либо не имеет смысла, либо является неоднозначным. Как правило, значение данных обозначается названием какого-либо свойства для того, чтобы точнее сформулировать элемент данных.

Неправильно: Деталь, имеющая форму О-образного уплотнительного кольца и производимая в компании ABC, имеет номер 94117. Высота поперечного сечения этой детали от 0,34 см (0,135 дюйма) до 0,36 см (0,143 дюйма), диаметр центрального отверстия от 39,29 см до 39,45 см, твердость от 60,0 до 70,0 единиц дюрометра (твердомера) A, кольцо выполнено из каучукового бутадиен-акрилонитрила в соответствии с AMS 7271, ссылочный номер проекта - 3003489P1.

Правильно: Эта формулировка, изложенная понятным языком, может быть разложена на элементы, представленные в таблице 1. В первую колонку включены обозначения, а во вторую - значения данных.

Таблица 1.2 - Данные характеристик О-образного уплотнительного кольца

Данные	Значение данных
Высота поперечного сечения	0,135 дюйма - минимум, 0,143 дюйма - максимум
Диаметр центрального отверстия	15,470 дюйма - минимум, 15,530 дюйма - максимум
Уровень твердости	60,0 единиц дюрометра - минимум, 70,0 единиц дюрометра - максимум
Материал	Каучуковый бутадиен-акрилонитрил
Материал и классификация соответствуют документу	AMS 7271
Код производителя	94117
Проектная ссылка	3003489P1

[1] https://docs.cntd.ru/document/1200114769

[2] https://docs.cntd.ru/document/1200169126

ГОСТ Р ИСО 8000-2-2019 «Качество данных. Часть 2. Словарь» определяет основную терминологию в области качества данных. Приведем некоторые выдержки из него:

Качество данных (data quality) – степень, с которой набор характеристик, присущих данным, отвечает конкретным требованиям с точки зрения их применения. Неправильно выстроенные уровни качества данных непосредственно влияют на успех проекта: можно либо задать слишком высокий уровень и не достигнуть его, либо установить слишком низкий уровень и тогда будет потерян смысл системы аналитики.

Управление качеством данных (data quality management, DQM) – согласованная деятельность по контролю и управлению структурой, имеющей непосредственное отношение к качеству данных, обеспечение соответствия данных целям их использования с поддержанием полноты, точности, корректности и своевременности.

Совокупность (набор данных, data set) – логически значимая группа данных.

Метаданные (metadata) – данные, определяющие и описывающие другие данные.

Верификация (verification) – подтверждение посредством представления объективных свидетельств того, что установленные требования выполнены.

Авторитетный источник данных (authoritative data source) – владелец процесса, производящего данные.

Утвержденное эталонное значение (accepted reference value) – значение, применяемое в качестве согласованной ссылки при сравнении данных (реестр).

Истинное значение (true value) – значение параметров характеристики какого-либо объекта в определенных условиях.

Согласно стандарту:

3.8.1 качество данных (data quality): Степень, с которой набор характеристик, присущих данным (3.2.2), отвечает требованиям (3.1.2).

3.2.2 данные (data): Интерпретируемое представление информации (3.2.1) в соответствующей форме, удобной для передачи, интерпретации и обработки.

3.1.2 требование (requirement): Потребность или ожидание, которое установлено, предполагается или является обязательным.

Таким образом, стандарт формально определяет качество данных как степень, с которой набор характеристик, присущих данным, отвечает конкретным требованиям с точки зрения их применения.

1.2.6. Метрики качества данных

Для количественной оценки состояния данных в организации, процессе и пр. на основе стандартов и требований разрабатываются метрики качества данных. Метрики качества данных – это конкретные индикаторы, используемые для оценки того, насколько хорош или плох набор данных – другими словами, соответствует ли набор данных поставленной цели. Состав и структура метрик качества данных зависят от типа организации и реализуемого процесса DS, но

есть и общие подходы к их отбору. Например, в [Shahid] предлагается разделить возможные метрики на группы, которые оценивают данные по четырем ключевым измерениям:

- Внутреннее измерение – описывает достоверность, объективность и репутацию данных;
- Контекстуальное измерение – отражает актуальность, своевременность и полноту данных;
- Представительское измерение – уделяется основное внимание форматированию и представлению данных;
- Универсальный доступ – обеспечивает простоту доступа к данным.

Метрики качества данных могут различаться в зависимости от сектора и предполагаемого использования данных. Однако некоторые показатели обычно применяются во многих отраслях из-за их фундаментальной важности для оценки работоспособности данных. Ниже приводятся некоторые часто используемые примеры показателей качества данных.

- Точность (Accuracy) означает, что информация верна и отражает реальную ситуацию. Неточная информация может вызвать значительные проблемы с серьезными последствиями. Например, если в банковском счете клиента есть ошибка, это может быть связано с тем, что кто-то получил к нему доступ без его ведома.
- Надежность (Reliability) в широком смысле означает, что информация получена из проверенных источников. В области качества данных надежность означает, что часть информации не противоречит другой части информации в другом источнике или системе. Для количественного выражения надежности данных можно использовать показатель согласованности (Consistency Score). Показатель согласованности можно измерить, установив пороговое значение, которое указывает величину разницы, которая может существовать между двумя наборами данных. Если информация совпадает, говорят, что она непротиворечива. Как правило, для устранения несоответствий в нескольких системах данных используются жесткие стратегии интеграции данных.
- Релевантность (Relevance) имеет двоякое значение как характеристика качества данных. С одной стороны, если вы собираете нерелевантную информацию, вы тратите время и деньги. С другой стороны, отсутствие релевантной информации не позволит полноценно решить поставленную задачу. Очевидно, что релевантность данных является проблемно-ориентированной характеристикой, и чаще всего этот показатель определяется итеративно в ходе решения бизнес-задачи или на основании опыта решения аналогичных задач.
- Коэффициент полноты (Completeness Ratio) отвечает на вопрос, в какой степени набор данных содержит все необходимые или ожидаемые элементы данных. Коэффициент полноты измеряет долю полных записей данных по сравнению с общим количеством ожидаемых записей в наборе данных.

Например, в базе данных клиентов предусмотрена такая информация о клиентах, как имя, адрес, адрес электронной почты и номер телефона. Если

база данных содержит одно или несколько пропущенных полей, то коэффициент полноты будет ниже, что указывает на более низкое качество данных.

- Затраты на хранение данных (Costs of Data Storage). Иногда стоимость хранения данных продолжает расти, а объем полезных данных остается прежним. Это происходит из-за избыточности, дублирования и несоответствий в наборах данных и является признаком некачественных данных. Неработоспособные данные также усложняют процессы резервного копирования и восстановления, поскольку в случае потери данных поиск и восстановление точных данных становится затруднительным. И наоборот, если операции с данными остаются постоянными, но наблюдается снижение затрат на хранение данных, скорее всего, данные имеют высокое качество.
- Соотношение данных и ошибок (Ratio of Data to Errors). Коэффициент ошибок – это мера, равная доле неправильных записей в наборе данных по сравнению с общим количеством записей. Предположим, у вас есть список из 1000 адресов, и 100 из них содержат ошибки, такие как неправильные почтовые индексы или названия городов с ошибками. Коэффициент ошибок составит 0.10 или 10%. Указывая процент ошибочных данных, коэффициент ошибок помогает выявить проблемные области в хранилище данных.
- Индекс своевременности (Timeliness Index). Этот показатель качества данных оценивает, насколько быстро данные собираются, обрабатываются и становятся доступными для использования. Для этого он оценивает время, прошедшее между возникновением события и доступностью его данных. Более низкий индекс своевременности предполагает неэффективность или задержки в доставке или доступности данных.
- Объемы темных данных (Amounts of Dark Data). К темным данным относятся данные, которые организация собирает, обрабатывает и хранит, но не использует ни для каких целей. Данные становятся «темными» прежде всего потому, что активно не используются и не управляются. Темные данные могут стать проблемой качества данных, потому что:
  - они могут содержать устаревшую или неточную информацию, что влияет на общую точность и надежность наборов данных вашей компании:
  - они часто включают незащищенную конфиденциальную информацию, что увеличивает риск утечки данных.

  Темные данные не обязательно означают плохое качество данных, но могут указывать на области, где качество данных может быть поставлено под угрозу.
- Коэффициент дублирования (Duplication Rate) измеряет долю повторяющихся записей или записей в наборе данных. Он подтверждает, является ли данная информация в наборе данных уникальной и появляется только один раз. Дублирование в наборах данных можно удалить. Для этого существуют инструменты и алгоритмы дедупликации данных, которые сравнивают записи на основе заранее определенных критериев, таких как пороговые значения сходства, в затем соответственно объединяют или удаляют дубликаты.

1.3. Основные методологии работы с данными

Стремление к повышению эффективности сложных и многогранных процессов DS закономерно привело к необходимости принятия и внедрения единой методологии DS внутри конкретной организации. На сегодняшний день спектр этих методологий достаточно широк, но отправной точкой для их развития послужило появление методологии Knowledge Discovery in Databases (KDD) [Fayyad], как это хорошо видно на рис. 1.4, где показана эволюция стандартизованных методологий, полезных в разработке систем Data Mining.

По результатам опросов портала KDnuggets, проведенного в 2014 году [17], 42% опрошенных компаний использовали методологию CRISP-DM, 10% – методологию SEMMA, 7% – методологию KDD; 6% использовали собственную методологию организации, 28% – свою личную методологию, другими методологиями пользовались 6% опрошенных. Несмотря на прогресс в этой области, распределение лидирующих позиций и сегодня остается стабильным [Plotnikova].

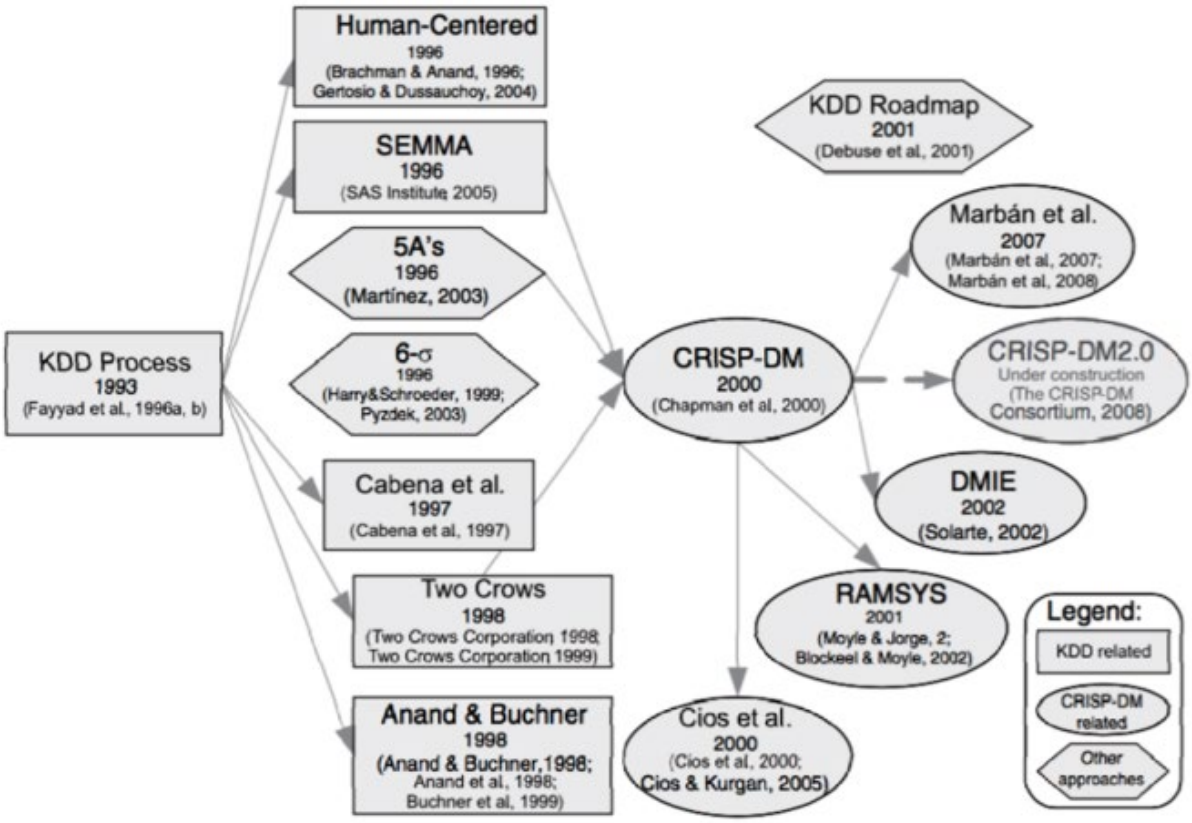


Рис. 1.4. Эволюция стандартизованных методологий в разработке систем Data Mining

1.3.1. Методология KDD process

Согласно определению авторов концепции [Fayyad], KDD представляет собой нетривиальный процесс обнаружения корректных, новых, потенциально полезных и интерпретируемых шаблонов в больших массивах данных.

Здесь под данными понимается множество фактов предметной области, представленных в виде записей базы данных, а шаблон – это выражение на некотором языке, описывающее подмножество данных или применяемую к нему модель. Таким образом, поиск шаблонов подразумевает также подгонку моделей к данным, обнаружение в них зависимостей, закономерностей и структур.

Термин «Процесс» означает, что KDD представляет собой многоэтапную, итеративную процедуру, включающую подготовку данных, построение моделей, оценку и уточнение результатов.

Термин «Нетривиальный» означает, что результаты KDD не должны быть очевидными, а процесс их получения не должен ограничиваться простыми вычислениями, например, средних значений.

Шаблоны, обнаруженные на имеющихся данных, должны быть корректными для любых новых данных предметной области с определенной степенью достоверности. Кроме этого, шаблоны должны были ранее неизвестными и потенциально полезными, то есть позволяющими получить некоторую выгоду при решении определенной задачи. И наконец, шаблоны должны быть понятными и интерпретируемыми, если не сразу, то после некоторой постобработки.

Процесс KDD является интерактивным и итеративным, содержит множество шагов, на каждом из которых пользователь может принимать определенные решения.

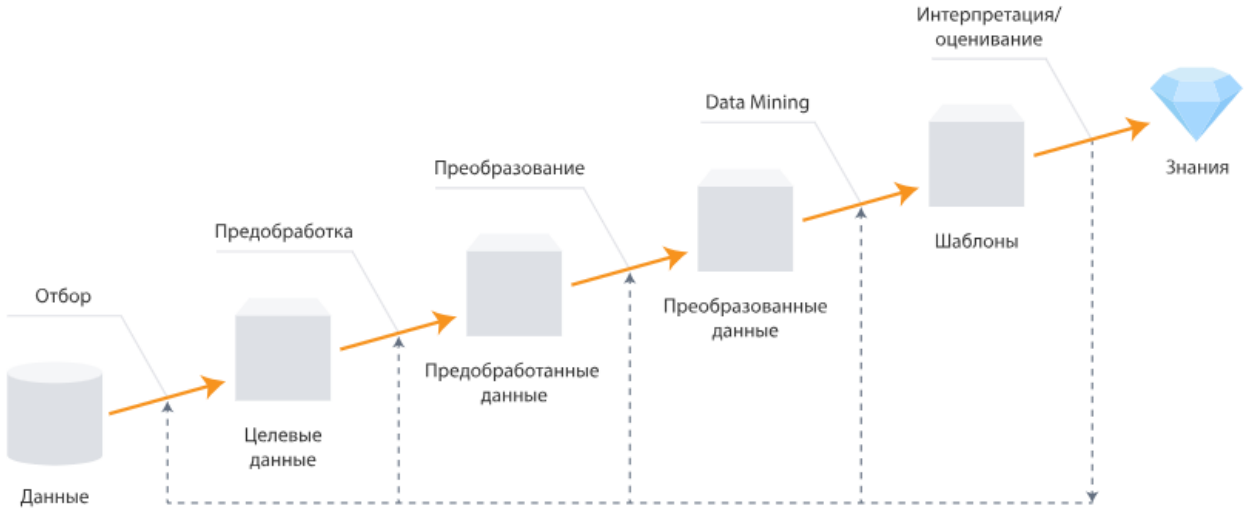


Рис. 1.5. Схема процесса KDD
(источник [https://wiki.loginom.ru/articles/knowledge-discovery-in-databases.html])

В общем случае процесс KDD (рис. 1.5) содержит следующие этапы:

Отбор – происходит понимание и осмысление предметной области и привлечение априорных знаний о ней, формулирование целей и задач процесса KDD, а также формирование целевого набора данных, на котором будет производиться поиск шаблонов.

Предобработка – включает очистку данных, отбор данных для построения моделей, выбор методов обработки пропусков и дубликатов, обработку временных рядов и т.д.

Преобразование (трансформация) – включает сокращение размерности данных, а также определение формы их представления, наиболее оптимальной с точки зрения решаемой задачи.

Data Mining – к отобранным и подготовленным данным применяются аналитическое методы и модели, решающие задачи классификации и регрессии, поиска ассоциативных правил и кластеризации, а также прогнозирования с целью обнаружения шаблонов.

Интерпретация – обнаруженные шаблоны представляются в виде решающих правил и их деревьев, структур кластеров, регрессии и т.д. На их основе формируются соответствующие знания. Обнаруженные знания могут быть использованы непосредственно, объединены со знаниями, полученными из других систем и предметных областей, применены для документирования и формирования отчетов. Также на данном этапе производится проверка наличия потенциальных конфликтов с ранее полученными знаниями и их разрешение.

KDD не предписывает, какие методы и алгоритмы обработки следует использовать при решении конкретной задачи, а определяет последовательность действий, которую необходимо выполнить для того, чтобы из исходных данных получить знания. Этот подход универсален и не зависит от предметной области.

1.3.2. Методология SEMMA

Согласно [Azevedo], SEMMA – это аббревиатура, которая расшифровывается как Sample, Explore, Modify, Model и Assess. Методология SEMMA представляет собой список последовательных шагов, разработанный SAS Institute, одним из крупнейших производителей программного обеспечения для статистики и бизнес-аналитики. Хотя SEMMA часто считается общей методологией добычи данных, SAS утверждает, что это, скорее, логическая организация функционального набора инструментов одного из их продуктов, SAS Enterprise Miner, для выполнения основных задач добычи данных [SAS]. SEMMA в большинстве случаев рассматривается как функциональная методология интеллектуального анализа данных, а не как конкретный инструмент.

Методология SEMMA состоит из пяти этапов (рис. 1.6).

Выборка – этот этап подразумевает выбор подмножества соответствующего набора данных объема из обширного набора данных, который был предоставлен для построения модели. Целью этого начального этапа процесса является определение переменных или факторов (как зависимых, так и независимых), влияющих на процесс. Затем собранная информация сортируется по категориям подготовки и проверки.

Изучение – на этом этапе проводится одномерный и многомерный анализ для изучения взаимосвязанных отношений между элементами данных и выявления пробелов в данных. В то время как многомерный анализ изучает отношения между переменными, одномерный анализ рассматривает каждый фактор по отдельности, чтобы понять его роль в общей схеме. Все факторы влияния, которые могут повлиять на результаты исследования, анализируются с большой опорой на визуализацию данных.

Изменение – на этом этапе уроки, извлеченные на этапе исследования из данных, собранных на этапе выборки, извлекаются с применением бизнес-логики. Другими словами, данные анализируются и очищаются, затем передаются на этап моделирования: здесь же выясняется, требуют ли данные уточнения и преобразования.

Моделирование – после уточнения переменных и очистки данных на этапе моделирования применяются различные методы добычи данных; в результате создается прогнозируемая модель того, как эти данные отражаются в конечном желаемом результате процесса DS.

Интерпретация результата – на этом заключительном этапе модель SEMMA оценивается на предмет ее полезности и надежности для DS-проекта. Теперь данные можно протестировать и использовать для оценки эффективности ее работы.

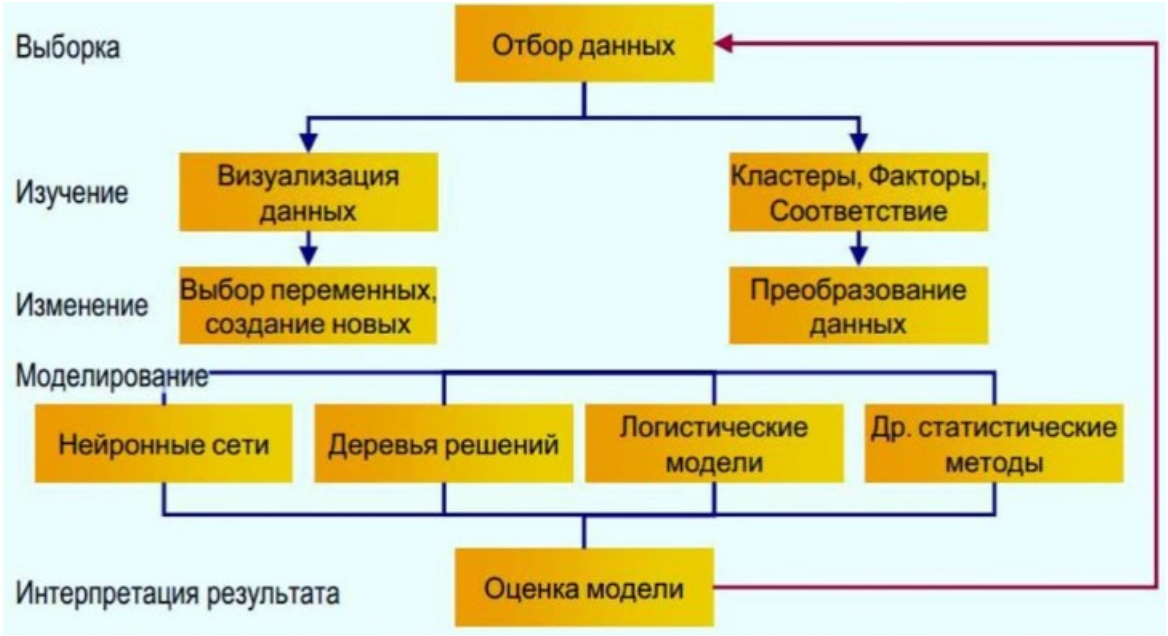


Рис. 1.6. Схема обработки данных в методологии SEMMA
(источник [https://ppt-online.org/131267])

1.3.3. Методология CRISP-DM

Методология CRoss Industry Standard Process for Data Mining (CRISP-DM) [Shearer] позиционируется как стандарт, описывающий общие процессы и подходы к аналитике данных, используемые в промышленных data-mining проектах независимо от конкретной задачи и индустрии. По оценке разработчиков, методология предоставляет следующие преимущества:

- универсальность – не зависит от отрасли, то есть может применяться в различных секторах и бизнес-задачах;
- структурированность – обеспечивает четкую дорожную карту для проектов по обработке данных, гарантируя, что важные шаги не будут упущены;
- итеративность и цикличность – при возникновении проблемы на каком-то этапе можно вернуться к более раннему этапу, т.е. поощряется постоянное совершенствование процесса обработки данных;

– внимание на бизнес-цели компании как исходную точку обработки данных;
– детальное документирование каждого шага, что, по мнению авторов методологии, позволяет менеджменту лучше понимать суть проекта, а аналитикам – больше влиять на принятие решений.

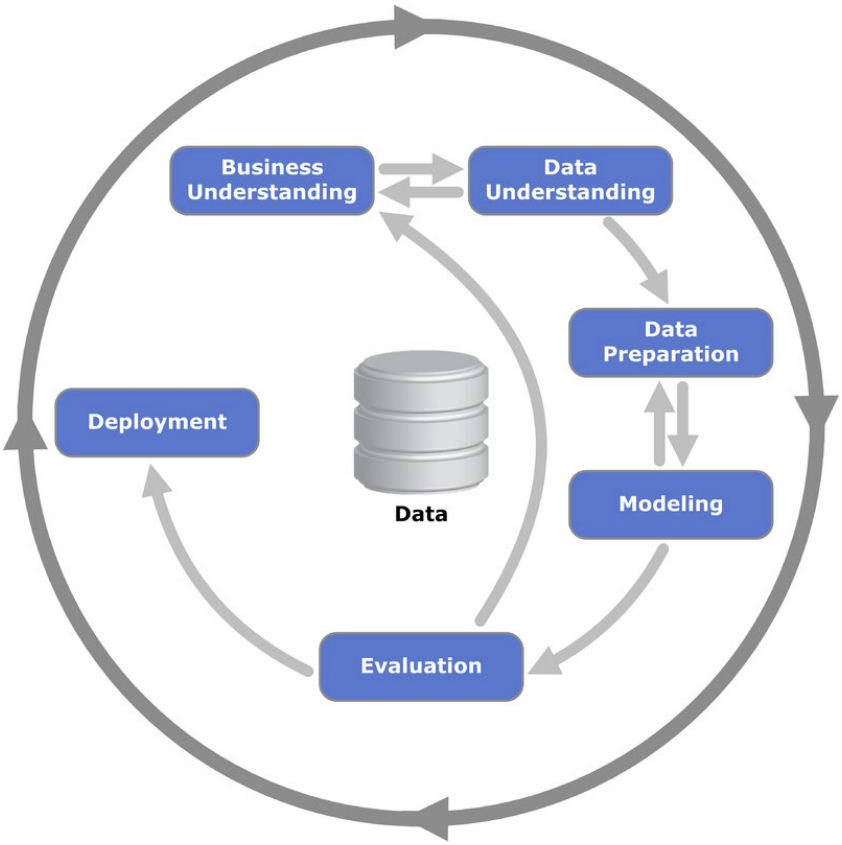


Рис. 1.6. Соотношение отдельных фаз CRISP-DM (источник [https://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining])

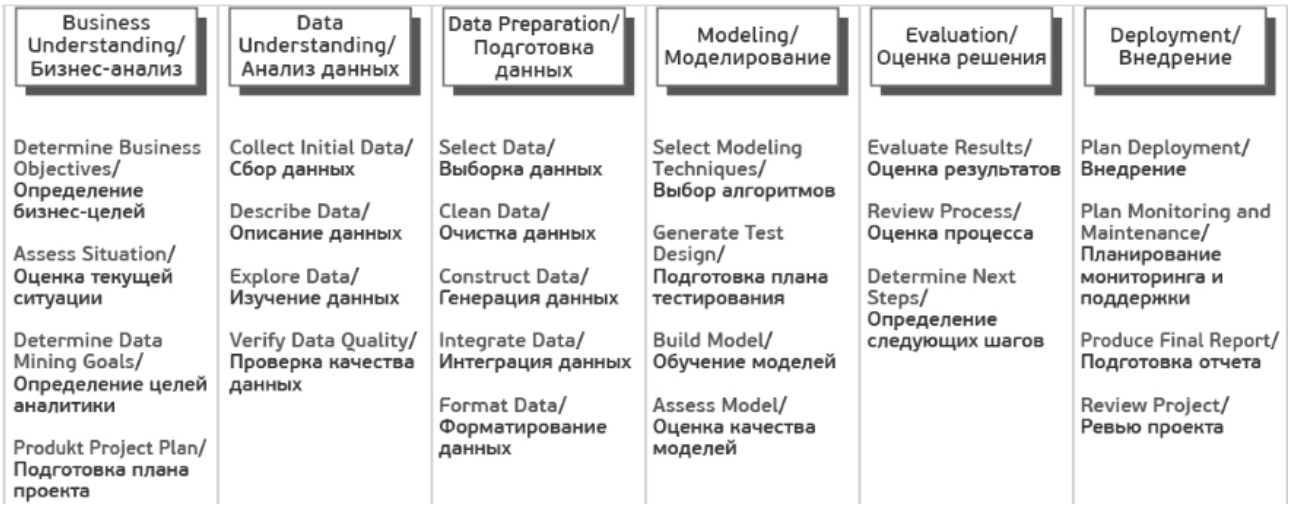


Рис. 1.7. Фазы и этапы CRISP-DM [Коточигов]

Взаимосвязь между основными фазами методологии CRISP-DM представлена на рис. 1.6, а содержание каждой фазы – на рис. 1.7.

Ниже приводятся краткие комментарии по содержанию каждой фазы [Коточигов], а также даются ссылки на разделы пособия, где эти фазы и составляющие их этапы рассматриваются детально.

1. Бизнес-анализ (Business Understanding). В этой фазе выявляются и согласуются с заказчиков основные цели проекта.

1.1. Определение бизнес-целей проекта (Business objectives)

На этом этапе необходимо выяснить и согласовать организационную структуру проекта, бизнес-цель проекта на содержательном и формальном уровне, проанализировать существующие решения и понять, почему они не подходят.

1.2 Оценка текущей ситуации (Assessing situation)

На этом этапе желательно оценить имеющиеся и необходимые ресурсы для проекта, вероятные риски проекта, их соотношение и план действий по их уменьшению, а также получить предварительную оценку ROI (Return On Investment, возврат инвестиций – коэффициент рентабельности инвестиций, который помогает рассчитать окупаемость вложений в проект), в том числе потенциальную выгоду от внедрения инструментов машинного обучения.

1.3. Определение целей аналитики (Data Mining goals)

Необходимо описать задачу в технических терминах, в частности, согласовать метрики оценки результата моделирования и критерий успешности моделирования).

1.4 Подготовка плана проекта (Project Plan)

2. Анализ данных (Data Understanding)

Цель фазы – понять слабые и сильные стороны имеющихся данных, определить их достаточность, предложить идеи, как их использовать, и лучше понять процессы заказчика. В этой фазе используется инструментарий разведочного анализа (см. раздел 4 Пособия), а также инструментарий визуализации данных (см. разделы 2.2.2, 2.2.3, 4.6 Пособия).

2.1 Сбор данных (Data collection)

Цель этапа – понимать, какими данными располагает заказчик и какие данные могут быть добавлены в проект. Необходимо проанализировать все источники, доступ к которым предоставляет заказчик (см. раздел 4.1 Пособия). Если собственных данных недостаточно, возможно, стоит закупить сторонние или организовать сбор новых данных.

2.2 Описание данных (Data description)

На этом этапе производится описание данных во всех источниках. Фиксируются их ключевые атрибуты (таблица, ключ, количество строк, количество столбцов, объем на диске и пр.). Если объем слишком велик для используемого ПО, создается сэмпл данных. Рассчитываются ключевые статистики по атрибутам, такие как минимум, максимум, кардинальность и т.д. (см. раздел 4.3 Пособия), выносится предварительное суждения об их статистическом распределении (см. раздел 4.5 Пособия). Выявляются потенциальные аномалии в данных, такие как наличие или отсутствие выбросов (см. раздел 4.4 Пособия).

2.3 Исследование данных (Data exploration)

С помощью графиков, таблиц и других средств визуализации (см. разделы 2.2.2, 2.2.3, 4.6 Пособия) исследуем данные, чтобы сформулировать гипотезы относительно того, как эти данные помогут решить задачу (см. раздел 2.1.1. Пособия).

2.4 Качество данных (Data quality)

На этом этапе предварительно оцениваются технические и организационные сложности с потенциально полезными данными и объем работы, необходимый для их устранения.

Результатом фазы 2 должна быть фиксация состояния данных – структура, основные статистики и показатели качества потенциально полезного набора данных, а также основные гипотезы по решению поставленной бизнес-задачи.

3. Подготовка данных (Data Preparation)

Подготовка данных – это традиционно наиболее затратная по времени фаза проекта (50–70% времени проекта). Цель фазы – подготовить обучающую выборку для использования в моделировании.

3.1 Выборка данных (Data Selection)

На этом этапе из всего множества потенциально доступных данных отбираются те, которые будут использоваться для обучения модели. Для отбора атрибутов используются различные методы, объединенные в три группы – фильтрационные (filter), встроенные (embedded) и оберточные (wrapper) (см. раздел 5.2 Пособия). Целесообразно также использовать эвристические методы и опыт предыдущих бизнес-проектов.

Важно, что отбор производится как по атрибутам (столбцам датасета), так и по кейсам (сэмплам, или строкам датасета).

3.2 Очистка данных (Data Cleaning)

После отбора потенциально интересных данных нужно проверить и по возможности улучшить их качество, как было запланировано на этапе 2.4 фазы 2 (см. разделы 5.3.1.

3.3 Генерация данных (Constructing new data)

Генерация признаков (feature engineering) – это один из важных этапов в подготовке данных: грамотно составленный признак может существенно улучшить качество модели. Сюда можно отнести агрегацию атрибутов (см. разделы 3.2.2 и 4.3 Пособия), генерацию кейсов (см. раздел 5.3.5 Пособия), нормализацию атрибутов (см. раздел 5.3.4 Пособия), заполнение пропущенных данных (см. раздел 5.3.3 Пособия), анонимизацию данных (см. раздел 5.3.6 Пособия) и другие операции, связанные с формированием новых признаков из уже существующих.

3.4 Интеграция данных (Integrating data)

Если данные загружаются из нескольких источников, то для подготовки обучающей выборки требуется их интеграция (см. раздел 5.4 Пособия).

3.5 Форматирование данных (Formatting Data)

Наконец, нужно привести данные к виду и формату, непосредственно пригодному для моделирования (см. раздел 5.5 Пособия).

Заметим, что в практике работы дата-сайентиста этапы 3.2–3.5 часто не разделяются, а рассматриваются как набор операций, выбираемых в соответствии с конкретной бизнес-задачей.

Результатом фазы 3 должны стать единая аналитическая таблица или другой формат представления данных, пригодные для загрузки в выбранную модель машинного обучения в качестве обучающей выборки.

4. Моделирование (Modeling)

4.1. Выбор алгоритмов (Selecting the modeling technique)

Выбор модели зависит от решаемой задачи, типов атрибутов и требований по сложности (например, если модель будет дальше внедряться в Excel, то RandomForest и XGBoost явно не подойдут).

4.2. Подготовка плана тестирования (Generating a test design)

Традиционный подход – это разделение выборки на 3 части (обучение, валидацию и тест) в примерной пропорции 60/20/20. В этом случае обучающая выборка используется для подгонки параметров модели, а валидация и тест для получения очищенной от эффекта переобучения оценки ее качества. Однако существуют и более сложные стратегии, рассматриваемые в разделе 6.2.

4.3. Обучение моделей (Building the models)

Запускаем цикл обучения и после каждой итерации фиксируем результат. На выходе получаем несколько обученных моделей. Сравниваем полученные модели, фиксируем выявляемые ими закономерности.

4.4 Оценка результатов (Assessing the model)

После того, как был сформирован пул моделей, нужно их еще раз детально проанализировать и выбрать модели-победители. На этом этапе производится оценка моделей по техническим показателям. На выходе желательно иметь список моделей, отсортированный по объективным и/или субъективным критериям, таким как эффективность (определяется метриками), готовность к внедрению, достижение заданных критериев качества. Если критерий успеха не достигнут, то можно либо улучшать текущую модель, либо пробовать новую.

Результатом четвертого шага является построенная математическая модель (model), а также найденные закономерности (findings).

5. Оценка результата (Evaluation)

5.1 Оценка результатов моделирования (Evaluating the results)

На этом этапе производится оценка моделей с точки зрения достижения бизнес-целей, активно взаимодействуя с заказчиком. Для этого результаты формулируются в бизнес-терминах, а также фиксируется получение дополнительной информации, важная для бизнеса – например, выявление нового сегмента рынка.

5.2 Разбор полетов (Review the process)

Желательно проанализировать ход проекта по всем шагам и сформулировать его сильные и слабые стороны.

5.3 Принятие решения (Determining the next steps)

Далее нужно либо внедрять модель, если она устраивает заказчика, либо, если виден потенциал для улучшения, попытаться еще ее улучшить. Если на

данном этапе у нас несколько удовлетворяющих моделей, то отбираем те, которые будем дальше внедрять.

6. Внедрение (Deployment)

Перед началом проекта с заказчиком всегда оговаривается способ поставки модели. В одном случае это может быть просто проскоренная база клиентов, в другом – SQL-формула, в третьем – полностью проработанное аналитическое решение, интегрированное в информационную систему.

На данном шаге осуществляется внедрение модели (если проект предполагает этап внедрения). Причем под внедрением может пониматься как физическое добавление функционала, так и инициирование изменений в бизнес-процессах компании.

6.1 Планирование развертывания (Planning Deployment)

6.2 Настройка мониторинга модели (Planning Monitoring)

Очень часто в проект включаются работы по поддержке решения.

6.3 Отчет по результатам моделирования (Final Report)

По окончании проекта, как правило, пишется отчет о результатах моделирования, в который добавляются результаты по каждому шагу, начиная от первичного анализа данных и заканчивая внедрением модели. В этот отчет также можно включить рекомендации по дальнейшему развитию модели.

Написанный отчет презентуется заказчику и всем заинтересованным лицам. В отсутствие ТЗ этот отчет является главным документом проекта.

1.3.4. Методология BADIR

BADIR (Business question – Analysis plan – Data collection – derive Insights – Recommendations) [Jain] – это фреймворк для поддержки принятия решений в бизнес-задачах. BADIR – это запатентованная платформа для преобразования данных в решения. Методология BADIR (рис. 1.8) включает в себя понимание бизнес-целей и задач, планирование анализа перед получением и обеспечением качества соответствующих данных, применение аналитики для получения информации и потенциального влияния на бизнес-задачи, разработку действенных рекомендаций, соответствующих стратегическим целям, и реализацию решений с одновременным мониторингом результатов и внесением необходимых корректировок.

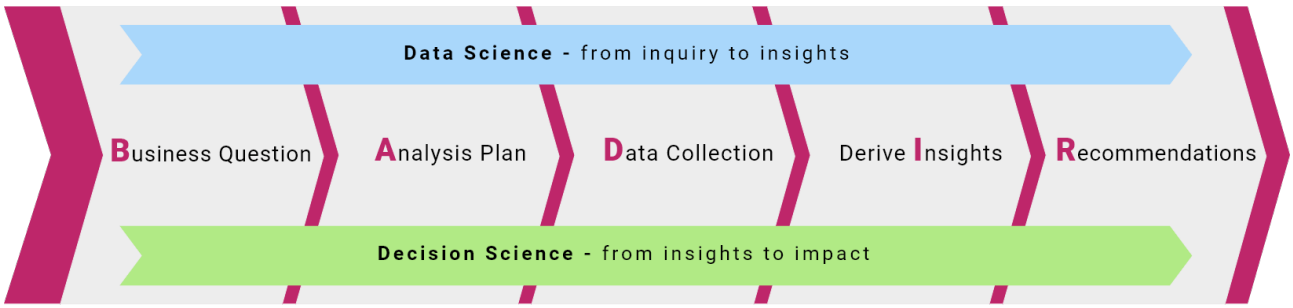


Рис. 1.8. Поток данных в методологии BADIR

Ключевые компоненты BADIR

Методология BADIR состоит из взаимосвязанных этапов, необходимых для принятия решений.

B = Бизнес-вопрос. Первый этап – определение реального бизнес-вопроса. Тенденции рынка, отзывы клиентов, действия конкурентов и т.д. – это источники данных, которые обычно используются компаниями. Однако эти данные сами по себе не помогают командам принимать решения.

A = План анализа. После определения фундаментального бизнес-вопроса следует генерация и проверка гипотез для изучения потенциальных стратегических направлений. Этап плана анализа включает в себя перевод бизнес-целей в конкретные задачи и определение области действия проекта науки о данных. Этот этап гарантирует, что решения основаны не на предположениях, а на проверенных данных.

D = Сбор данных. На основе четкого плана анализа следующим шагом будет сбор необходимых данных. Этот этап имеет решающее значение, поскольку качество данных напрямую влияет на надежность выводов, полученных из них.

I = Вывод идей. Этот этап включает анализ отобранных данных для выявления закономерностей, тенденций и результатов, которые либо подтверждают, либо опровергают предложенные гипотезы.

R = Рекомендации. Последний этап – перевод информации, полученной в результате анализа данных, в практические рекомендации, которые могут помочь в принятии стратегических бизнес-решений.

Специфика методологии BADIR заключается в том, что она делает акцент на понимании ключевых пунктов и целей бизнес-плана, подчеркивая важность согласования проектов науки о данных с бизнес-целями. Кроме того, методология нацелена на быстрое достижение применимых результатов с минимальными затратами ресурсов, т.е. может быть использована для проектов любых масштабов – от личного проекта одного менеджера по продажам до директора крупного торгового центра.

1.3.5. Сравнение методологий анализа данных

Любая методология анализа данных не является универсальным рецептом. Это просто попытка формально описать последовательность действий, которую в той или иной степени выполняет любой аналитик, занимающийся анализом данных. Более того, хотя рис. 1.2 демонстрирует целый «зоопарк» методологий анализа данных, в целом они очень похожи друг на друга. Хотя здесь сложно придумать что-то принципиально новое постоянно появляются новые методологии, имеющие свою специфику – примером может служить методология BADIR.

При выборе методологии для конкретной задачи нужно учитывать сложность проекта. Для сложных проектов комбинированный подход может быть полезным. Кроме того, нужно ориентироваться на существующую у заказчика и у исполнителей организационную культуру и процессы. Некоторые организации могут обнаружить, что одна структура лучше соответствует их существующим практикам, а другие могут выбрать интегрированный подход.

Тем не менее, методология CRISP-DM заслужила популярность как наиболее полная и детальная, применимая к совершенно разным задачам – при предсказании вероятности отклика клиента торговой сети на рекламное предложение, при моделировании оценки кредитоспособности заемщика в коммерческом банке, при разработке сервиса рекомендаций товаров для интернет-магазина, и т.д. По сравнению с ней KDD является более общей и теоретической, а SEMMA – это просто организация функций по целевому предназначению в инструменте SAS Enterprise Miner, которая затрагивает исключительно технические аспекты моделирования, никак не касаясь бизнес-постановки задачи.

Конечно, методология CRISP-DM во многом является избыточной. На практике многие вещи делаются куда менее формально, чем требует методология. В первую очередь это относится к промежуточной отчетности, которую реализует дата-сайентист.

Строгое следование методологии CRISP-DM предполагает написание промежуточного отчета не только после каждой фазы, но и даже после каждого этапа проекта. На практике это требование не обязательно по форме, хотя очень желательно по содержанию. Можно, например, фиксировать результаты в виде аудиофайлов или zoom-файлов с записями резюме митингов команды. Важно подчеркнуть, что должны быть озвучены и осмыслены все полученные на данном этапе результаты, в том числе и отрицательные. Тем самым будут отсеяны неперспективные гипотезы, что позволит сократить затраты времени и ресурсов команды.

В то же время огромным достоинством методологии CRISP-DM является акцент на взаимодействие с заказчиком, в первую очередь на этапах планирования и оценки результата. Методология описывает эти фазы не в виде набора общих слов, а в виде конкретных этапов, каждым из которых ни в коем случае нельзя пренебрегать, несмотря на кажущуюся очевидность и полное понимание со стороны заказчика.

В целом можно рекомендовать методологию CRISP-DM как отличный ориентир для работы дата-сайентиста над конкретным проектом. В то же время нужно помнить, что эта методология, как и все другие, является достаточно общей, а удачные конкретные решения, как правило, находятся в процессе творческого поиска.

Вопросы для самопроверки

1. Дайте определение науке о данных?
2. Какие типы задач DS выделяются в рамках двух подходов: дедуктивное обучение и индуктивное обучение (обучение по прецедентам)?
3. Опишите два положения, принципиально важных для организации процесса DS?
4. В чем специфика анализа большие данные в DS?
5. Перечислите группы метрик оценки качества данных, используемых в DS?
6. Какие существуют методологии в разработке систем Data Mining?
7. В чем специфика и основные преимущества методологии (CRISP-DM)?

8. Что нужно учитывать при выборе методологии для конкретной задачи?
9. На каком этапе методологии CRISP-DM производится оценка моделей по техническим показателям?
10. Какое значение имеет такая характеристика качества данных как Релевантность?
11. Перечислите Акторов процесса DS и их роли?

2. АРТЕФАКТЫ И ИНСТРУМЕНТАРИЙ НАУКИ О ДАННЫХ

2.1. Артефакты науки о данных

Как и любой технологический процесс, DS предполагает создание и использование некоторых базовых объектов, физических или виртуальных, по ходу процесса. В работе [Зыков] эти объекты предложено называть артефактами. Именно их создание знаменует собой достижение цели конкретного этапа процесса DS.

Артефакты процесса DS можно разделить на три вида (рис. 2.1): артефакты аналитики, артефакты машинного обучения и артефакты инженерии данных.

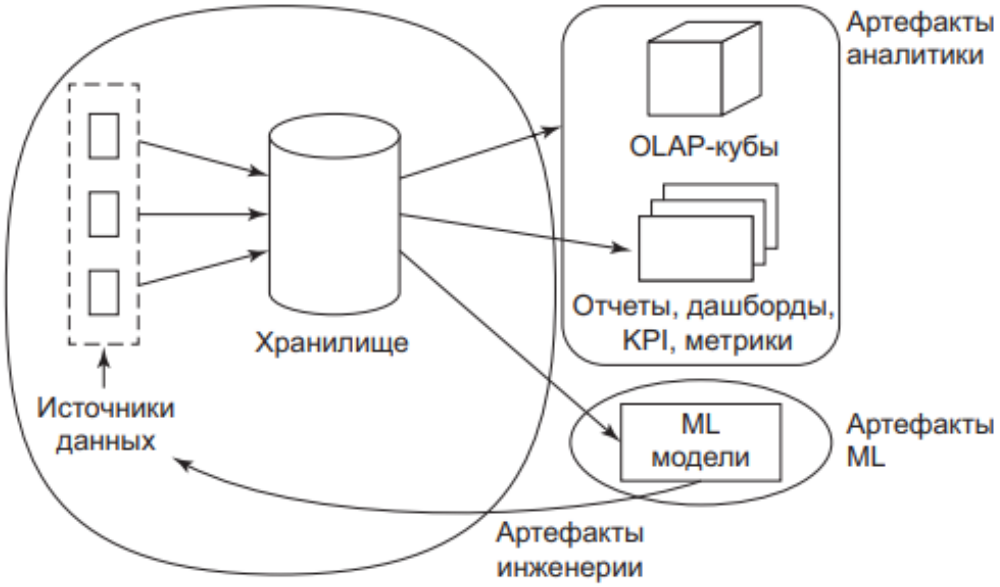


Рис. 2.1. Артефакты DS [Зыков]

2.1.1. Артефакты аналитики

Как следует из методологии процесса DS, очень обобщенно цель этапа аналитики можно обозначить так: создать гипотезы, описывающие нужные закономерности исследуемого бизнеса, и обосновать их справедливость. Соответственно, артефакты аналитики можно разделить на две категории:

- инсайты и гипотезы как виртуальные объекты;
- их материальное представление (в виде отчетов, дашбордов, метрик, результатов визуализации).

Инсайты и гипотезы. Слово «инсайт» (insight) имеет несколько вариантов перевода с английского – понимание, интуиция, проницательность, и даже озарение. Инсайт в DS – это догадка, на основе которой уже формулируется гипотеза. В [Зыков] предлагается несколько направлений мысли для формирования первых инсайтов:

- Обращаемся к истории процесса. Не происходило ли что-нибудь подобное раньше? Если да, то какие тому были причины. Как правило, это самая первая и самая вероятная гипотеза.

– Обращаемся к бизнес-контексту. Не происходило ли каких-либо неординарных событий? Часто как раз параллельные события влияют на возникновение проблемы. Еще плюс пара гипотез.
– Обращаемся к визуализации. Смотрим данные в аналитической системе (например, кубах OLAP) – не видно ли каких-либо аномалий на глаз? Например, какие-либо распределения изменились во времени (типы клиентов, структура продаж и т. д.).

С другой стороны, как отмечается в [Kozyrkov], здесь легко впасть в подтверждение предвзятости (confirmation bias): «…если в данных двадцать историй, вы заметите только ту, которая поддерживает то, во что вы уже верите... и пропустите остальные». Поэтому именно в удачных догадках и проявляется искусство специалиста по данным.

На основании догадок создаются гипотезы в форме закономерностей, описывающие потенциально возможные причины проблемы. Список гипотез по улучшению конкретного продукта называют бэклогом (backlog). Он является важным стратегическим элементом развития компании. Гипотезы сортируются по вероятности реализации или по стоимости проверки, а затем проверяются статистическими методами в порядке убывания.

OLAP-кубы. Куб данных (Datacube) – это многомерный массив данных, которые необходимо организовать и смоделировать для анализа бизнес-объекта или бизнес-процесса. Его измерениями (осями) служат основные атрибуты анализируемого бизнес-процесса. «Разрезая» куб по разным направлениям, можно получить сводные (например, по годам), или детальные (по дням) отчеты.

Термин OLAP (On Line Analytical Processing) исторически имел значение набора технологий (OLAP-серверов) для поддержки многомерных кубов, но сегодня его значение расширилось до концепции многомерного анализа. Несмотря на прогресс технологий хранения данных, концепция OLAP-кубов попрежнему актуальна [OLAP], так как представляет собой естественную, интуитивно понятную модель данных. Используя кубы и операции OLAP, такие как срезы, кубики, свертывание, детализация и сводные данные, бизнес-аналитики могут создавать довольно сложные аналитические модели.

Отчеты и дашборды. Отчет – статичный документ, который содержит данные с выводами о проведенном эксперименте. Как правило, отчет содержит графики, несколько таблиц или диаграмм. Для построения отчета достаточно Excel или Google Docs.

Дашборд (панель мониторинга) – информационная панель с важной информацией, сгруппированной на одном экране, где можно отслеживать данные в реальном времени. Дашборд содержит несколько модулей (графики, отчеты и набор данных), отражающих самые важные показатели. Другими словами, дашборд – это система, которая не только показывает данные, но и способна анализировать их; в нее подтягивают показатели из систем аналитики, таблиц, CRM, социальных сетей и баз данных.

Эта многогранность имеет и обратную сторону: разработка дашбордов – отдельная задача, которая может стоить больших денег.

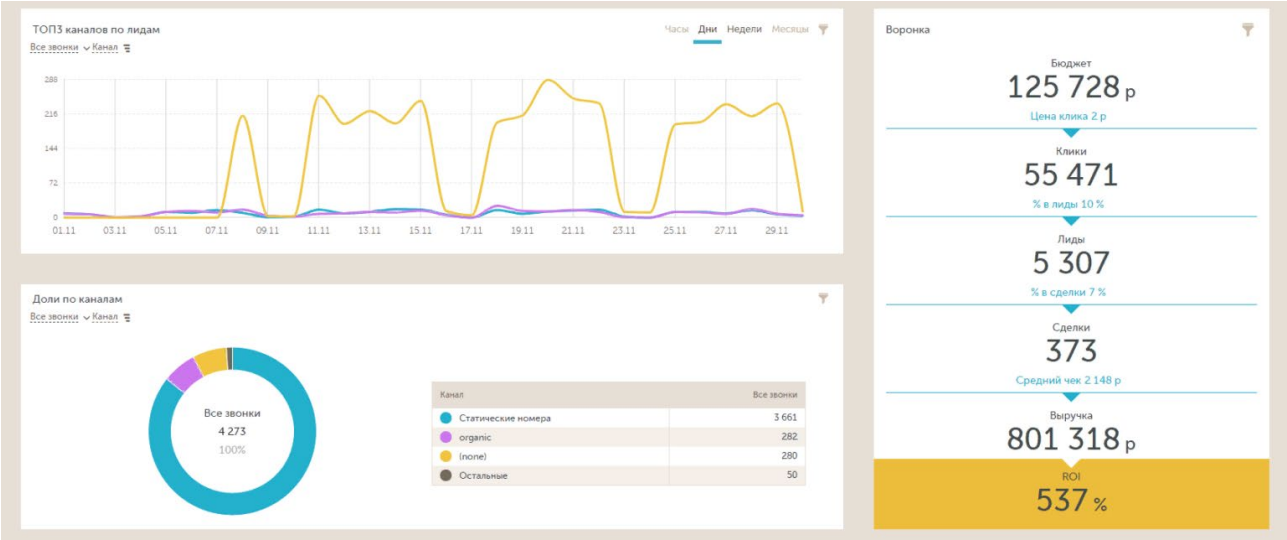


Рис. 2.1. Пример дашборда с анализом ROI (источник [SendPulse])

По мнению [Зыков], «никакой дашборд не заменит интерактивный анализ, для которого нужны соответствующая аналитическая система (SQL, OLAP, Google Data Studio, Tableau) и знание контекста. Мы никогда не сможем придумать ограниченный набор отчетов, которые будут отвечать на вопрос «почему». Поэтому я всегда за лаконичные автоматические отчеты, которые будут отвечать на два вопроса: есть ли проблема и где она возникла. Если проблема есть, нужно лезть в интерактивные системы анализа данных».

Метрики и KPI. Метрика – это цифра, которая характеризует процесс безотносительно его цели. Ключевой показатель (key performance indicator, KPI) – это индикатор, который показывает, насколько далеко процесс находится от цели. Если в компании продукт и бизнес-процесс не устоялись, для нее крайне сложно придумать KPI, и приходится пользоваться метриками.

Отбор метрик и, тем более, разработка KPI требуют от специалиста по данным высокой квалификации. Например, одна из первых метрик для оценки эффективности интернет-магазина – конверсия, т.е. отношение числа посетителей сайта, выполнивших на нём какие-либо целевые действия (покупку, регистрацию и пр.), к общему числу посетителей сайта. Но для более полного анализа экономики интернет-магазина этого недостаточно, и нужны другие метрики: средняя выручка на посетителя сайта, средняя стоимость заказа, среднее число товаров в заказе, маржа и пр. Одновременно эти показатели можно делить по верхним категориям товаров и группам пользователей (если достаточно данных).

2.1.2. Артефакты машинного обучения, инженерии данных, DevOps и MLOps

Артефактом машинного обучения является отобранная модель машинного обучения, обученная и протестированная на требуемых датасетах, в виде документированного программного кода.

Артефактами инженерии данных являются архитектура аналитической системы и реализующий ее программный код. В свою очередь, архитектура ана-

литической системы создается, документируется и реализуется на трех уровнях [Зыков]:

- физический – серверы и каналы связи между ними;
- уровень данных – хранилища данных;
- приложения – программы, с помощью которых пользователи получают доступ к данным, а также публикуют модели ML.

В идеале этих двух артефактов достаточно, чтобы развернуть (подготовить) аналитическую систему за минимальное время. В крутых реализациях это можно сделать автоматически, нажатием одной кнопки. Это очень важно для устойчивой работы аналитической системы.

В реальности аналитическая система для реализации проекта DS собирается из отдельных «кирпичиков», и ее устойчивую работу обеспечивают специальные люди – DevOps-инженеры [SkillFactory]. Название произошло от двух сокращений: Dev – development (разработка) и Ops – operations (поддержка). До появления подхода DevOps члены команды ИТ-проекта часто работали вразнобой, и это приводило к замедлению работы, нарушению процессов и множеству незамеченных ошибок. Для внесения любого обновления проект приходилось останавливать и пересобирать вручную. Это было неэффективно. Идея DevOps выросла из гибкой методологии Agile, подразумевающей непрерывную разработку маленькими итерациями. Основная задача DevOps-инженера заключается в следующем: так настроить систему, чтобы все изменения в нее можно было вносить непрерывно, «бесшовно», без падений, наблюдаемо и т.д.

В подходе DevOps используются пять практик:

- непрерывная интеграция, поставка и развертывание, или CI/CD (continuous integration and continuous delivery/deployment). Разработчик не обновляет код вручную и не собирает его своими руками. Он отправляет его в специальный репозиторий, где тот сам собирается и интегрируется в другие части кода, а нужные участки обновляются. После сборки продукт автоматически разворачивается в какой-то тестовой среде для проверки. После тестирования он так же автоматически отправляется на рабочий сервер, где разворачивается и начинает работать. В результате внесение изменений происходит в фоновом для пользователя режиме, т.е. устраняются простои продукта;
- непрерывное тестирование. После развертывания на тестовом сервере программа проверяется автоматически. Запускаются юнит-тесты. Если программа их не пройдет, ее автоматически отправляют на доработку;
- непрерывный мониторинг. В автоматическом режиме показатели развернутого продукта, такие как нагрузка на процессор и оперативную память, использование пространства на диске, исполнение политики безопасности и действия пользователей, непрерывно контролируются и выводятся на дашборд для анализа;
- микросервисы – архитектура продукта, согласно которой он разбит на множество мелких и независимых друг от друга модулей. В результате при добавлении нового микросервиса не понадобится изменять другие – они работают независимо. Микросервисы связаны друг с другом через API – специ-

альный интерфейс, который помогает модулям «общаться» без вмешательства в их внутреннюю работу;

– инфраструктура как код, или IaC. Вместо того чтобы вручную настраивать среду, DevOps-инженер создает конфигурационные файлы, которые в нужный момент запускаются из консоли – достаточно пары строчек с командами. Среда и окружение настраиваются автоматически согласно этим файлам.

Подход DevOps активно используется при реализации ML-проектов, соответствующие инженеры получили отдельное название – MLOps-инженеры. Помимо задач, общих с DevOps-инженерами, они решают аналогичные задачи применительно к обучаемой ML-модели: они автоматизируют пайплайны подготовки датасетов, собственно обучения и его мониторинга, а также развертывания готовой модели.

Артефактами DevOps и MLOps являются:

– настроенные пайплайны, т.е. конвейеры обработки данных для решения всех вышеперечисленных задач, в форме документированного программного кода;
– разработанные и настроенные дашборды, по которым можно отслеживать ход выполнения всех вышеперечисленных операций;
– результат развертывания модели (или измененного компонента, или микросервиса, и т.д.) в адресной среде реализации.

2.2. Инструментарий науки о данных

2.2.1. Инструментарий общего назначения

Область науки о данных стремительно развивается, и появилось множество инструментов, помогающих специалистам по данным в их работе. Инструменты науки о данных необходимы для того, чтобы помочь ученым и аналитикам данных извлекать ценную информацию из данных. Эти инструменты полезны для очистки, обработки, визуализации и моделирования данных.

С появлением больших языковых моделей (LLM) все больше инструментов интегрируются с ними, что еще больше упрощает для ученых анализ данных и построение моделей. Например, возможности генеративного ИИ (PandasAI) перешли в более простые инструменты, такие как pandas, позволяя пользователям получать результаты, записывая подсказки на естественном языке. Однако эти новые инструменты пока не получили широкого распространения среди специалистов по данным.

Более того, инструменты науки о данных не ограничиваются выполнением только одной функции. Например, MLFlow в основном используется для отслеживания (tracking) моделей. Однако его также можно использовать для регистрации, развертывания и инференса моделей.

Обзор [Awan] выделяет топ-10 инструментов науки о данных, наиболее востребованных в 2024 году:

Инструменты на основе Python для науки о данных

1. Pandas – библиотека, наиболее часто используемая профессионалами в области обработки данных для всех видов задач.
2. Seaborn – мощная библиотека визуализации данных, созданная на основе Matplotlib.
3. Scikit-learn – популярная библиотека Python для машинного обучения.

Инструменты для анализа данных с открытым исходным кодом

4. Jupyter Notebooks – популярное веб-приложение с открытым исходным кодом, позволяющее специалистам по обработке данных создавать общедоступные документы, объединяющие живой код, визуализации, уравнения и текстовые пояснения.
5. Pytorch – очень гибкая среда машинного обучения с открытым исходным кодом, которая широко используется для разработки моделей нейронных сетей.
6. MLFlow – платформа с открытым исходным кодом от Databricks для управления сквозным жизненным циклом машинного обучения.
7. Hugging Face – универсальное решение для разработки машинного обучения с открытым исходным кодом, образующее свою экосистему.

Проприетарные инструменты для анализа данных

8. Tableau – лидер в области программного обеспечения для бизнес-аналитики. Он обеспечивает интуитивную интерактивную визуализацию данных и дашборды.
9. RapidMiner – это комплексная платформа расширенной аналитики для создания машинного обучения и конвейеров данных, которая предлагает визуальный конструктор рабочих процессов для бесшовной сборки процесса.

Инструменты искусственного интеллекта

10. ChatGPT – это инструмент на базе искусственного интеллекта, который поможет вам в решении различных задач по обработке и анализу данных.

Аналогичный список от [Pratt] содержит 18 инструментов:

1. Apache Spark – система обработки и анализа данных с открытым исходным кодом, которая может обрабатывать большие объемы данных – по словам сторонников, свыше нескольких петабайт.
2. D3.js – библиотека JavaScript для создания пользовательских визуализаций данных в веб-браузере.
3. IBM SPSS – семейство программного обеспечения для управления и анализа сложных статистических данных. Оно включает два основных продукта: SPSS Statistics, инструмент статистического анализа, визуализации данных и отчетности, и SPSS Modeler, платформа науки о данных и предиктивной аналитики с пользовательским интерфейсом с функцией перетаскивания и возможностями машинного обучения.
4. Julia – язык программирования с открытым исходным кодом, используемый для числовых вычислений, а также для машинного обучения и других видов приложений в области науки о данных.
5. Jupyter Notebook – веб-приложение с открытым исходным кодом, обеспечивающее интерактивное сотрудничество между специалистами по обработке

данных, инженерами по обработке данных, математиками, исследователями и другими пользователями.

6. Keras – программный интерфейс, который позволяет специалистам по данным получать доступ к платформе машинного обучения TensorFlow и использовать ее более легко. Это API и фреймворк глубокого обучения с открытым исходным кодом, написанные на Python, которые работают поверх TensorFlow.
7. Matlab – язык программирования высокого уровня и аналитическая среда для численных вычислений, математического моделирования и визуализации данных.
8. Matplotlib – библиотека Python с открытым исходным кодом для построения графиков, которая используется для чтения, импорта и визуализации данных в аналитических приложениях.
9. NumPy (Numerical Python, численный Python) – библиотека Python с открытым исходным кодом, которая широко используется в научных вычислениях, инженерии, а также в приложениях для обработки и анализа данных и машинного обучения.
10. Pandas – другая популярная библиотека Python с открытым исходным кодом, обычно используется для анализа и обработки данных. Созданная на основе NumPy, она включает две основные структуры данных: одномерный массив Series и DataFrame, двумерную структуру для обработки данных с интегрированной индексацией.
11. Python – наиболее широко используемый язык программирования для науки о данных и машинного обучения и один из самых популярных языков в целом.
12. PyTorch – фреймворк с открытым исходным кодом, используемый для создания и обучения моделей глубокого обучения на основе нейронных сетей. Его сторонники хвалят его за поддержку быстрого и гибкого экспериментирования, а также за плавный переход к развертыванию в производственной среде.
13. Язык программирования R – среда с открытым исходным кодом, предназначенная для статистических вычислений и графических приложений, а также для обработки, анализа и визуализации данных.
14. SAS – интегрированный программный пакет для статистического анализа, расширенной аналитики, бизнес-аналитики и управления данными.
15. Scikit-learn – библиотека машинного обучения с открытым исходным кодом для Python, созданная на основе библиотек научных вычислений SciPy и NumPy, а также Matplotlib для построения графиков данных.
16. SciPy (Scientific Python) – еще одна библиотека Python с открытым исходным кодом, которая поддерживает научные вычисления.
17. TensorFlow – платформа машинного обучения с открытым исходным кодом, разработанная Google, которая особенно популярна для реализации нейронных сетей глубокого обучения.
18. Weka – платформа с открытым исходным кодом, предоставляющая набор алгоритмов машинного обучения для использования в задачах добычи дан-

ных. Алгоритмы Weka, называемые классификаторами, можно применять непосредственно к наборам данных без программирования через графический интерфейс или интерфейс командной строки, который предлагает дополнительные функции. Их также можно реализовать через Java API.

Как легко заметить, эти списки во многом схожи. Среди них встречаются как профессиональные инструменты, так и платформы для обучающихся и желающих «пощупать» DS без программирования (например, Tableau и RapidMiner). Однако, как показывает практика, в среде профессиональных специалистов по DS и ML в основном используются языки программирования – Python / R / C++ / JS (по убыванию частоты встречаемости).

2.2.2. Инструментарий визуализации данных низкой размерности

Визуализация данных [Яу, Маккэндлесс] представляет собой самостоятельную задачу, специфичную для DS. Визуализация данных позволяет акторам DS-проекта интегрально оценить структуру данных и тем самым помогает выделить скрытые связи, построить инсайты и гипотезы. Поэтому выбор инструментария визуализации может существенно повлиять на результат DS-проекта.

Рассмотрим вначале наиболее распространенные инструменты для визуализации данных низкой размерности.

Таблицы и матрицы – стандартные способы представления данных в виде сетки из строк и столбцов. Полезны для отображения большого количества данных и сравнения значений между различными категориями или временными периодами.

Сети и графы – способ визуализации отношений между объектами и их взаимодействиями. Они могут быть представлены в виде узлов, соединенных линиями, которые отражают связи между объектами.

Графики и диаграммы – наиболее распространенные способы визуализации данных. Они позволяют наглядно отображать информацию и выявлять закономерности и зависимости. К ним относятся:

- столбчатые диаграммы и гистограммы;
- круговые диаграммы;
- линейные графики;
- точечные диаграммы;
- диаграммы «ящик–усы»;
- сводные графики.

Тепловые карты – двумерные изображения, в которых каждая ячейка окрашена в определенный цвет в зависимости от значения соответствующего ей параметра. Они полезны для отображения распределения значений по двум измерениям и определения областей с наибольшей и наименьшей концентрацией значений.

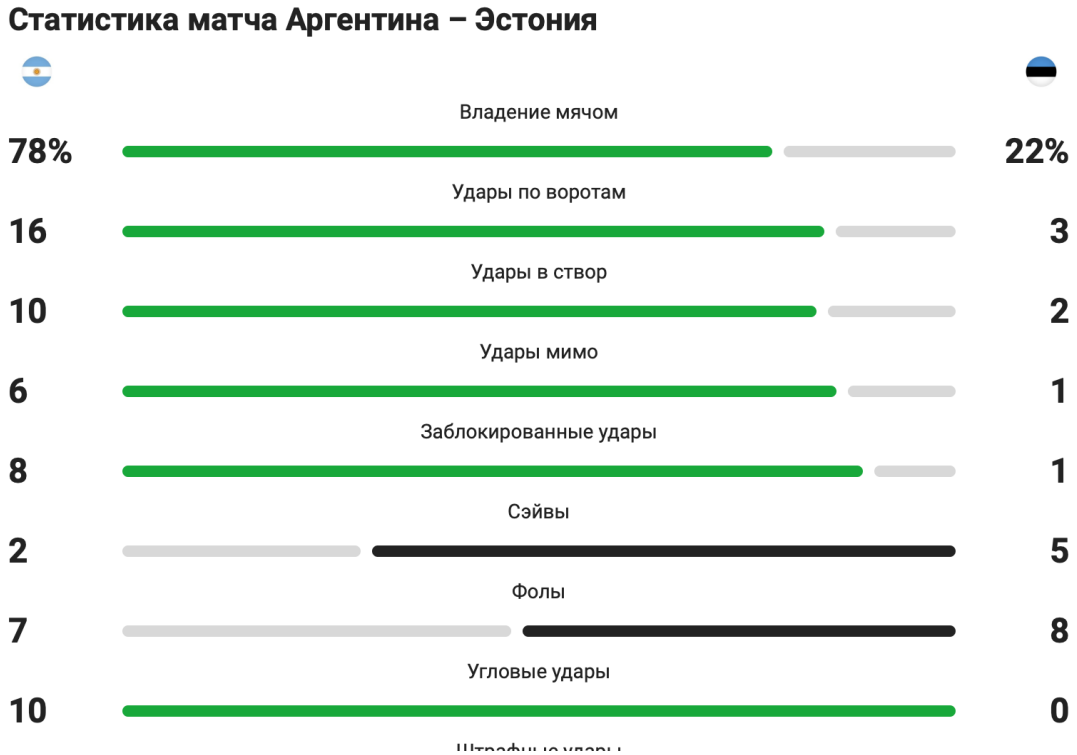
Географические карты позволяют отображать данные, связанные с определенными географическими областями. Они полезны для анализа распределе-

ния значений по географическим координатам и определения областей с наибольшей и наименьшей концентрацией значений.

Деревья и иерархии используются для визуализации отношений между объектами и группирования их по определенным категориям. Могут быть представлены в виде деревьев, радиальных графиков или диаграммы с узлами и связями.

Инфографика – это графическое представление информации, данных или знаний, которое позволяет легко и быстро воспринимать информацию. Она может включать в себя различные виды диаграмм, графиков, иконок и текста, которые складывается в определенную историю, инструкцию. Примеры удачной инфографики представлены на рис. 2.2, а, б.

а



б

Рис. 2.2. Примеры удачной инфографики

2.2.2. Инструментарий визуализации данных высокой размерности

В условиях Big Data в задачах DS часто требуется обрабатывать датасеты высокой и сверхвысокой размерности. Инструменты для визуализации таких данных должны снизить размерность данных до обозримой человеком (2–3) и при этом сохранить и продемонстрировать основные взаимосвязи в датасете.

Сегодня дата-сайентист имеет в своем арсенале три базовых инструмента, основанных на алгоритмах снижения размерности. Самым старым (предложен в 1901 г.) и базовым является линейный алгоритм снижения размерности, основанный на методе главных компонент (PCA); за ним последовали получившие широкое распространение нелинейные алгоритмы t-SNE (2008 г.) и UMAP (2020 г.).

Метод главных компонент (Principal Component Analysis, PCA) [Loginom Skills] – ортогональное линейное преобразование, которое отображает данные из исходного пространство признаков в новое пространство меньшей размерности.

При этом первая ось новой системы координат строится таким образом, чтобы дисперсия данных вдоль нее была бы максимальна. Первая главная компонента PC1 (ось Z на рис. 2.3) ориентирована вдоль направления наибольшей вытянутости эллипсоида рассеяния точек объектов исходного набора данных в пространстве признаков, т.е. с ней связана наибольшая дисперсия. Вторая ось (ось W на рис. 2.3) строится ортогонально первой так, чтобы дисперсия данных вдоль нее была бы максимальной их оставшихся возможных, и т.д. Первая ось называется первой главной компонентой, вторая – второй и т.д. Таким образом, для исходной многомерной выборки данных строятся новые координатные оси.

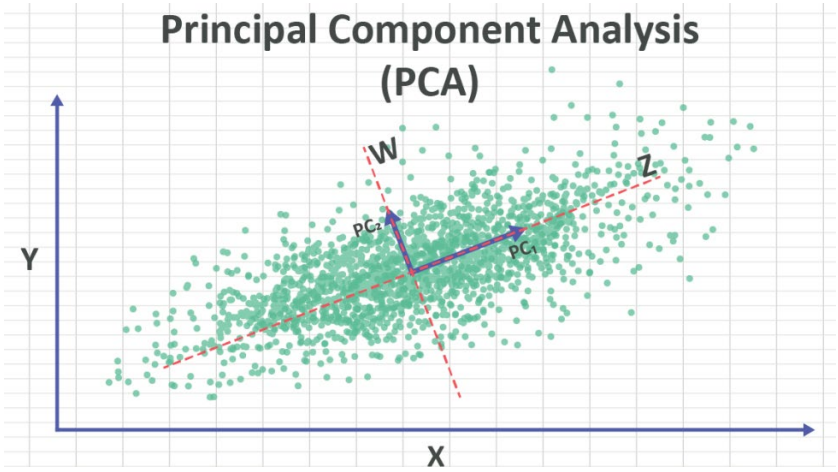


Рис. 2.3. Иллюстрация метода главных компонент
(источник [https://numxl.com/blogs/principal-component-analysis-pca-101/])

Свойства PCA как средства визуализации [NEERC]:

- для визуализации многомерных данных используются только две первых оси, PC1 и PC2, а остальные оси отбрасываются;
- проекция на главные компоненты является наименее искаженной из всех линейных проекций многомерной выборки на какую-либо пару осей: координаты имеют физический смысл, их можно интерпретировать, расстояния имеют физический смысл;

– оси главных компонент отражают «основные тенденции» в данных: как правило, в них удаётся увидеть наиболее существенные особенности исходных данных, даже несмотря на неизбежные искажения. В частности, можно судить о наличии кластерных структур и выбросов.

Алгоритм t-SNE (t-distributed stochastic neighbor embedding) [Maaten] представляет собою вариант техники нелинейного снижения размерности и визуализации многомерных переменных. Алгоритм решает следующую задачу: имеется набор данных с точками, описываемыми многомерной переменной с размерностью пространства существенно больше трех; необходимо получить новую переменную, существующую в двумерном или трехмерном пространстве, которая бы в максимальной степени сохраняла структуру и закономерности в исходных данных. Алгоритм основан на вычислении условных вероятностей того, насколько близки друг к другу пары точек X_j и X_i в исходном пространстве высокой размерности и их двумерные или трехмерные «коллеги» Y_i и Y_j. Если точки отображения Y_i и Y_j корректно моделируют сходство между исходными точками высокой размерности X_i и X_j, то соответствующие условные вероятности $p_{j|i}$ и $q_{j|i}$ будут эквивалентны.

Алгоритм минимизирует сумму расстояний между вероятностными распределениями, выраженных через дивергенцию Кульбака-Лейблера KL(Pi || Qi) для всех точек отображения, при помощи градиентного спуска. Функция потерь для данного метода будет определяться формулой

$$Cost = \sum_i KL(P_i \parallel Q_i) = \sum_i \sum_j p_{j|i} log \frac{p_{j|i}}{q_{j|i}}$$

Для улучшения разделимости точек в моделирующем пространстве распределение точек в исходном (многомерном) пространстве считается гауссовым, а в пространстве отображения используется t-распределение (Стьюдента), «тяжелые» хвосты которого облегчают оптимизацию.

Свойства t-SNE как средства визуализации [Хмельков]:

– в отличие от PCA, t-SNE – это стохастический алгоритм, поэтому он может продуцировать разные результаты в зависимости от выбора гиперпараметров,
– функция потерь t-SNE оптимизирует локальные расстояния, а не глобальные (из-за особенности t-распределения),
– в отличие от PCA, в t-SNE взаимное расположение кластеров друг относительно друга не имеет физического смысла.

Алгоритм UMAP (Uniform Manifold Approximation and Projection) [McInnes] – также нелинейный алгоритм снижения размерности. UMAP концептуально очень похож на t-SNE, но использует не поточечное, а «попереберное» сравнение графов, сформированных в пространствах высокой и низкой размерности, соответственно.

Алгоритм основан на топологических структурах в пространстве высокой размерности. Здесь UMAP выполняет построение взвешенного графа, соединяя ребрами только те объекты, которые являются ближайшими соседями. Множество ребер графа – это нечёткое множество с функцией принадлежности, кото-

рая определяется как вероятность существования ребра между двумя вершинами. Затем алгоритм создает граф в низкоразмерном пространстве и приближает его к исходному, минимизируя сумму дивергенций Кульбака–Лейблера для каждого ребра из множеств. Минимизация, аналогично t-SNE, выполняется с помощью стохастического градиентного спуска. Полученное множество из ребер определяет новое расположение объектов и, соответственно, низкоразмерное отображение исходного множества данных.

Свойства UMAP как средства визуализации [Bex]:

– UMAP имеет такую же высокую скорость работы, как PCA;
– UMAP может находить структуры в шуме;
– UMAP хорошо описывает локальные структуры, при этом сохраняет глобальную структуру данных лучше t-SNE [Becht].

Сравнение PCA, t-SNE и UMAP проиллюстрируем на примере их применения для визуализации датасета MNIST [Кушнарева]. Датасет MNIST состоит из черно-белых изображений рукописных цифр от 0 до 9 размерами 28×28 пикселей, т.е. размерность одного сэмпла датасета составляет 28×28=784. Хочется посмотреть, насколько классы этих изображений отличаются друг от друга, т.е. выполнить кластеризацию и отобразить ее на плоскость. Результат кластеризации представлен на рис. 2.4. Выделим его особенности:

– все алгоритмы сформировали кластеры из точек, соответствующих примерам с одинаковыми метками (обозначены точками одинакового цвета), но у разных алгоритмов эти кластеры выглядят по-разному;
– расстояние между кластерами в алгоритме PCA можно содержательно интерпретировать. Например, на графике единица (красный кластер) находится дальше от нуля (бордовый кластер), чем четверка (желтый кластер). И действительно, по написанию единица больше похожа на четверку, чем на ноль. В то же время в алгоритмах t-SNE и UMAP эти расстояния ничего не значат.

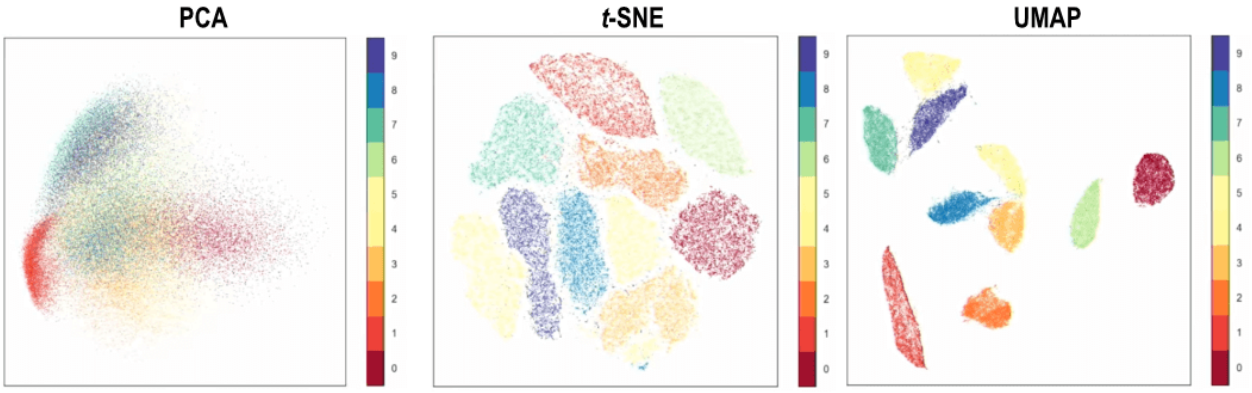


Рис. 2.4. Результат применения алгоритмов; цвет точки означает метку соответствующего примера (источник [https://meta.caspershire.net/umap/])

– при многократных запусках алгоритма PCA мы будем получать практически идентичную картину расположения кластеров, в то время как для t-SNE и UMAP рассчитывать на надёжную воспроизводимость результатов нельзя. Особенно невоспроизводимые результаты получаются на t-SNE (рис. 2.5).

Рис. 2.5. Результаты разных запусков t-SNE (источник: [Wang Y.])

– все три алгоритма имеют хорошее теоретическое обоснование и надежную документацию по имплементации:
  – Документация по PCA в sklearn: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html
  – Документация пакета UMAP: https://umap-learn.readthedocs.io/en/latest/index.html
  – Документация по t-SNE в sklearn: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

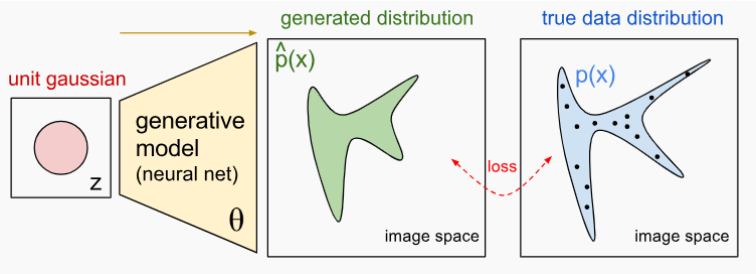
Прогресс не стоит на месте, и постоянно предлагаются новые алгоритмы для визуализации высокоразмерных данных. Примером может служить алгоритм Deep TDA, использующий топологический подход к описанию датасета. Содержательное описание алгоритма и ссылки на более подробные источники можно найти в [Kibardin, 2020].

Алгоритм глубокого топологического анализа (Deep Topological Analysis, DTA) представляет собой комбинацию топологического анализа данных (Topological Data Analysis, TDA) [Zia] и глубоких генеративных моделей. Алгоритм базируется на последовательности шагов, типичной для TDA:

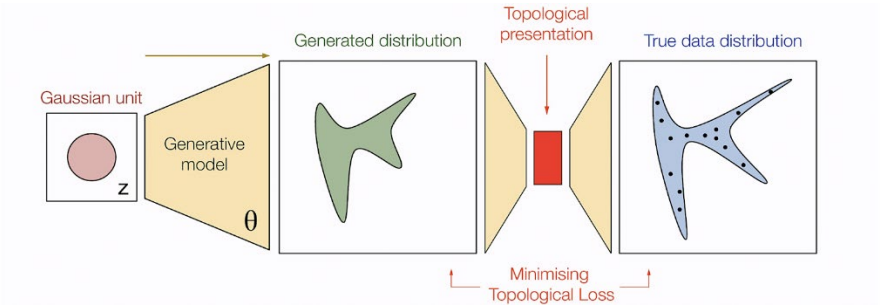
– на исходном облаке точек высокой размерности вычисляются вложенные топологические комплексы (используется алгоритм Виеториса–Рипса). При этом близлежащие точки соединяются в топологические структуры;
– для этих комплексов вычисляются баркоды, которые позволяют идентифицировать устойчивые элементы структуры, и строится граф Риба, который представляет собой дву- и трехмерное представление исходного датасета.

Однако построение графа Риба очень затратно в вычислительном плане, особенно для больших объемов данных с большой размерностью. Поэтому в DTA этот шаг реализуется посредством генеративной модели, в качестве которой используется автоэнкодер [Karpathy] (рис. 2.6, а).

Таким образом, алгоритм DTA [Kibardin, 2020] представляет собою вариант автоэнкодера, в котором в качестве функции потерь используются дифференцируемые топологические функции исходного датасета (рис. 2.6, б). Эмбеддинги, формируемые автоэнкодером на старших слоях, характеризуют кластерную структуру датасета и могут быть визуализированы.



а



б

Рис. 2.6. Схема формирования алгоритма TDA:
а – исходный автоэнкодер [Karpathy]; б – алгоритм TDA [Kibardin, 2020]

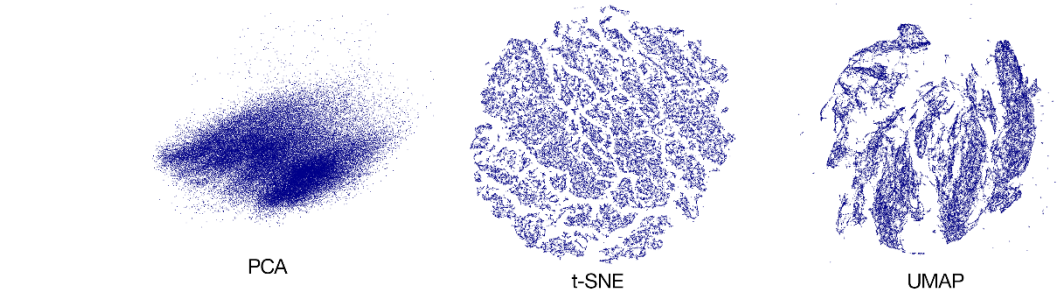
Авторы алгоритма выделяют следующие преимущества DTA по сравнению с t-SNE или UMAP:

- *Робастность (Robustness).* DTA более устойчив к шуму и выбросам в данных по сравнению с t-SNE и UMAP. Это связано с тем, что DTA основан на топологии данных, которая более устойчива к шуму, чем геометрические методы, такие как t-SNE и UMAP;
- *Многомасштабный анализ (Multiscale Analysis).* DTA может обнаруживать и представлять структуру данных в нескольких масштабах, что позволяет более полно и детально понимать данные. Напротив, t-SNE и UMAP в основном предназначены для захвата локальных структур в данных и могут пропускать глобальные структуры;
- *Способность изучать сложные закономерности (Ability to learn complex patterns).* Глубокое обучение без учителя (self-supervised deep learning), реализуемое генеративной моделью, позволяет DTA изучать сложные иерархические представления данных, что может быть особенно полезно для многомерных данных с нелинейными отношениями, которые может быть трудно охватить такими методами, как t-SNE и UMAP;
- *Отсутствие параметров (parameter-free).* DTA относительно свободен от параметров, т.е. он не требует большой настройки или предварительного знания данных. Напротив, t-SNE и UMAP требуют тщательного выбора параметров для достижения хороших результатов;
- *Масштабируемость (Scalability).* DTA масштабируется до больших наборов данных, и последние достижения в вычислительных методах сделали DTA более эффективным с вычислительной точки зрения. Это делает DTA подходящим для анализа сложных и многомерных наборов данных, которые часто встречаются в реальных приложениях.

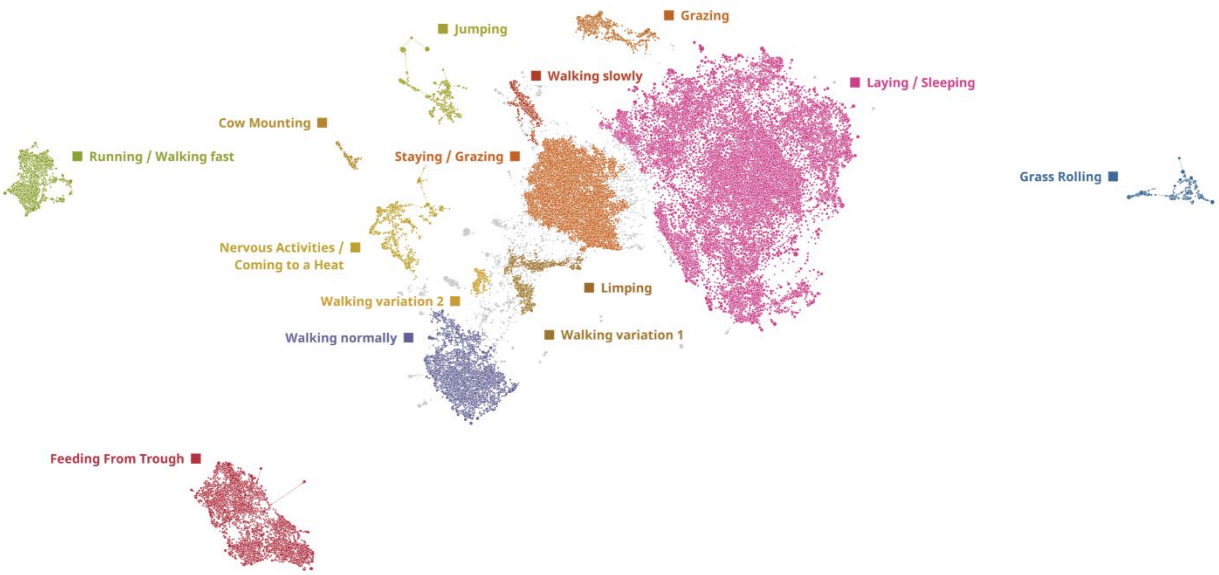
Если первые три свойства представляются достаточно очевидными, то последние два свойства вызывают сомнения и требуют экспериментальной проверки, которую авторы не предоставляют. Кроме того, следует ожидать еще большей невоспроизводимости результатов кластеризации при разных запусках алгоритма, чем при использовании t-SNE и UMAP. Как ни странно, именно за счет невоспроизводимости алгоритм представляется особенно перспективным для формирования догадок (инсайтов) на первых этапах DS-процесса.

Ниже приведены примеры [Kibardin, 2023] использования алгоритма DTA для визуализации высокоразмерных данных различной природы (временные ряды, тексты, изображения) по сравнению с другими алгоритмами, решающими ту же задачу.

Кластеризация временных рядов (рис. 2.7). Использован датасет, содержащий информацию об активности коров в стаде. На ошейнике коровы крепятся датчики, генерирующие 9 потоков данных: положение (данные компаса в трех направлениях), скорость (гироскоп в трех направлениях), ускорение (акселерометр в трех направлениях). Общий период наблюдения составил около 2000 минут для каждой коровы, всего в стаде 43 коровы. Каждая точка представляет собой окно 10-минутной активности коровы (скользящее по минутам). Отдельные кластеры представляют собой уникальные типы активности.



а



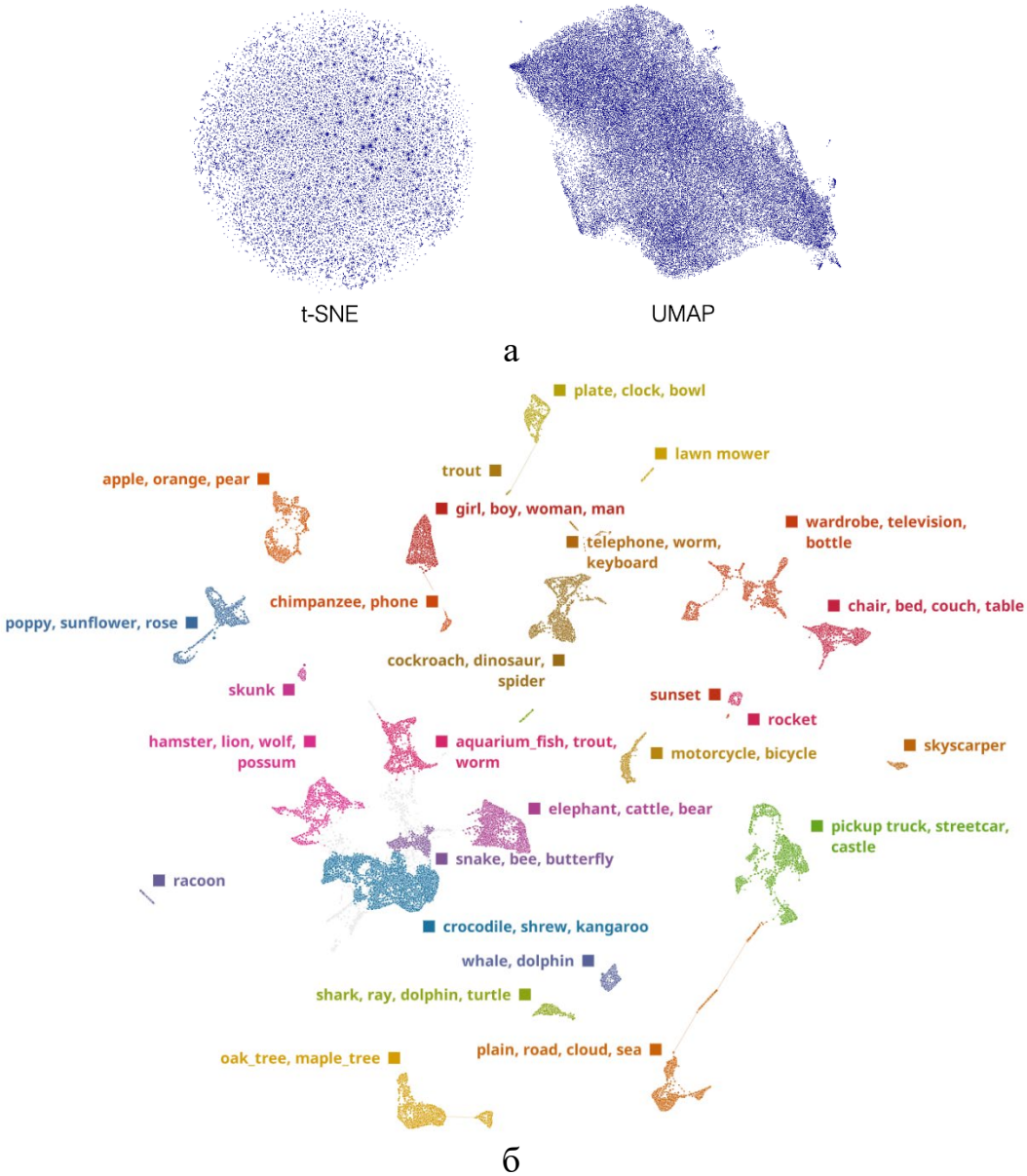
б

Рис. 2.7. Кластеризация временных рядов: а – традиционные алгоритмы, б – DTA

Как видно из 2.7, а, PCA вообще не выявил кластерной структуры в данных; UMAP и t-SNE сформировали некоторую структуру кластеров, но ее ока-

залось невозможно содержательно интерпретировать. В то же время DTA выявил структуру кластеров, которые соответствуют реальной активности животных: большую часть времени коровы проводят лежа (laying/sleeping) и питаясь (grazing/feeding), но есть и другие виды деятельности, такие как прыжки (jumping) или катание по траве (grass rolling). Эти результаты дают возможность составить паттерны поведения для каждой коровы, что важно для фермера с точки зрения формирования стада, ухода за ним и пр.

Кластеризация изображений (рис. 2.8). Использован популярный датасет CIFAR100. Он состоит из 60 000 цветных изображений размером 32×32×3 пикселя (в системе RGB) в 100 классах, по 600 изображений на класс. Общий размер обрабатываемой матрицы составляет [60000, 3072]. Классы в CIFAR100 более детализированы, чем в CIFAR10, включая такие подкатегории, как водные млекопитающие, цветы, насекомые и домашняя мебель.



а

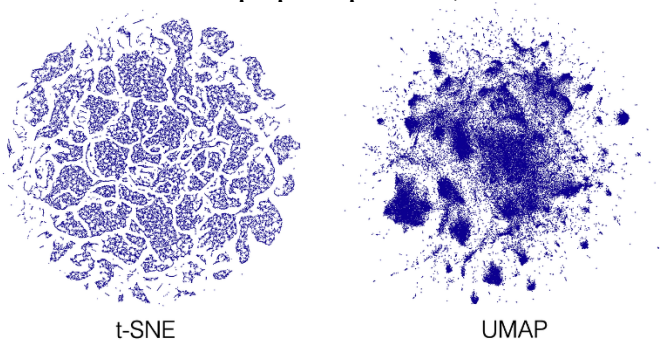
б

Рис. 2.8. Кластеризация изображений: а – традиционные алгоритмы, б – DTA

Как видно из рис. 2.8, в связи с высокой сложностью данных UMAP и t-SNE не сформировали структуру кластеров. В то же время DTA выявил структуру кластеров (кластеры названы по наиболее доминирующим категориям в столбце меток), которые хорошо соответствуют семантическому группированию изображений, производимому человеком.

Кластеризация текстов (рис. 2.9). Использованы 100 000 записей из датасета Amazon Fine Food reviews[1].

Здесь традиционные алгоритмы выделяют некоторую структуру, но ее весьма трудно содержательно интерпретировать, чего нельзя сказать о DTA.



а

б

Рис. 2.9. Кластеризация текстов: а – традиционные алгоритмы, б – DTA

[1] https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews

Существенным достоинством DTA является не только хорошая семантическая интерпретируемость формируемых кластеров, но и то обстоятельство, что расстояния между ними имеют физический смысл: чем дальше кластеры друг от друга, тем сильнее они содержательно различны. В этом аспекте алгоритм DTA, будучи существенно нелинейным, повторяет линейный алгоритм PCA. Однако, как отмечают сами разработчики [Kibardin, 2023], в настоящее время DTA недоступен в библиотеках с открытым исходным кодом. Кроме того, выполнение анализа изображений или временных рядов требует соответствующей обработки данных, что может быть сложной задачей и при неправильном исполнении может исказить результаты анализа.

Вопросы для самопроверки

1. Что такое Артефакты процесса DS, перечислите их?
2. Как образуются инсайты и гипотезы в DS?
3. Приведите пример использования OLAP-куба?
4. Какие практики используются в подходе DevOps?
5. Чем занимаются DevOps- и MLOps-инженеры в реализации DS проекта?
6. Перечислите инструменты науки о данных, наиболее востребованные на сегодняшний день?
7. Какие существуют инструменты для визуализации данных низкой размерности?
8. Перечислите отличительные свойства основных инструментов, основанных на алгоритмах снижения размерности?
9. Какие существуют основные преимущества DTA по сравнению с t-SNE или UMAP?
10. Приведите примеры использования алгоритма DTA для визуализации высокоразмерных данных различной природы?
11. Перечислите свойства t-SNE и PCA как средств визуализации?

3. СТАТИСТИЧЕСКИЕ И ВЕРОЯТНОСТНЫЕ МЕТОДЫ В НАУКЕ О ДАННЫХ

3.1. Статистика vs теория вероятностей в DS

3.1.1. Соотношение статистики и теории вероятностей

Данные, которые являются предметом изучения в DS, характеризуют не один объект или единичное явление, а являются результатом оценки параметров повторяющихся, вплоть до массовых, явлений. Соответственно, теоретический фундамент для DS составляют теория вероятностей и статистика. Обе науки с математических позиций изучают массовые явления, однако делают это принципиально по-разному [Пономаренко]:

- статистика индуктивна, т.е. строит свои выводы от частного к общему. В статистике суждения о свойствах явления в целом делаются на основе анализа параметров отдельных единиц, входящих в генеральную совокупность. Научный метод называется статистическим, если он связывает эмпирические факты, определённого рода гипотезы и методы их проверки.
- теория вероятностей дедуктивна, т.е. строит свои выводы от общего к частному. Она основана на системе аксиом, которым должен удовлетворять базовый артефакт – функция распределения вероятностей. Упрощая, можно сказать так: теория вероятностей изучает все закономерности, которые можно получить из функции распределения; если для некоторой совокупности нельзя построить функцию распределения, то она не может быть исследована методами теорвера.

Рассмотрим два примера [Гатман].

Работа любого казино основана на использовании теории вероятностей. Когда игрок делает ставку, кладет фишку на стол или дергает за рычаг игрового автомата, казино точно знает вероятность его выигрыша. Владельцы казино оптимизируют соотношение выигрышей и проигрышей, чтобы поддерживать в игроках определенный уровень интереса и возбуждения, но теорвер позволяет гарантировать, что в долгосрочной перспективе казино окажется в выигрыше.

В основе опросов общественного мнения, будь то опросы покупателей, избирателей и пр., лежит статистика. Организаторы опросов не знают, какова функция распределения мнений избирателей (а может ли она вообще быть построена с необходимой точностью?), но они пытаются, анализируя результаты опросов, выявить некоторые закономерности поведения респондентов. Если на этом основании удается скорректировать рекламные, предвыборные и пр. кампании в лучшую сторону, то применение статистических методов считается эффективным (хотя тут никто ничего не гарантирует).

3.1.2. Статистические совокупности и их репрезентативность

Статистические совокупности являются объектом изучения и для теории вероятностей, и для статистики.

Статистическая совокупность [Кузнецов] – множество единиц изучаемого явления, имеющих единую качественную основу, но различающихся отдель-

ными признаками. Другими словами, это группа, состоящая из большого числа относительно похожих единиц (жителей какой-то территории, покупателей какого-то продукта и пр.).

Чтобы совокупность объектов была статистической, она должна обладать следующими свойствами.

- Неразложимость: при появлении или изъятии элементов качественная основа статистической совокупности не разрушается. Например, характеристика студентов не меняется, хотя каждый год в их ряды вступают первокурсники, а выпускники их покидают.
- Однородность: всегда есть как минимум один общий признак для всех элементов. Этот признак может иметь разные значения для разных единиц. Однородность статистической совокупности устанавливается во время исследования и зависит от поставленных целей и задач.

Статистическая совокупность может рассматриваться как генеральная или как выборочная:

- генеральная совокупность – это совокупность всех без исключения единиц изучаемого объекта, всех единиц, которые соответствуют цели исследования.
- выборочная совокупность (выборка) – это часть единиц генеральной совокупности, отобранная специальным методом и предназначенная для характеристики всей генеральной совокупности.

Возможность изучать свойства совокупности по выборке опирается на закон больших чисел [Пасхавер], согласно которому среднее значение конечной выборки из фиксированного распределения близко к математическому ожиданию этого распределения. Другими словами, чем больше объем выборки (чем больше в ней единиц наблюдения), тем больше можно доверять выводам исследования.

Однако просто увеличивать объем наблюдений недостаточно – сама выборка должна обладать свойством репрезентативности.

Репрезентативность статистической совокупности [Орешков] показывает, содержат ли анализируемые данные достаточно информации для построения модели, т.е. отражает способность данных представлять зависимости и закономерности исследуемой предметной области, которые должна обнаружить и научиться воспроизводить построенная модель.

Репрезентативность генеральной совокупности отражает ее способность совокупности описывать существенные свойства, зависимости и закономерности объектов, процессов и явлений предметной области. Она достигается за счет правильной организации сбора и консолидации первичных данных.

Репрезентативность выборки описывает способность выборочных данных отражать структурные свойства совокупности, из которой они были извлечены, т.е. дает ответ на вопрос: можно ли в исследовании заменить совокупность на выборку без значимого ухудшения результатов анализа. Репрезентативность выборки достигается с помощью правильного выбора метода сэмплинга. Например, интернет-опрос может показать, что 100% людей пользуется интернетом, хотя это не соответствует действительности (т.е. репрезентативность нарушена).

Таким образом, репрезентативность выборки касается только воспроизведения характеристик совокупности. Если сама исходная совокупность плохо представляет предметную область, то, даже если полученная из нее выборка будет репрезентативной, построить на ее основе корректную с точки зрения предметной области модель невозможно.

Репрезентативность выборки разделяется на качественную (структурную) и количественную (рис. 3.1).

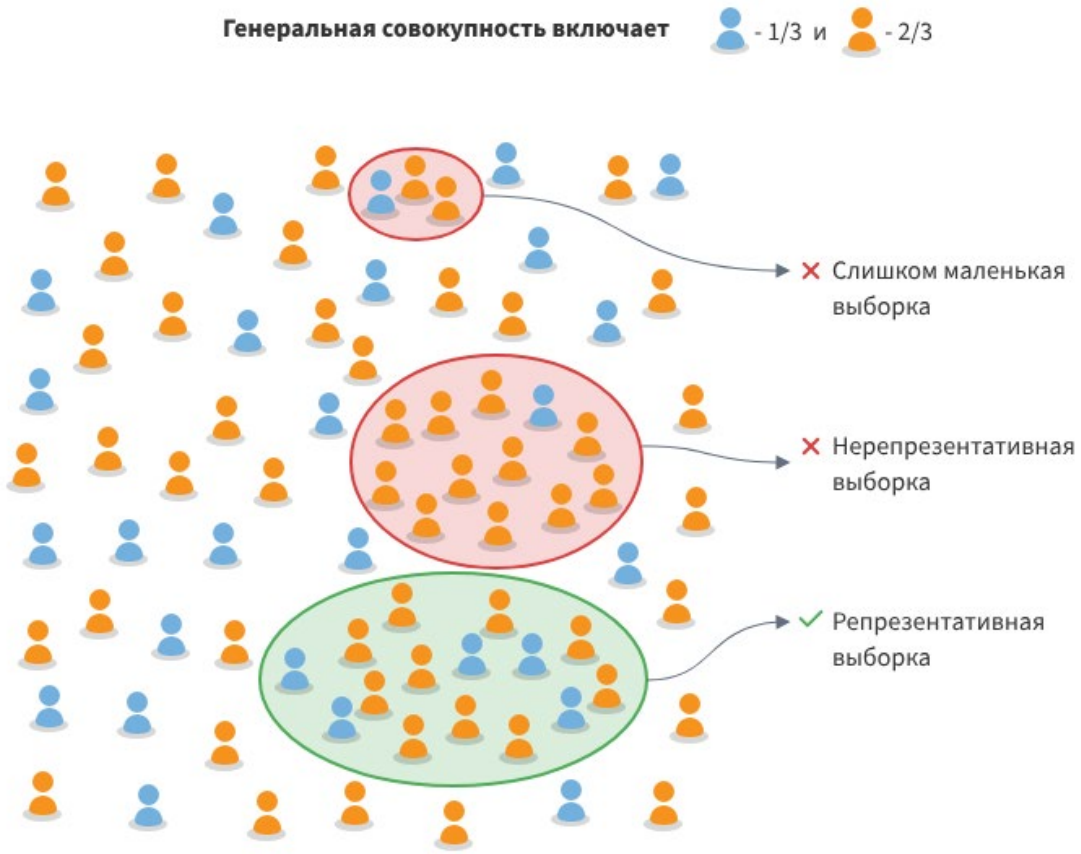


Рис. 3.1. Качественная и количественная репрезентативность выборки

– Качественная репрезентативность показывает, что все группы, присутствующие в совокупности, будут представлены и в выборке. Для этого каждый элемент совокупности должен иметь равную вероятность быть выбранным, а сама выборка должна производиться из однородных групп.
– Количественная репрезентативность показывает, является ли достаточным число элементов выборки для представления характеристик генеральной совокупности с заданной погрешностью. В целом число наблюдений, требуемое для получения репрезентативной выборки, изменяется обратно пропорционально квадрату допустимой ошибки; более точные расчеты требуют учета контекста.

Как оценить репрезентативность выборки? Самыми старыми являются статистические методы оценки репрезентативности выборки. Формируют две независимые выборки, вычисляют и сравнивают их статистические параметры (среднее значение, дисперсия, среднеквадратичное отклонение и т.д.) , и если они совпадают (различаются незначимо – например, не более чем на 5%), то выборки считаются репрезентативными.

Для сравнения двух выборок предложен целый ряд критериев [Энциклопедия], в том числе:

- критерий Уилкоксона (Wilcoxon signed-rank test);
- U-критерий Манна-Уитни (англ. Mann–Whitney U-test);
- t-критерий Стьюдента;
- критерий однородности Смирнова;
- Q-критерий Розенбаума и др.

Она различаются по сравниваемым параметрам и по организации процедуры сравнения. Например, U-критерий Манна-Уитни определяет, достаточно ли мала зона перекрещивающихся значений между двумя рядами (ранжированным рядом значений параметра в первой выборке и таким же во второй выборке). Чем меньше значение критерия, тем вероятнее, что различия между значениями параметра в выборках достоверны. Критерий однородности Смирнова используется для проверки гипотезы о принадлежности двух независимых выборок одному закону распределения, то есть о том, что два эмпирических распределения соответствуют одному и тому же закону. Критерий Стьюдента проверяет равенство средних значений в двух выборках.

Статистические методы оценки репрезентативности выборки, хотя и строго обоснованы, но сложны в использовании и имеют ряд формальных ограничений на характеристики выборок. Поэтому их имеет смысл применять только тогда, когда весь последующий анализ делается статистическими средствами.

Можно идти другим путем: не оценивать репрезентативность уже имеющейся выборки, а сразу формировать ее как репрезентативную. Для этого существует ряд методов:

- Механический отбор. Единицы наблюдения отбираются по какому-либо условному признаку, ничего не значащему для цели исследования и никак не влияющему на результат – например, все пациенты на букву М; каждая пятая покупка.
- Серийный (гнездный) отбор. При этом единицы наблюдения отбираются не по одной, а группами (сериями, гнездами). Например: если на территории района тридцать примерно одинаковых населенных пунктов, то для изучения отберем населенные пункты А, В и С, но внутри каждой группы изучаем всех жителей.
- Типологический (типический) отбор. Генеральная совокупность предварительно разбивается на типы (по полу, возрасту, уровню образования и пр.) с последующим отбором единиц наблюдения из каждой типологической группы пропорционально численности этих групп – например, пропорционально соотношению мужчин и женщин.
- Парно-сопряженный отбор (метод «копи-пара»). Единицы наблюдения подбираются парами: одна в основной группе, одна в контрольной группе. Эти парные единицы являются копиями друг для друга, т.е. имеют одинаковые значения самых важных для данного исследования факторных признаков. Исследуемое воздействие применяется только к основной группе.

Хотя эти методы содержательно вполне разумны, строгое доказательство репрезентативности выборок и, значит, исследования в целом в этих случаях невозможно.

Отдельная ситуация возникает тогда, когда в качестве средства моделирования используется машинное обучение. Методы машинного обучения по своей природе являются эвристическими и в большинстве случаев не обеспечивают точного и единственного решения. Для них показателем репрезентативности выборки, на основе которой строилась обучаемая модель, является точность и обобщающая способность самой модели (эти вопросы рассматриваются в курсе машинного обучения).

Как улучшить репрезентативность выборки? Возможны следующие варианты [Орешков]:

- Коррекция выборки – замена ранее выбранных объектов совокупности. Выполняется, если в выборке произошло искажение распределения объектов относительно исходной совокупности, например, получился избыток пенсионеров, мужчин, женщин или людей с определённым уровнем образования. Замена может быть произвольной (например, следующий клиент по списку) или эквивалентной (подыскивается клиент с теми же параметрами — пенсионера меняем на пенсионера и т.д.).
- Расширение основы выборки – включение в выборку дополнительных элементов генеральной совокупности. Например, если изначально основой выборки являлась только люди пенсионного возраста, то при необходимости она может быть расширена и на людей предпенсионного возраста.
- Взвешивание – введение весовых коэффициентов, которые могут учитываться в алгоритме анализа. Например, повышенные весовые коэффициенты могут присваиваться клиентам, которые наиболее активно пользовались услугами компании (купили товаров и услуг на сумму выше некоторого порога).

Такие и подобные методы, как правило, входят в набор процедур подготовки данных.

3.2. Артефакты описания случайных величин в DS

3.2.1. Функция распределения и плотность вероятности

Функция распределения – вероятность того, что случайная величина примет значение меньше либо равное x, где x – произвольное действительное число:

$$F(x) = \Pr(X \le x) = \int_{-\infty}^{\infty} f(x)dx . \qquad (1)$$

Здесь $f(x)$ – плотность вероятности – вещественная функция, характеризующая сравнительную вероятность реализации тех или иных значений случайной переменной. Интеграл от $f(x)$ в некотором интервале от a до b есть вероятность обнаружить случайную величину в этом интервале:

$$\Pr(a \le x \le b) = \int_a^b f(x)dx. \qquad (2)$$

Сравнивая (1) и (2), видим, что плотность вероятности (рис. 3.2, б) – это производная функции распределения.

Понятно, что функция $F(x)$ должна быть неотрицательна и что интеграл должен быть нормирован на единицу.

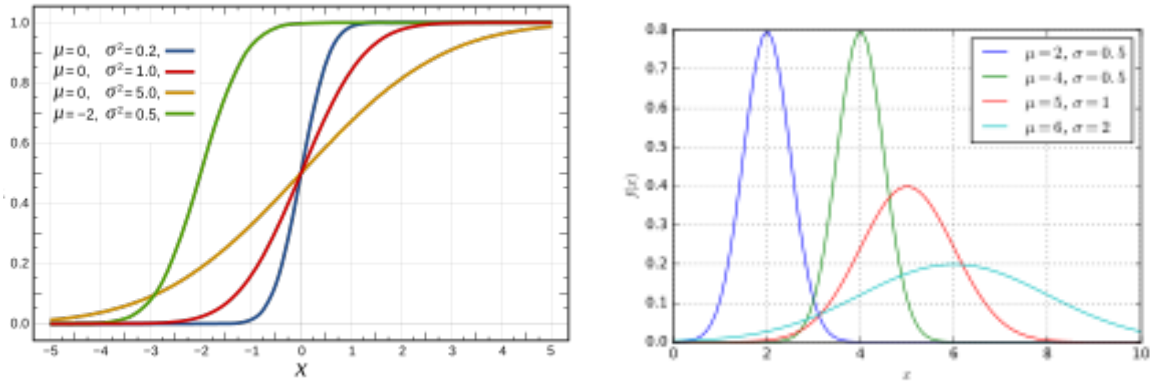
Если случайная величина X дискретна, т.е. ее распределение однозначно задается функцией вероятности

$$P(X=x)=p_i,\ i=1,2,\ldots,$$

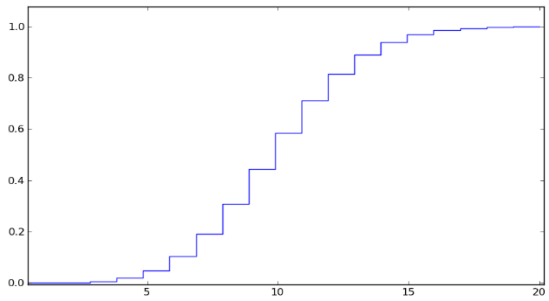
то функция распределения F_X этой случайной величины кусочно-постоянна (рис. 3.2, в) и может быть записана как

$$F_X(x) = \sum_{i:x_i<x} p_i. \qquad (3)$$

Для дискретной случайной величины плотность вероятностей не вводится.



а б



в

Рис. 3.2. а – функция распределения: а – для непрерывной случайной величины, б – плотность распределения для непрерывной случайной величины; в – функция распределения для дискретной случайной величины

3.2.2. Числовые характеристики случайных величин

Основную роль на практике играют:

- характеристики локации, т.е. положения центрального или типичного значения данных (математическое ожидание, мода, медиана, квантиль);
- характеристики масштаба, т.е. разброса значений, или вариативности, данных (дисперсия, стандартное отклонение, диапазон, среднее абсолютное отклонение, медианное абсолютное отклонение);

– характеристики симметрии данных (скошенность, или асимметрия, и эксцесс).

Характеристики локации. *Математическим ожиданием,* или *средним значением (mean), M(X)* дискретной случайной величины *X* называется сумма произведений всех ее значений на соответствующие им вероятности:

$$M(X) = \sum_{i=1}^{n} x_i p_i$$

для непрерывной случайной величины

$$M(X) = \int_{-\infty}^{+\infty} x f(x) dx$$

Модой (mode) Mo(X) случайной величины *X* называется ее наиболее вероятное значение (для которого вероятность p_i или плотность вероятности $p(x)$ достигает максимума). У случайной величины может быть несколько мод, соответствующих нескольким максимумам плотности вероятности (рис. 3.3, а). Если вероятность или плотность вероятности достигает максимума не в одной, а в нескольких точках, распределение называется *полимодальным.*

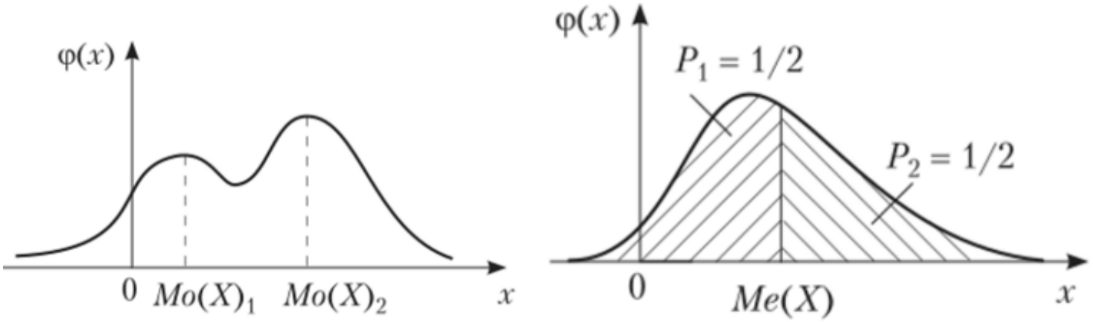


Рис. 3.3. Моды (а) и медиана (б) случайной величины

Медианой (median) Me(X) случайной величины *X* называется такое ее значение, для которого

$$P\left(X < Me(X)\right) = P\left(X > Me(X)\right) = \frac{1}{2},$$

Геометрически вертикальная прямая $x = Me(X)$, проходящая через точку с абсциссой, равной *Me(X)*, делит площадь фигуры иод кривой распределения на две равные части (рис. 3.3, б). Очевидно, что в точке $x = Me(X)$ функция распределения равна 1/2, т.е. $P(Me(X)) = ½$.

- Моды
- Геометрическое среднее: $\sqrt[n]{x_1 x_2 \ldots x_n}$
- Гармоническое среднее: $n/(x_1^{-1} + x_2^{-1} + \ldots + x_n^{-1})$
- Midhinge: $(Q_{0.25}(x) + Q_{0.75}(x))/2$
- Trimean: $(Q_{0.25}(x) + 2Q_{0.50}(x) + Q_{0.75}(x))/4$

Для выбора метрики нужно подумать о том, зачем вы считаете эту метрику, какие у нее статистические характеристики, на каких распределениях вы будете ее считать. Основное требование – робастность метрики: насколько она устойчива к добавлению новых значений в выборку. Например, средний рост участников вечеринки может очень сильно поменяться, если туда придет центровой-баскетболист. Только с учетом всех явных и неявных бизнес-требований можно принять разумное решение.

Квантилем уровня <q (или *q-квантилем)* называется такое значение x_q случайной величины, при котором функция ее распределения принимает значение, равное q:

$$F(x_q) = P(X < x_q) = q.$$

Некоторые квантили получили особое название. Очевидно, что медиана случайной величины есть квантиль уровня 0,5, т.е. $Me(X) = x_{05}$. Квантили $x_{0.25}$ и $x_{0.75}$ получили название соответственно *нижнего* и *верхнего квартилей.*

variance) D(X) случайной величины X называется математическое ожидание квадрата ее отклонения от математического ожидания:

$$D(X)=M[X-M(X)]^2 \qquad (*)$$

Для расчета дисперсии по выборке из N значений используется формула

$$s^2 = \sum_{i=1}^{n}(X_i - \widetilde{X})^2/(N-1),$$

где $\tilde{X}$ – среднее значение выборки.

Средним квадратическим отклонением (стандартным отклонением, standard deviation) случайной величины X называется арифметическое значение корня квадратного из ее дисперсии:

$$\sigma_x=\sqrt{D(X)}$$

Для выборки из N значений

$$s= \sqrt{\sum_{i=1}^{N}(X_i - \widetilde{X})^2/(N-1)}.$$

В случаях, если распределение не является нормальным, стандартное отклонение может быть обманчивым. На рис. 3.4 показаны три распределения A, B и C. Все они унимодальные и очень похожи на нормальное, но их стандартные отклонения равны, соответственно, 1, 11 и 3. Как же так получается?

На самом деле только распределение A является стандартным нормальным распределением с единичным стандартным отклонением. Распределение B на 95% состоит из стандартного нормального распределения и на 5% из нормального распределения со стандартным отклонением 49. Распределение C на 90% состоит из стандартного нормального распределения и на 10% из нормального распределения со стандартным отклонением 9. В статистике такие смеси часто используются для моделирования экстремальных выбросов.

$$\mathbf{A} = \mathcal{N}(0,1^2); \sigma_A = 1$$

$$B = 0.95\mathcal{N}(0,1^2) + 0.05\mathcal{N}(0,49^2); \sigma_B = \sqrt{0.95 \cdot 1^2 + 0.05 \cdot 49^2} = 11$$

$$C = 0.9\mathcal{N}(0,1^2) + 0.1\mathcal{N}(0,9^2); \sigma_C = \sqrt{0.9 \cdot 1^2 + 0.1 \cdot 9^2} = 3$$

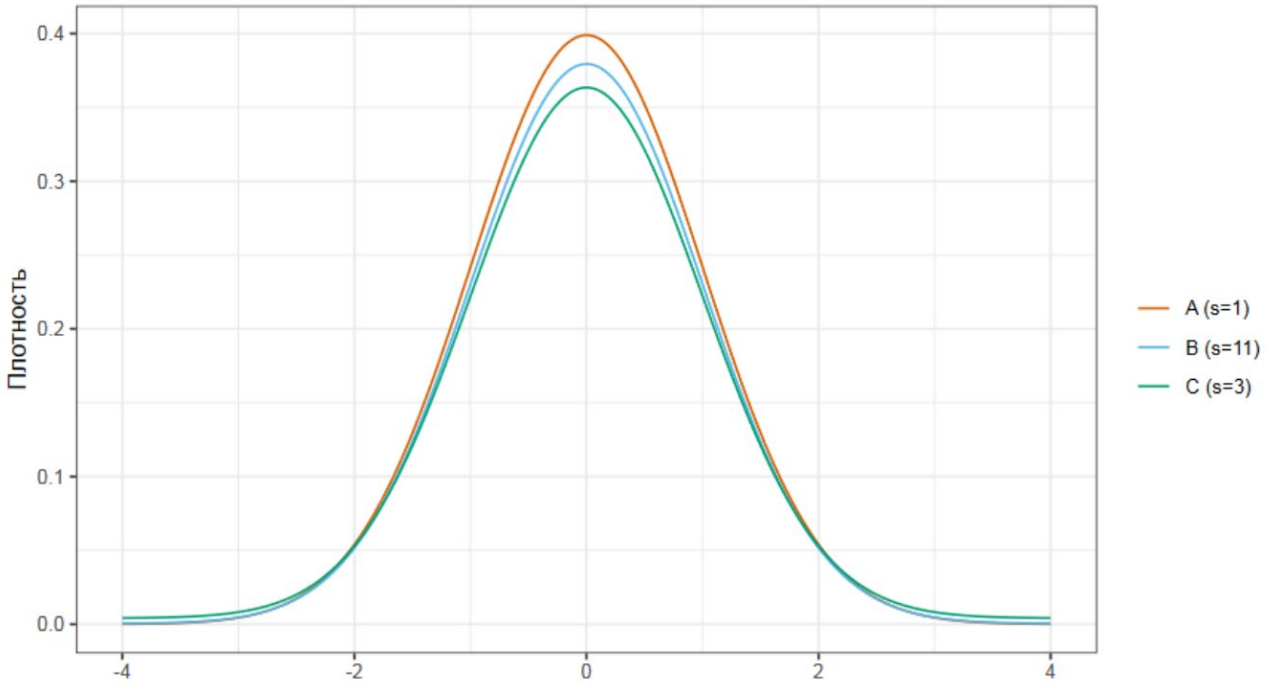


Рис. 3.4. Стандартные отклонения разных распределений (источник [https://habr.com/ru/companies/jugru/articles/722342/])

Существуют робастные альтернативы, в том числе AAD и MAD.

Среднее абсолютное отклонение (average absolute deviation, AAD) определяется как

$$ADD = \sum_{i=1}^{n}(X_i - \bar{X})/N$$

где $\bar{X}$ – это среднее значение данных, а $|X|$ – это абсолютное значение X. Эта мера не возводит в квадрат расстояние от среднего, поэтому на нее меньше влияют экстремальные наблюдения, чем на дисперсию и стандартное отклонение.

Медианное абсолютное отклонение (median absolute deviation, MAD) определяется как

$$MAD = median(|X_i - \tilde{X}|)$$

где $\tilde{X}$ – это медиана данных, а $|X|$ является абсолютным значением X. Это вариация *AAD*, на которую еще меньше влияют экстремумы на краях (хвостах) распределения, поскольку данные в хвостах оказывают меньшее влияние на расчет медианы, чем на среднее значение.

Диапазон (range) – это наибольшее значение минус наименьшее значение в наборе данных.

Межквартильный диапазон (interquartile range) – это значение 75-го процентиля минус значение 25-го процентиля. Эта мера масштаба пытается измерить изменчивость точек вблизи центра.

Характеристики симметрии данных. Скошенность, или асимметрия (skewness) – это мера симметрии или, точнее, отсутствия симметрии в наборе данных. Распределение или набор данных симметричны, если они выглядят одинаково слева и справа от центральной точки (рис. 3.5).

Для одномерных данных X_1, X_2, ..., X_N формула для асимметрии имеет вид

$$g_1 = \frac{\sum_{i=1}^{N}(X_i - \bar{X})^3/N}{s^3} \qquad (**)$$

где $\bar{X}$ - среднее значение данных, s – среднеквадратичное отклонение, N – число точек в выборке.

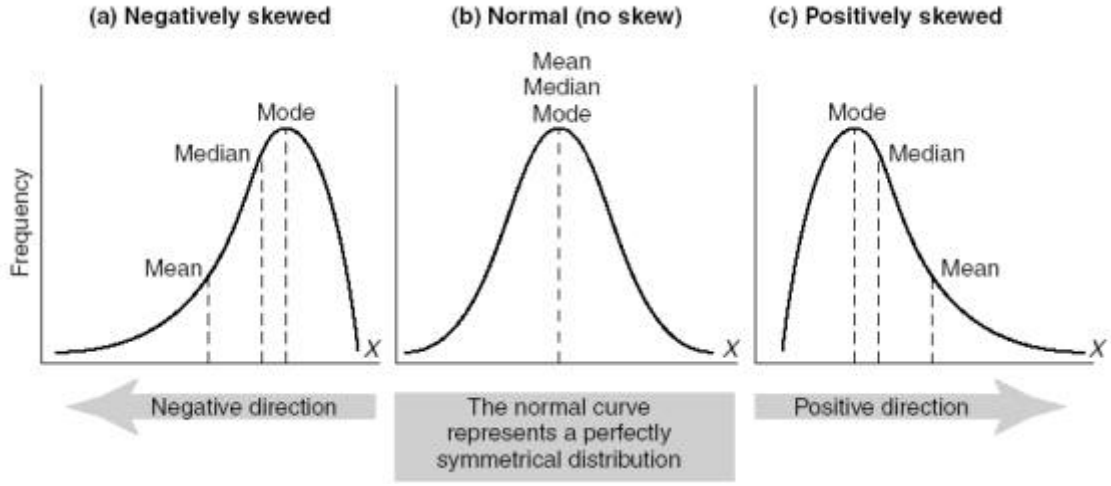
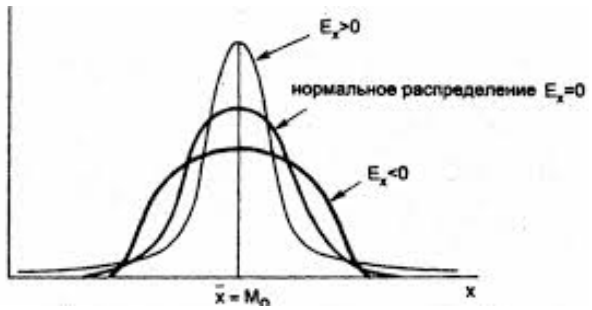


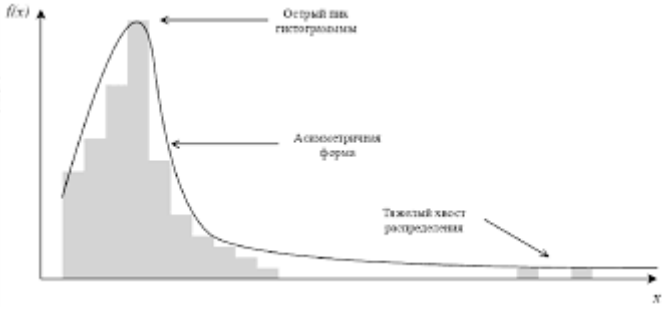
Рис. 3.5. Скошенные (асимметричные) распределения в сравнении с нормальным

Fisher-Pearson coefficient of skewness). Существуют также другие варианты: формула Гальтона, в которой асимметрия оценивается через соотношение нижнего и верхнего квартилей, и второй вариант формулы Пирсона, в которой сравниваются среднее и медианное значение для данной выборки. Однако именно формула (*) является наиболее распространенной, ее расчет входит практически во все статистические пакеты.

Эксцесс (*kurtosis*) – это мера того, насколько поведение данных в центральной зоне и в краевых зонах (на хвостах) отличается от нормального распределения (рис. 3.6, а). Наборы данных с низким эксцессом имеют плоскую вершину и легкие хвосты или отсутствие выбросов. Наборы данных с высоким эксцессом имеют острый пик в центре гистограммы и тяжелые хвосты или выбросы. Высокий эксцесс, как правило, создает большие проблемы при обработке данных, поэтому на рис. 3.6, б, этот случай показан более подробно.



а



б

Для одномерных данных X_1, X_2, ..., X_N классическая формула для эксцесса имеет вид

$$kurtosis = \frac{\sum_{i=1}^{N}(X_i - \bar{X})^4/N}{s^4}$$

где $\bar{X}$ – среднее значение данных, s – среднеквадратичное отклонение, N – число точек в выборке. Согласно этой формуле, значение эксцесса для стандартного нормального распределения равно 3. По этой причине некоторые источники используют следующее определение эксцесса (часто называемое «избыточным эксцессом»):

$$kurtosis = \frac{\sum_{i=1}^{N}(X_i - \bar{X})^4/N}{s^4} - 3 \qquad (***)$$

В этом случае стандартное нормальное распределение имеет эксцесс, равный нулю, положительный эксцесс указывает на распределение с «тяжелым хвостом», а отрицательный эксцесс – на распределение с «легким хвостом».

Моменты случайных величин. Сравнивая выражения (*), (**) и (***) для базовых числовых характеристик распределений, можно заметить определенную закономерность: все они основаны на вычислении выражения

$$\mu_k = M[X - M(X)]^k,$$

в котором показатель степени для (*) k=2, для (**) k=3, а для (***) k=4. Это выражение – *центральный момент k-го порядка* для случайной величины X – легко содержательно интерпретировать. Действительно, разность [X–M(X)] показывает, насколько текущий элемент датасета X «отпрыгивает» от центральной точки M(X). Если мы хотим получить усредненный показатель такого «отпрыгивания от центра», то просто складывать разности нельзя (значения с разными знаками уравновесят друг друга), поэтому мы усредняем квадраты разностей, т.е. берем k=2 – получается дисперсия (*). Если мы хотим оценить несимметричность распределения, то разумно использовать разницу «отпрыгиваний» в разные стороны от центральной точки; для этого нужен нечетный показатель степени k=3 – получается асимметрия (**). Наконец, эксцесс (***) с четным показателем k=4 показывает тонкие отличия поведения X от нормального распределения.

К сожалению, не все так просто: для многих распределений, имеющих содержательную интерпретацию и область применения в DS, высшие моменты (с k=2 и выше) расходятся (т.е. неограниченно увеличиваются с ростом числа

наблюдений N). Тем не менее, вычисление или хотя бы приближенная оценка моментов является мощным инструментом анализа данных.

Связи между случайными величинами. Существует два вида связей между переменными:

- функциональная зависимость – каждому значению переменной x соответствует строго одно значение y, т.е. $y=f(x)$ (при этом y и x могут быть векторами). По существу, это сильно упрощенная математическая модель;
- статистическая зависимость – каждому значению переменной x соответствует некоторое распределение вероятности переменной y.

Статистические связи могут быть разными: при изменении значения переменной x могут закономерным образом изменяться различные характеристики плотности распределения второй переменной y (матожидание, дисперсия, асимметрия и т.д.). В DS чаще всего оценивается связь по матожиданию, которая имеет специальное название – корреляционная связь.

Корреляционная связь означает, что каждому значению одной переменной соответствует определенное математическое ожидание другой переменной.

Корреляционная связь между случайными переменными x и y называется *линейной корреляционной связью*, если матожидание переменной y линейно зависит от значений переменной x, и наоборот, матожидание переменной x тоже линейно зависит от значений переменной y, т.е. имеет место взаимная линейность корреляционных связей.

Пусть математическое ожидание и дисперсия случайных величин X и Y равны, соответственно, μ_x и σ_x^2; μ_y и σ_y^2. Если X и Y независимы, то матожидание их произведения равно произведению их матожиданий по отдельности:

$$\mu_{xy} = \mu_x\mu_y$$

а для зависимых случайных величин это равенство не выполняется.

Ковариация – это отклонение математического ожидания произведения двух случайных величин от произведения их математических ожиданий:

$$\mathrm{cov}(x,y) = \sigma_{xy} = \sigma_{yx} = M(xy) - \mu_x\mu_y = M[(x-\mu_x)(y-\mu_y)].$$

Так как это отклонение существует только для зависимых величин, то ковариация характеризует степень этой зависимости. Чем она больше отличается от нуля, тем больше зависимость.

Матрица ковариаций для нескольких случайных величин X, Y, ..., Z всегда симметрична, причем на главной диагонали этой матрицы всегда стоят положительные числа, равные дисперсиям случайных величин X, Y..., Z:

$$\begin{pmatrix} \sigma_x^2 & \sigma_{xy} & \cdots & \sigma_{xz} \\ \sigma_{yx} & \sigma_y^2 & & \sigma_{yz} \\ \vdots & & \ddots & \vdots \\ \sigma_{zx} & \sigma_{zy} & \cdots & \sigma_z^2 \end{pmatrix}$$

Ковариация неудобна тем, что имеет размерность квадрата случайных величин. Кроме того, ковариация маленькой статистической зависимости двух случайных величин дисперсией (у хотя бы одной из этих величин) получается такой же, как большая статистическая зависимость у двух других случайных

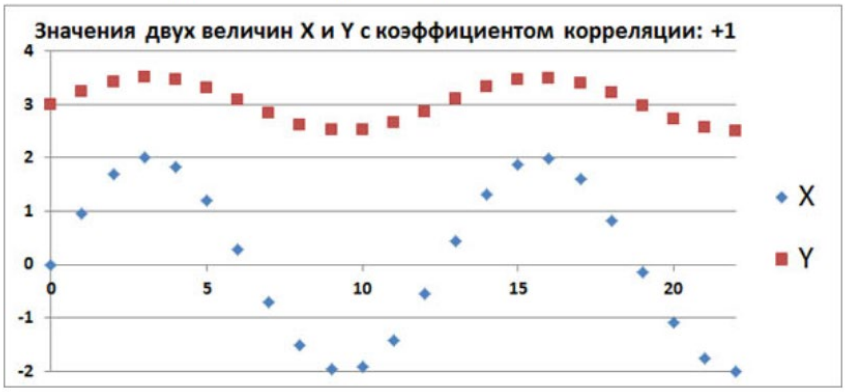
величин с маленькими дисперсиями. Поэтому ковариацию удобно нормировать на среднеквадратичные отклонения.

Коэффициент корреляции – это ковариация, нормированная на среднеквадратичные отклонения двух случайных величин.

$$\rho_{xy} = \rho_{yx} = M\left(\frac{(x-\mu_x)}{\sigma_x}\frac{(y-\mu_y)}{\sigma_y}\right) = \frac{M(xy) - \mu_x\mu_y}{\sigma_x\sigma_y}$$

Z симметрична, и на ее главной диагонали всегда стоят единицы.

$$\begin{pmatrix} 1 & \rho_{xy} & \cdots & \rho_{xz} \\ \rho_{yx} & 1 & & \rho_{yz} \\ \vdots & & \ddots & \vdots \\ \rho_{zx} & \rho_{zy} & \cdots & 1 \end{pmatrix}$$



(источник [https://chance.nanoquant.ru/correlation.htm])

EURUSD

Рис. 3.8. Изменения цен валютных пар евро–доллар (а) и евро–иена (б) в 2017 г. (источник [https://chance.nanoquant.ru/correlation.htm])

Коэффициент корреляции между независимыми случайными величинами равен нулю. Но обратное неверно! Если коэффициент корреляции двух случайных величин равен нулю, то они могут быть статистически связаны по другим числовым параметрам (дисперсии и пр.).

3.2.3. Функционалы качества для задач DS

Для измерения эффективности решения задач DS используются оценочные функционалы качества. Выбор конкретного функционала качества зависит от типа задачи DS (см. раздел 1.1.3).

Задачи оценки статистических гипотез (частотный подход). Здесь применяются статистические критерии, которые можно разделить на две группы: параметрические и непараметрические.

Параметрические критерии основаны на том, что распределение данных известно. Как правило, многие параметрические критерии предполагают нормальность распределения данных.

Стандартом де-факто среди параметрических критериев является *критерий p-value* (probability value). Смысл критерия рассмотрим на примере оценки улучшений дизайна сайта и поясним на рис. 3.9. Пусть эффективность дизайна оценивается числом кликов пользователей (откладывается по оси x); предполагаем, что среднее значение числа кликов распределено по нормальному закону.

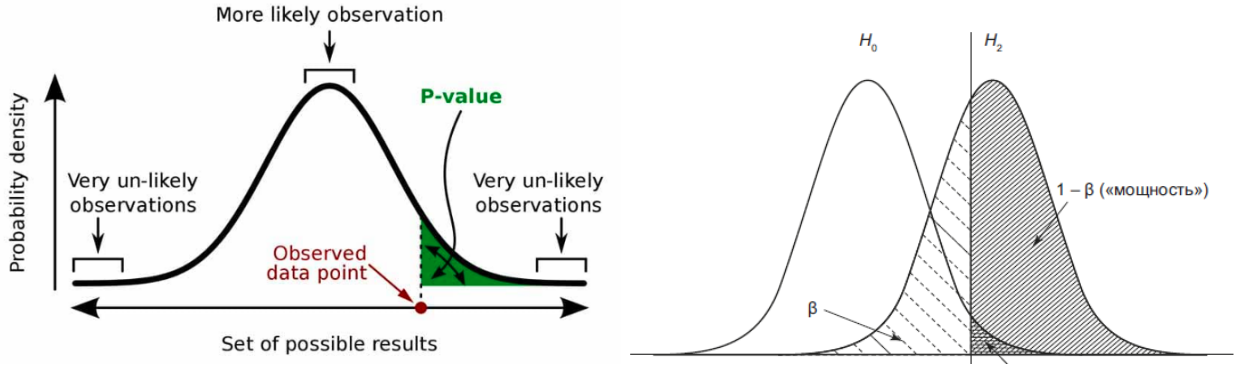


Рис. 3.9. Оценка эффективности дизайна сайта с помощью критерия p-value

В стандартном дизайне H_0 (рис. 3.9, а) среднее число кликов распределено по нормальному закону со средним x_0. Мы изменили дизайн и предполагаем, что это приведет к сдвигу ожидаемого распределения числа кликов до H_1 (рис. 3.9, б). Верно ли наше предположение?

Каждую версию сайта мы показываем отдельной группе пользователей, при этом считаем, что они (группы) статистически идентичны (отдельный вопрос – сколько будет стоить организация такого эксперимента). В конкретные дни на обоих версиях сайта мы получали среднее значение кликов Observed data point. О чем это говорит с точки зрения проверки нашего предположения?

«Фишка» критерия p-value (и за это его справедливо критикуют) заключается в том, что для ответа на этот вопрос мы стремимся не подтвердить гипотезу H_1, а опровергнуть гипотезу H_0. Для этого мы считаем две вероятности (рис. 3.9, б):

- того, что среднее число кликов будет больше или равно Observed data point, если выполняется H_0. Эта вероятность, очевидно, равна площади зеленого участка под кривой плотности распределения H_0. Эта площадь обозначается p-value, а левая граница p-value называется уровнем значимости α. Очевидно, что чем дальше Observed data point от x_0, тем абсурднее предположение о справедливости H_0;
- того, что среднее число кликов будет меньше или равно Observed data point, если выполняется H_1. Это площадь редко заштрихованной зоны распределения H_1, она обозначается β, а величина $(1–\beta)$ называется мощностью критерия. Содержательно β – это вероятность того, что гипотеза H_0 будет принята, хотя она и неверна (см. табл. 3.1).

Таблица 3.1. Ошибки статистических гипотез

Решение Реальность	Принять H_0	Отклонить H_0
H_0 верна	Ошибки нет	H_0 неверно отклонена (ошибка 1-го рода) с уровнем значимости α
H_0 неверна	H_0 неверно принята (ошибка 2-го рода) с вероятностью β	Ошибки нет

Таким образом, чтобы уверенно опровергнуть гипотезу H_0, нам нужно организовать эксперимент так, чтобы значения α и β оказались как можно меньше. В статистике принято считать эксперимент достоверным тогда, когда в нем $\alpha \le 0{,}05$ (а для особо ответственных случаев – даже $\alpha \le 0{,}01$). Как же этого добиться? Формально – очень легко: увеличить объем выборки, по которой мы проводим оценку. Вспомним, что в эксперименте мы сравниваем не идеальные распределения, показанные на рис. 3.9, а их выборочные статистики, т.е. гистограммы, при объеме выборки n. Оценка стандартного отклонения (т.е. ширины гистограммы) имеет вид

$$s = \sqrt{\frac{\sum(x - \bar{x})^2}{n - 1}}$$

где x – истинное среднее по всей популяции, а $\bar{x}$ – среднее по выборке. Следовательно, с ростом объема выборки n гистограммы «суживаются» до идеальных значений, тем самым уменьшая α и β.

Для расчета необходимого объема данных для тестов разработаны калькуляторы мощности. В калькулятор вводится минимальная детектируемая разность в значениях параметров (т.е. «расстояние» между H_0 и H_1), а также желаемый уровень α и β ошибок. На выходе будет объем данных, которые нужно собрать. Закономерность здесь проста – чем меньшую разницу вы хотите детектировать, тем больше данных для этого нужно.

Существуют модификации критерия p-value [Айвазян, 1985; Энциклопедия], которые в той или иной мере снимают жесткие предположения критерия p-value, Например, *t-критерий Стьюдента* ориентирован на применение в достаточно небольших выборках. Однако все они вносят свои допущения, не всегда приемлемые для конкретной задачи. Более того, сама методология критерия p-value давно подвергается справедливой критике, с которой можно познакомиться в специальной литературе.

Подчеркнем, что в реальной практике DS часто возникает принципиальное препятствие для использования критерия p-value – не существует или недоступна выборка необходимого объема. Например, для проверки эффективности лекарства для определенного заболевания нужна группа однотипных больных, а ее часто невозможно набрать – очень редкое заболевание, очень быстро нужны результаты, и т.п. Эта ситуация особенно остро проявилась в начальный период борьбы с эпидемией COVID-19.

Непараметрические критерии исходят из того, что распределение данных неизвестно. Поэтому при использовании этих критериев часто действия производятся не с самими значениями в выборке/выборках, а с их рангами. Непараметрические критерии используются для следующих переменных:

- для количественных переменных, распределение которых не подчиняется нормальному закону распределения;
- для переменных, измеренных в порядковой шкале;
- для переменных, измеренных в номинальной шкале.

Непараметрические тесты можно разделить на три группы:

- одновыборочные критерии позволяют проверить гипотезу о равенстве распределения выборки заданному (биномиальный критерий, критерий хи-квадрат, критерий Колмогорова-Смирнова, критерий знаков Вилкоксона, критерий серий и др.);
- критерии для независимых выборок позволяют проверять гипотезы о совпадении распределений в выборках (критерий Манна-Уитни, критерий Колмогорова-Смирнова для двух выборок, критерий Вальда-Вольфовица, критерий Мозеса, непараметрический дисперсионный анализ Крускала-Уоллиса, медианный критерий, критерий Джонкхира-Терпстры и др.);

– критерии для зависимых или связанных выборок позволяют сравнить повторные измерения на одних и тех же объектах (критерий МакНемара, критерий Кохрана, критерий согласия Кендалла, непараметрический дисперсионный анализ Фридмана, критерий знаков, критерий знаковых разностей Вилкоксона, критерий маргинальной однородности и др.).

Байесовская статистика. В этом случае вместо утверждений «гипотеза истинна» или «гипотеза ложна» вводится «гипотеза истинна с вероятностью p».

Обозначим априорные знания о величине θ как $p(\theta)$. В процессе наблюдений мы получаем серию значений $x = (x_1, ..., x_n)$. При разных θ наблюдение выборки x более или менее вероятно и определяется значением правдоподобия $p(x|\theta)$. За счет наблюдений наши представления о значении θ меняются согласно формуле Байеса:

$$p(\theta|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\theta)p(\theta)}{p(\boldsymbol{x})} = \frac{p(\boldsymbol{x}|\theta)p(\theta)}{\int p(\boldsymbol{x}|\theta)p(\theta)d\theta}$$

В знаменателе стоит нормировочный интеграл. Его значение от θ не зависит, но его вычисление в общем случае представляет проблему, особенно если θ – вектор высокой размерности. К счастью, существуют пары распределений (они называются сопряженными), для которых интеграл в знаменателе берется аналитически. Поэтому априорное распределение $p(\theta)$ имеет смысл выбирать из класса распределений, сопряженных правдоподобию $p(x|\theta)$. Тогда ответ, т.е. апостериорное распределение $p(\theta|x)$, можно будет записать в явном виде. Такие пары распределений хорошо известны, все расчеты для них приводятся в справочниках.

Можно идти другим путем – подбирать апостериорное распределение численными методами, т.е. в ходе итераций. Для этого используется метод Монте-Карло для марковских цепей (MCMC). Алгоритм выбирает случайное значение из исходной выборки и по определенному правилу принимает или отвергает это значение в зависимости от его соответствия данным эксперимента и априорному распределению. Этот процесс повторяется много (несколько тысяч) раз, после чего обычно сходится к апостериорному распределению. На выходе получается большая выборка из распределения.

Сейчас есть большое количество библиотек, предоставляющих различные инструменты для воплощения байесовского подхода к анализу данных.

Сравнение частотного и байесовского подходов на примере А/Б теста. А/Б-тестирование, или сплит-тестирование, – это метод исследования, при котором сравнивают эффективность двух вариантов какого-то объекта, например страницы сайта. Эти варианты показывают аудитории и оценивают, на какой из них потребители реагируют лучше.

В эксперименте [Лукьянова] в качестве показателя эффективности использована конверсия λ – доля посетителей сайта, которые совершили целевое действие (например, купили товар), от общего числа посетителей сайта. Для эксперимента сгенерировали распределения конверсий для вариантов сайта А и Б с разницей средних значений (эффектом) 10%, 25% и 50% и провели 10 000 итераций: рандомно брали из каждого распределения по 1000 наблюдений, для каждой итерации подводили итог по байесовскому и частотному подходу. В

качестве априорного распределения использована бета-функция с параметрами (1, 1), что эквивалентно равномерному распределению (т.е. до начала эксперимента нет никаких предположений о предпочтениях пользователей). Для байесовского подхода использован порог 0,95 для вероятности того, что конверсия в группе А больше, чем в группе Б, т.е. $p(\lambda_А>\lambda_Б)>0{,}95$. Для частотного подхода выбран порог для *p*-value – 0,05. Результаты представлены в таблице 3.2.

Таблица 3.2. Сравнение оценок результатов А/Б тестирования

Размер выборки/ Эффект	10%		25%		50%	
	Ч	Б	Ч	Б	Ч	Б
15	2,7	0,5	3,5	0,9	5	2
50	5,6	5,1	7	8	12	14
100	6	7	8	12	18	25
500	8	12	21	30	59	71
1000	10	16	36	49	92	96

Значения в ячейках – это доля итераций, где разница была обнаружена. Например, в зеленых ячейках байесовский подход задетектил разницу в 16% случаев, частотный – в 10%. Другими словами, с ростом выборки байесовский подход становится все более чувствительным (чаще видит отличия там, где частотный подход их не видит).

Конечно, этот эксперимент не абсолютно строг (его детальную критику можно найти в комментариях к [Лукьянова]). Однако он наглядно демонстрирует достоинства байесовского подхода:

- содержательно интерпретируемый результат можно получить на любом объеме данных. На практике бывает, что тест с достаточными выборками провести не получается, а выводы все равно сделать хочется. Там, где *p*-value просто скажет об отсутствии статистически значимых отличий, Байес покажет, насколько велика разница между группами по разным критериям и насколько эта разница вероятна. Но здесь нужно быть осторожным – чем меньше данных, тем меньше уверенность в сделанных выводах (детальнее про расчет уверенности можно прочитать в литературе);
- байесовский подход обычно более чувствителен к различиям в группах.

Однако частотный подход также имеет свои достоинства:

- простота. В противовес подходу p-value, в простом калькуляторе или Excel-табличке вычисления байесовой статистики не проделать [Жолудова];
- гораздо более традиционен и лучше воспринимается заказчиками – возможно, это дело привычки, которую нужно ломать!

Задачи классификации. Здесь функционал качества чаще всего определяется как средняя ошибка модели на тестовой выборке (предполагается, что искомый алгоритм должен его минимизировать). Гипотеза в явном виде здесь отсутствует, а проверка производится по матрице ошибок (confusion matrix, таб-

лица сопряженности), которая показывает, совпали ли ответы модели $\hat{y}$ с заранее размеченными ответами y (табл. 3.3).

Таблица 3.3. Матрица ошибок

Разметка Ответ модели	y=0	y=1
$\hat{y} = 0$	True Positive — TP	False Positive — FP
$\hat{y} = 1$	False Negative — FN	True Negative — TN

На основании матрицы ошибок, заполненной по тестовой выборке, формируются различные метрики, в том числе:

– Accuracy (аккуратность, меткость)

$$ACC = \frac{TP+TN}{TP+TN+FP+FN}.$$

– Precision (точность) = positive predictive value (PPV)

$$Pr = PPV = \frac{TP}{TP+FP}.$$

– Recall (полнота) = true positive rate (TPR) = sensitivity (чувствительность)

$$Re = TPR = \frac{TP}{TP+FN}.$$

– Specificity (чувствительность)

$$Sp = TNR = \frac{TN}{TN+FP}.$$

– False positive rate (показатель ложной тревоги)

$$1 - TNR = FPR = \frac{FP}{FP+TN}.$$

– F1-measure (F1-мера)

$$F1 = \frac{2 \cdot PPV \cdot TPR}{PPV+TPR} = \frac{2 \cdot TP}{2 \cdot TP+FP+FN}.$$

Смысл этих метрик проиллюстрирован на рис. 3.10.

Комплексной графической метрикой качества классификатора является ROC-кривая, или кривая рабочих характеристик приемника (Receiver Operating Characteristics curve) (рис. 3.11). Она строится в координатах FPR – TPR, или (1–специфичность) – чувствительность. Идеальная модель не допускает никаких ошибок, т.е. для нее вероятность правильных ответов составляет 100%; в этом случае ROC-кривая проходит по граням квадрата, т.е. через точки (0, 0), (0, 1) и (1, 1), а площадь под кривой (Area Under Curve, AUC) равна 1. Необученная модель воспроизводит случайный поиск, т.е. для нее вероятность любого ответа составляет 50%; в этом случае ROC-кривая совпадает с диагональю квадрата, а площадь под кривой равна 0,5. Реальная модель занимает некоторое промежуточное положение, причем чем выше и левее расположена кривая, тем больше предсказательная способность модели.

Метрика ROC-AUC наглядна, хорошо интерпретируется, но не отражает изменения баланса классов, а в условиях дисбаланса классов завышает качество модели.

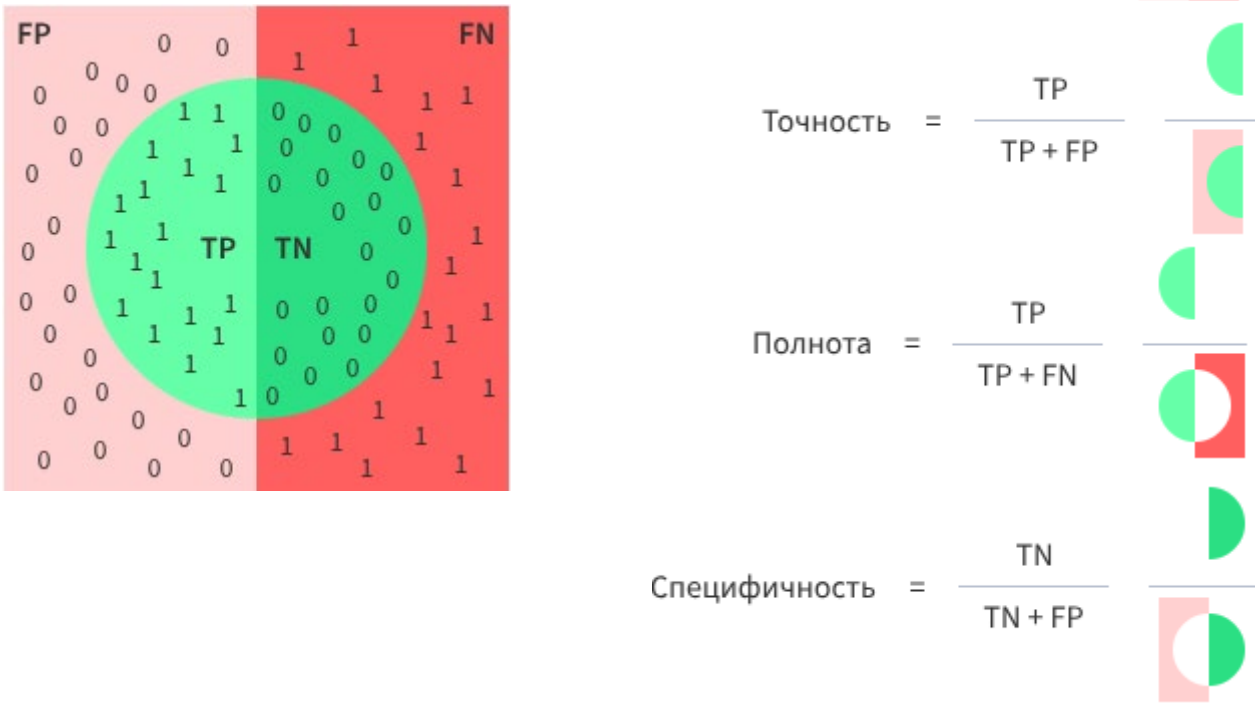


Рис. 3.10. Метрики качества классификации
(источник [https://loginom.ru/blog/classification-quality])

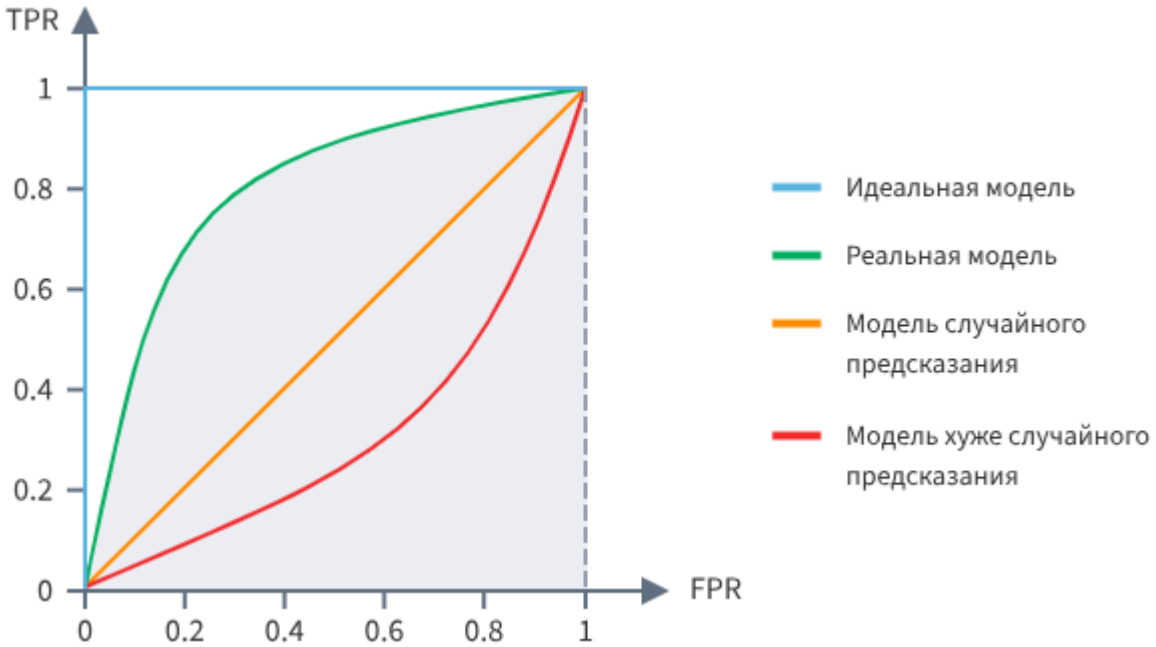


Рис. 3.11. ROC-кривая (источник [https://loginom.ru/blog/classification-quality])

Задачи регрессии. Здесь применяются различные функционалы качества, в том числе:

– Средняя квадратичная ошибка (англ. Mean Squared Error, MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(a(x_i) - y_i)^2$$

– Средняя абсолютная ошибка (англ. Mean Absolute Error, MAE)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|a(x_i) - y_i|$$

MSE применяется тогда, когда нужно выбрать модель, которая дает меньше больших ошибок прогноза (за счет того, что ошибку прогноза мы возводим в квадрат), но MSE более чувствителен к выбросам по сравнению с MAE.

– Коэффициент детерминации (нормированная MSE) – измеряет долю дисперсии, объясненную моделью, в общей дисперсии целевой переменной.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(a(x_i) - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Кросс-валидация (скользящий контроль, или перекрестная проверка) [Скользящий] является стандартной методикой тестирования и сравнения алгоритмов классификации, регрессии и прогнозирования.

В этом случае фиксируется некоторое множество разбиений исходной выборки на две подвыборки: обучающую и контрольную. Для каждого разбиения выполняется настройка алгоритма по обучающей подвыборке, затем оценивается его средняя ошибка на объектах контрольной подвыборки. Оценкой скользящего контроля называется средняя по всем разбиениям величина ошибки на контрольных подвыборках. При этом можно получить несмещенную оценку вероятности ошибки.

Задачи кластеризации. Так как задача кластеризации не может быть решена однозначно, то и функционалы качества здесь могут определяться поразному. К настоящему времени предложено более 30 вариантов (см. сводку в [https://neerc.ifmo.ru/wiki/index.php?title=Оценка_качества_в_задаче_кластеризации]), которые делятся на две группы:

– внешние (External) меры основаны на сравнении результата кластеризации с априори известным разделением на классы;

– внутренние (Internal) меры отображают качество кластеризации только по информации в данных.

В работе [Arbelaitz] показано, что для искусственных датасетов лучше использовать меры DB*, Silhouette и Calinski−Harabasz, а на реальных датасетах – Score function. Все они являются внутренними и основаны на сопоставлении компактности (Cohesion) и отделимости (Separation) кластеров. Компактность выражает идею о том, что чем ближе друг к другу находятся объекты внутри кластеров, тем лучше разделение: у компактных кластеров должно быть минимально внутриклассовое расстояние. Отделимость выражает противоположную идею: чем дальше друг от друга находятся объекты разных кластеров, тем лучше.

В частности, *DB** (индекс Дэвиса-Болдуина, Davies–Bouldin Index) – одна из самых используемых мер оценки качества кластеризации – вычисляет компактность как расстояние от объектов кластера x_i до их центроидов c_k, а отделимость – как расстояние между центроидами l-го и k-го кластеров:

$$DB^*(C) = \frac{1}{K} \sum_{c_k \in C} \frac{\max\limits_{c_l \in C \setminus c_k} \{S(c_k) + S(c_l)\}}{\min\limits_{c_l \in C \setminus c_k} \{\|\overline{c_k} - \overline{c_l}\|\}}$$

где

$$S(c_k) = \frac{1}{|c_k|} \sum_{x_i \in c_k} \|x_i - \overline{c_k}\|$$

Чем меньше *DB**(*C*), тем лучше качество кластеризации *C*.

В метрике Score function компактность оценивается как расстояние от точек кластера до его центроида, а разделимость – как расстояние от центроидов кластеров до глобального центроида.

$$SF(C) = 1 - \frac{1}{e^{e^{bcd(C) - wcd(C)}}},$$

где

$$bcd(C) = \frac{\sum\limits_{c_k \in C} |c_k| \cdot \|\overline{c_k} - \overline{X}\|}{N \times K},$$

$$wcd(C) = \sum_{c_k \in C} \frac{1}{|c_k|} \sum_{x_i \in c_k} \|x_i - \overline{c_k}\|$$

Чем больше *SF*(*C*), тем выше качество кластеризации *C*.

Задачи сегментации являются частным случаем задачи кластеризации, но имеют свою содержательную специфику, которая во многом определяет выбор функционала качества.

Сегментация изображения рассматривается как задача поиска групп пикселей, каждая из которых характеризует один смысловой объект. Базовым алгоритмом является графо-ориентированная сегментация (Graph-based segmentation). Здесь содержательно вводится мера отличия конкретного пикселями от окружающих восьми соседних (например, разница цветовой интенсивности). Тогда для любого региона изображения *R* его внутренняя разница определяется как наибольшая мера отличия в минимальном остовном дереве региона:

$$Int(R) = \min_{e \in MST(R)} w(e)$$

Алгоритм объединяет любые две соседние области, разница которых меньше минимальной внутренней разности этих двух областей,

$$MInt(R_1, R_2) = \min(Int(R_1) + \tau(R_1), Int(R_2) + \tau(R_2)),$$

в порядке убывания веса разделяющих их ребер, где τ(*R*) – эвристически задаваемый штраф по региону. По существу, алгоритм воспроизводит древовидную структуру регионов изображения, где каждый регион имеет пиксели примерно

одинаковой яркости. Результат работы алгоритма проиллюстрирован на рис. 3.12.

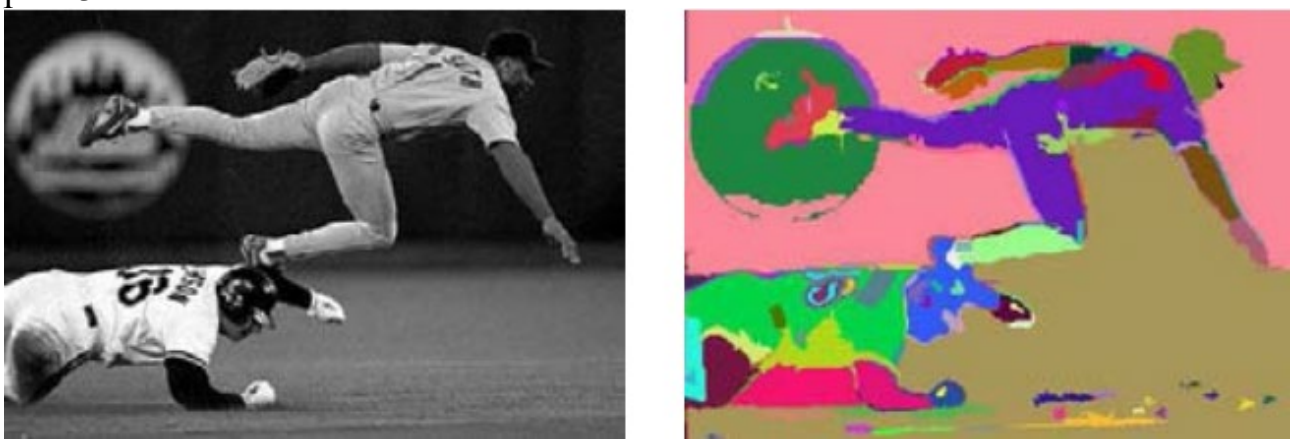


Рис. 3.11. Графо-ориентированная сегментация (источник [https://neerc.ifmo.ru/wiki/index.php?title=Сегментация_изображений])

Регионы на изображении в общем случае образуют не строго древовидную, а произвольную графовую структуру. В этом случае задача сегментации решается как задача поиска разреза на графе. Например, метод нормализованных срезов (Normalized cuts) исследует сходство между соседними пикселями и пытается разделить их на группы (A и B), которые, в свою очередь, связаны слабо (рис. 3.12).

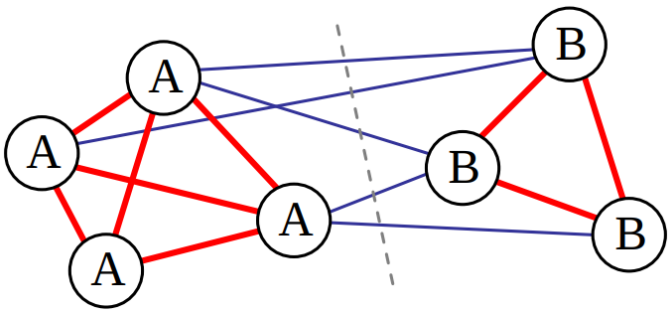


Рис. 3.12. Сегментация методом нормализованных срезов (источник [https://neerc.ifmo.ru/wiki/index.php?title=Сегментация_изображений])

При сегментации изображений с помощью глубоких нейронных сетей однородные регионы изображения выделяются в процессе свертки. Классификация достигается за счёт выбора максимума по классам из значений тензора размерности C×W×H, где C – множество классов, заранее заданных перед обучением и к которым могут принадлежать пиксели изображения, W×H – размер изображения. Такую модель можно обучить при помощи обратного распространения ошибок, а в качестве функции потерь для пикселей использовать кросс-энтропию. При этом на входе формируется тепловая карта нахождения классов на изображении, соответствующая разным размерам сверток.

На сегодняшний день общепризнанным (state-of-the-art, SOTA) подходом является метод расширенных сверток [Chen]. Расширенная свертка заключается в том, чтобы применять свертки с ядрами разного размера и разным шагом над

прямоугольниками с одним и тем же центром, а впоследствии комбинировать полученные таким образом признаки (рис. 3.13).

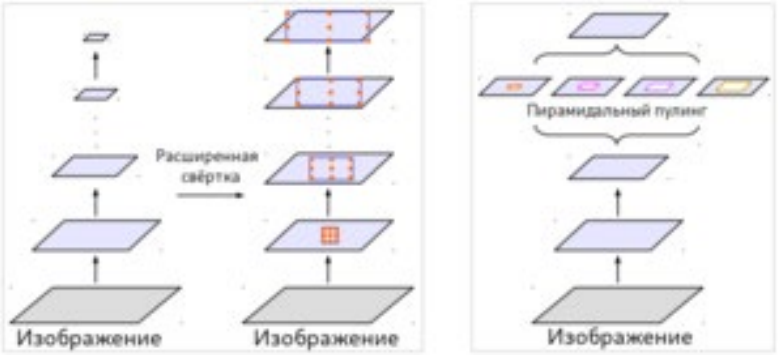


Рис. 3.13. Сегментация методом расширенных сверток (источник [https://neerc.ifmo.ru/wiki/index.php?title=Сегментация_изображений])

Расширенные свертки могут применяться как каскадно (рис. 3.13, посередине), так и параллельно, т.е. на одном и том же слое сверточной сети с пулингом в конце (рис. 3.13, справа). Такой подход позволил достичь лучших результатов в изображениях с объектами разных масштабов.

Столь разные алгоритмические подходы к сегментации изображений не позволяют сформировать для этой задачи единую внутреннюю метрику качества, поэтому здесь используются внешние (External) меры. В качестве эталона для сравнения выбирается, как правило, экспертная разметка: правильно сегментированные фрагменты изображения должны совпадать с теми, которые выделил эксперт. Совпадение оценивают либо по площади (наиболее частый вариант), либо по контуру.

При сравнении по площади используются следующие метрики:

– Попиксельная точность (pixel accuracy)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

где $TP + TN$– количество правильно проклассифицированных пикселей (true positives + true negatives), $TP+TN+FP+FN$–общее количество пикселей (см. рис. 3.14)

		Предсказание	
		True	**False**
Разметка	**True**	TP	FN
	False	FP	TN

Рис. 3.14. Построение метрик для сегментации изображений

– Метрика IoU (Intersection over Union), или индекс Жаккара (Jaccard index)

$$IoU = \frac{TP}{TP + FP + FN}$$

– Индекс Дайса (Dice index), или F1-score (рис. 3.15)

$$DICE = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

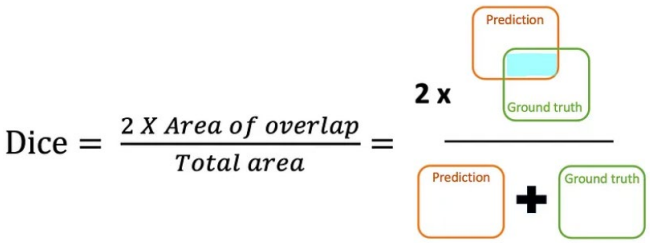


Рис. 3.15. Графическое представление индекса Дайса

При сравнении совпадения контуров выделенных фрагментов используются эвристические метрики расстояния между двумя кривыми, в том числе:

– Расстояние Хаусдорфа (рис. 3.16) определяется как

$$d_H(X,Y) = \max\left\{\sup_{x\in X}\inf_{y\in Y} d(x,y), \sup_{y\in Y}\inf_{x\in X} d(x,y)\right\}.$$

В основе определения лежит следующая интуиция: содержательно два множества «близки», если для любой точки любого множества ближайшая точка другого множества «не слишком далека».

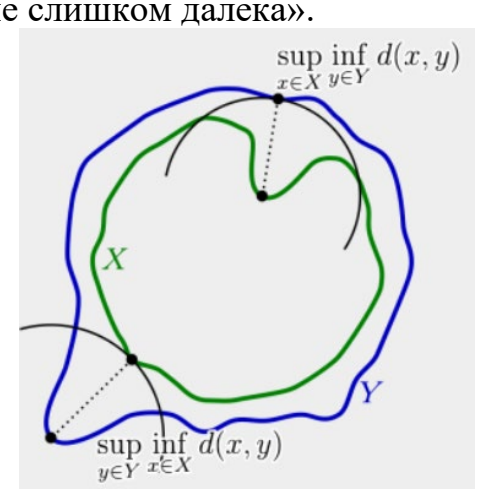


Рис. 3.16. К расчету расстояния Хаусдорфа
(источник [https://en.wikipedia.org/wiki/Hausdorff_distance])

– Расстояние Фреше (рис. 3.17)

$$D_{Frechet}(A, B) = \min_{\mu}\max_{a\in A} d(a, \mu(a))$$

где μ: A → B – взаимно-однозначное непрерывное отображение из точки на траектории A в точку на траектории B.

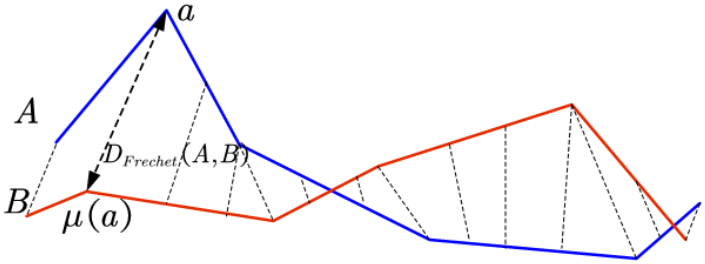


Рис. 3.17. К расчету расстояния Фреше (источник [Guo])

Если представить себе красную и синюю линию как траектории движения хозяина и собаки, которую он ведет на поводке, то расстояние Фреше – это минимальная длина поводка, которая позволяет этой паре следовать своими траекториями (интересно, что определение симметрично, т.е. не задает формально, кто кого выгуливает!)

Сегментация целевой аудитории является одной из главных задач в работе дата-сайентиста, так как эффективная сегментация целевой аудитории предопределяет успех кампании по продвижению любой продукции – товара, услуги или идеи.

Сегментация целевой аудитории [Корнелюк] – это процесс, в котором происходит деление всех потенциальных клиентов на группы с общими характеристиками.

Хотя формально задача сегментации является одним из видов задачи кластеризации, т.е. обучения без учителя, в реальной практике сегментации целевой аудитории чаще используется классификации по типовым наборам классифицирующих признаков, например:

- методика 5W (Who, What, When, Where, Why) [Шеррингтон] – пять групп вопросов о потенциальной аудитории (потребитель – продукт – время – место – мотивация);
- методика Jobs to Be Done (JTBD) [JTBD] – два варианта вопросов о потенциальной аудитории (ситуация – мотивация – результат или тип клиента – задача клиента – результат).

Таблица 3.4. Сегментация клиентов туристического агентства

Подход к сегментации	Классификация	Кластеризация
Инструменты выявления групп	Типовые признаки – возраст, пол, локация	Шаги процесса: 1. Интервью (опросы среди текущих и потенциальных клиентов). 2. Анализ полученных данных (что важнее для клиентов)
Выявленные группы	Возраст: 18–30 лет (молодежь, ищущая приключения), 31–50 лет (семьи и без семьи); Пол: мужчины и женщины Локация: города с высоким уровнем дохода	Доминирующие сегменты: клиенты, ищущие активный отдых; клиенты, интересующиеся историей и искусством; семьи, ищущие удобные и безопасные поездки
Результат сегментации	Предлагаем туры со скидкой	Предлагаем уникальные туры по интересам: приключенческие туры, культурные туры, семейные туры

Однако для поиска уникальных сегментов рынка, т.е. для формирования гипотезы продвижения товаров и услуг, нужен подход кластеризации:

1. Интервью – общаемся с текущими и потенциальными клиентами.
2. Анализ данных – выявляем общие черты и потребности.

3. Деление на сегменты – обозначаем вновь выявленные сегменты с общими характеристиками.

Таким образом, сегментация одной и той же аудитории может дать совершенно разные результаты, что хорошо видно на примере сегментации клиентов туристического агентства (табл. 3.4).

Независимо от того, какой подход использовался при сегментации целевой аудитории, ее главная цель – получение прибыли для бизнеса. Поэтому все метрики, используемые для оценки эффективности сегментации целевой аудитории, являются сугубо целевыми, т.е. оценивают то, насколько полученная сегментация эффективна для поддержки бизнеса (табл. 3.5) [Корнелюк].

Таблица 3.5. Метрики оценки эффективности сегментации клиентов

Оцениваемый показатель	Метрика оценки эффективности выполнения шага сегментации
Оценка релевантности сегментов	Сегменты соответствуют стратегии бизнеса и маркетинговым целям; Демографические, психографические и поведенческие характеристики сегментов подтверждаются внешними данными
Измерение размера сегментов	Оценен размер каждого сегмента в цифровом и процентном выражении; Оценена динамика изменения размера сегментов во времени
Оценка доступности и достижимости	Определены доступные каналы коммуникации с каждым сегментом (онлайн, офлайн, социальные сети и т. д.); Оценена готовность и возможность сегментов взаимодействовать с предложениями компании
Измерение рентабельности	Рассчитана потенциальная рентабельность каждого сегмента – стоимость привлечения клиента (CAC), прибыль от одного клиента (LTV), разница между себестоимостью товара и ценой, по которой он продается клиенту (маржинальность); Оценены затраты на привлечение и обслуживание клиентов в каждом сегменте
Отслеживание метрик успешности	Следим за ключевыми показателями (KPI) для каждого сегмента (конверсия, удержание клиентов, средний чек и др.); Сравниваем метрики между сегментами для определения наиболее успешных
Анализ изменений и корректировка стратегий	Осуществляем регулярный мониторинг изменений в рыночной среде и поведении клиентов; Корректируем стратегии сегментации и маркетинга в соответствии с новыми данными и условиями рынка

Задача поиска ассоциативных правил. Ассоциативные правила позволяют находить закономерности между связанными событиями. Такая задача, очень часто встречается в практике DS, но не имеет общеформальной постановки, т.е. ассоциацию нельзя строго определить в общем виде. Соответственно, и критерии ее решения определяются содержательно.

Типичной задачей поиска ассоциативных правил является *анализ рыночной корзины (market basket analysis)*. Пример такого правила: покупатель, приобретающий «Хлеб», приобретет и «Молоко» с вероятностью 75%.

Простейший вариант задания ассоциативных правил основан на бинарной (булевской) логике [Поддержка]. Пусть $I = i_1, i_2, i_3, \ldots, i_n$ – множество (набор) товаров, называемых элементами, а D – множество транзакций T, где каждая из них является набором элементов из I,$T \subseteq I$. Любая транзакция представляет собой бинарный вектор, где $t[k]=1$, если i_k элемент присутствует в ней, иначе $t[k]=0$. Мы говорим, что транзакция T содержит X, некоторый набор элементов из I, если $X \subseteq T$. Ассоциативным правилом называется импликация $X \rightarrow Y$, где $X \subset I$, $Y \subset I$ и $X \cap Y = \emptyset$.

Другими словами, ассоциативное правило в этом случае фиксирует следующую зависимость: если в транзакции встретился некоторый набор элементов X, то в ней появится и другой набор элементов Y.

Качество ассоциативного правила задается показателями Support (поддержка) и Confidence (достоверность). Правило $X \Rightarrow Y$ имеет поддержку s (support), если s% транзакций из D содержат $X \cup Y$, т.е.

$supp\ (X \Rightarrow Y) = supp\ (X \cup Y)$.

Достоверность правила показывает, какова вероятность того, что из X следует Y. Правило $X \Rightarrow Y$ справедливо с достоверностью (confidence) c, если c% транзакций из D, содержащих X, также содержат Y, т.е.

$conf\ (X \Rightarrow Y) = supp\ (X \cup Y)/supp(X)$.

Например, если 75% транзакций, содержащих хлеб, также содержат молоко, при этом 3% от общего числа всех транзакций содержат оба товара, то 75% – это достоверность (confidence) правила, а 3% – это поддержка (support).

Задача нахождения ассоциативных правил решается в два этапа:

- нахождение всех наборов элементов, которые удовлетворяют порогу *minsupport*. Такие наборы элементов называются часто встречающимися;
- генерация правил из часто встречающихся наборов элементов с достоверностью не хуже порога *minconfidence*.

Разработано множество алгоритмов, эффективно решающих подобный класс задач; первым был предложен алгоритм APriori [Agrawal], который входит во все статистические пакеты, но сейчас разработаны алгоритмы с более высокой вычислительной эффективностью. Кроме сугубо бинарных показателей, они позволяют включать в анализ ассоциаций и разнородные атрибуты, в том числе числовые и категориальные (возраст и доход покупателей, бренд товара, время покупки и т.п.).

Основной задачей дата-аналитика здесь является удачный выбор порогов *minsupport* и *minconfidence:* нужно ограничить количество найденных правил и

при этом не потерять потенциально важные. Если значение *minsupport* велико, то большинство найденных правил будет тривиальным (например, в большинство покупок входит пакет). С другой стороны, низкие значения *minsupport* порождают огромное количество правил, находящихся на грани шумовых (т.е. фиксируются случайные комбинации покупок), что, конечно, требует существенных вычислительных ресурсов. Тем не менее, большинство интересных правил находится именно при низком значении порога поддержки.

3.3. Описание распределений в DS

3.3.1. Предварительные замечания

Вероятностные распределения – это основа для применения методов теории вероятностей и (опосредованно) статистики. Без выбора, хотя бы предварительного, вида вероятностного распределения дата-сайентист не может решить поставленную перед ним задачу. С какими же вызовами он сталкивается при этом?

Во-первых, в математике существуют сотни различных распределений, они сложным образом взаимосвязаны, и хотя бы их классифицировать – уже самостоятельная задача, результаты которой впечатляют и даже пугают. В качестве примера на рисунке 3.9 представлен упрощенный вариант концепт-карты взаимосвязей основных видов функций распределения вероятностей. Подробную интерактивную схему, содержащую 76 распределений вероятностей (19 дискретных и 57 непрерывных распределений) можно найти в [Univariate].

Во-вторых, хотя многие распределения визуально очень похожи друг на друга, их неудачный выбор может привести к тяжелым, если не катастрофическим, последствиям. Например, на первый взгляд два распределения (рис. 3.10) почти одинаковы. Более детальное рассмотрение показывает, что распределения Коши имеет более острый, по сравнению с распределением Гаусса, пик на медиане плотности вероятности (наиболее очевидный при $\gamma = 1$ на рис. 3.10), а также более широкие крылья на распределении вероятности (особенно очевидные при $|x| \to \infty$), что может указывать на наличие тяжелых хвостов.

Казалось бы, разница не так уж велика. Однако, согласно [SmartRisk], оценка риска инвестиционного портфеля по стандартной методике 60/40 Portfolio на основе распределения Коши составляет 26,10%, а на основе распределения Гаусса – 8,80%, т.е. занижается в 3 раза!

В-третьих, в теории вероятностей каждое распределение выводится на основе идеализированного сценария («шары вынимаются из урны… а потом возвращаются в урну…»), который воспроизводится неограниченное число раз; в результате строится функция распределения. В реальной жизни дата-сайентист, как правило, имеет конечную выборку, для которой пытается подобрать функцию распределения из набора возможных, сопоставляя контекст задачи и сценарий построения каждой функции из набора. Рассмотрим два примера (рис. 3.11, 3.12).

На рис. 3.11 по выборке студентов одного потока построена гистограмма их возрастов. Каким распределением лучше аппроксимируется рост – равномерным или нормальным? А если это он-лайн обучение? А если вспомнить, что в РФ нет законодательных ограничений на возраст обучающихся?

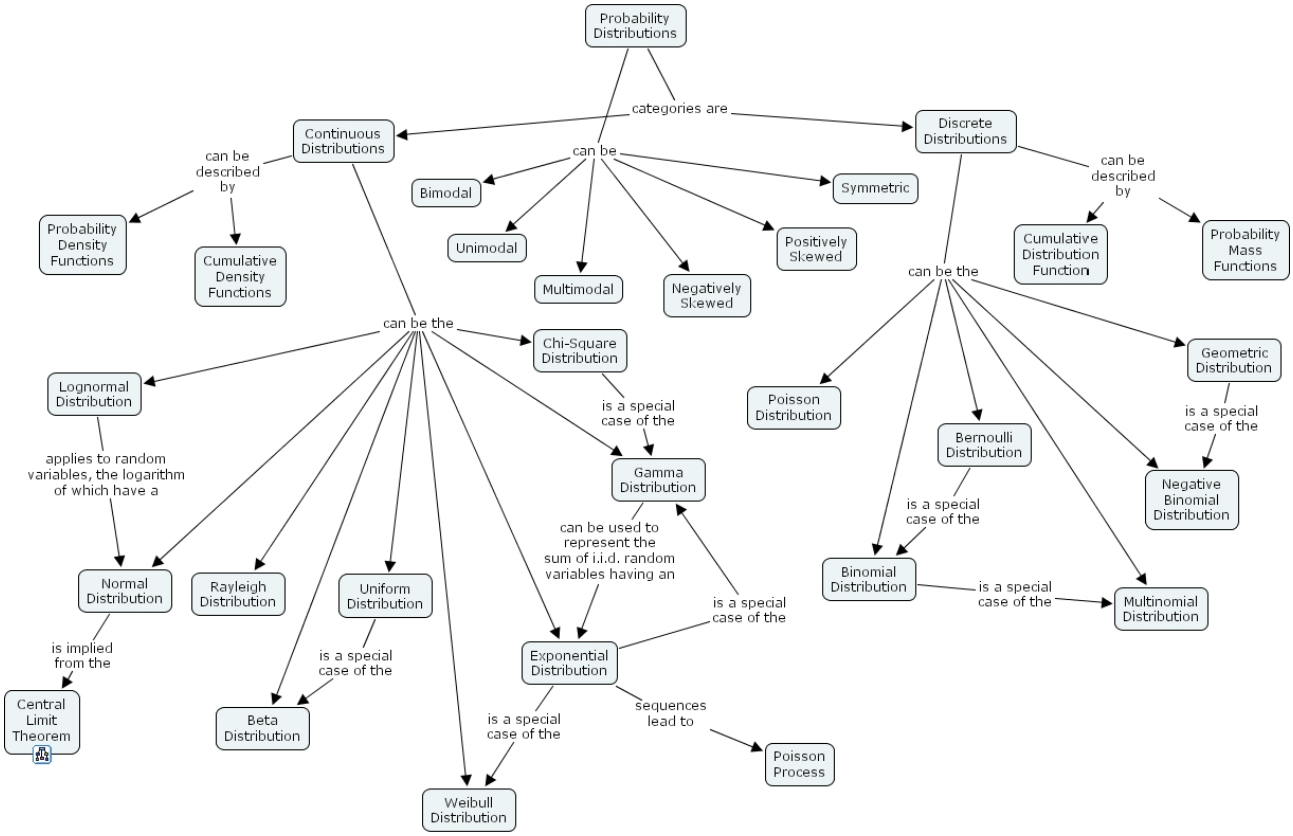


Рис. 3.9. Функции распределения вероятностей и их взаимосвязь (источник [https://skat.ihmc.us/rid=1110240761600_1633886338_5337/Probability_Distributions.cmap])

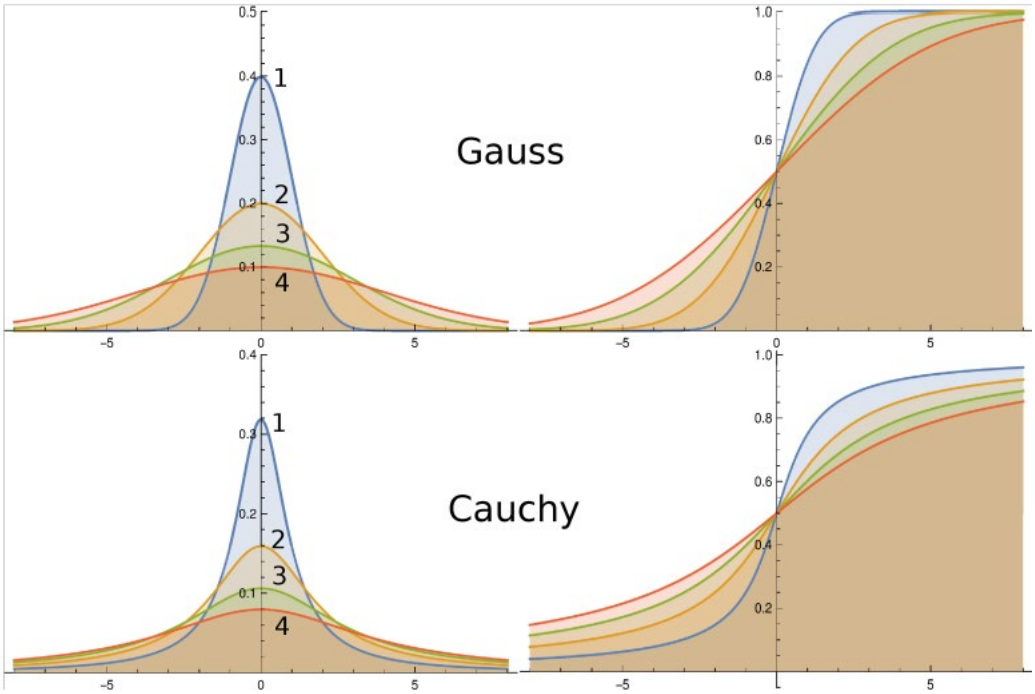


Рис. 3.10. Функции плотности вероятности Гаусса и Коши (левые панели) и распределения вероятности (правые панели). Числа около кривых Гаусса указывают значение дисперсии (σ^2 = 1, 2, 3 или 4) для каждой кривой, числа около кривых Коши указывают значение параметра ширины (γ = 1, 2, 3 или 4) [Sevcik]

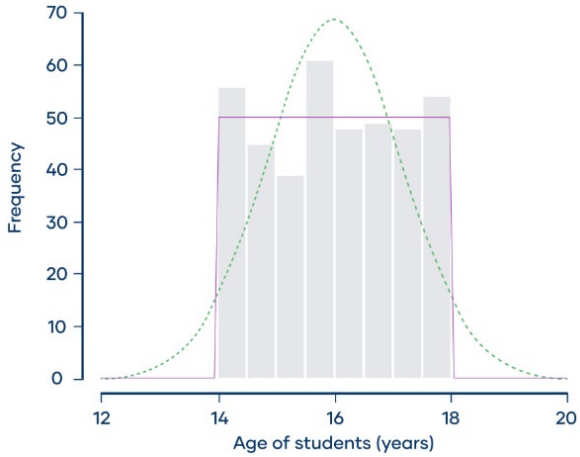


Рис. 3.11. К выбору плотности вероятности по гистограмме (источник [https://www.scribbr.com/statistics/kurtosis/#:~:text=Kurtosis%20is%20a%20measure%20of,(thin%20tails)%20are%20platykurtic])

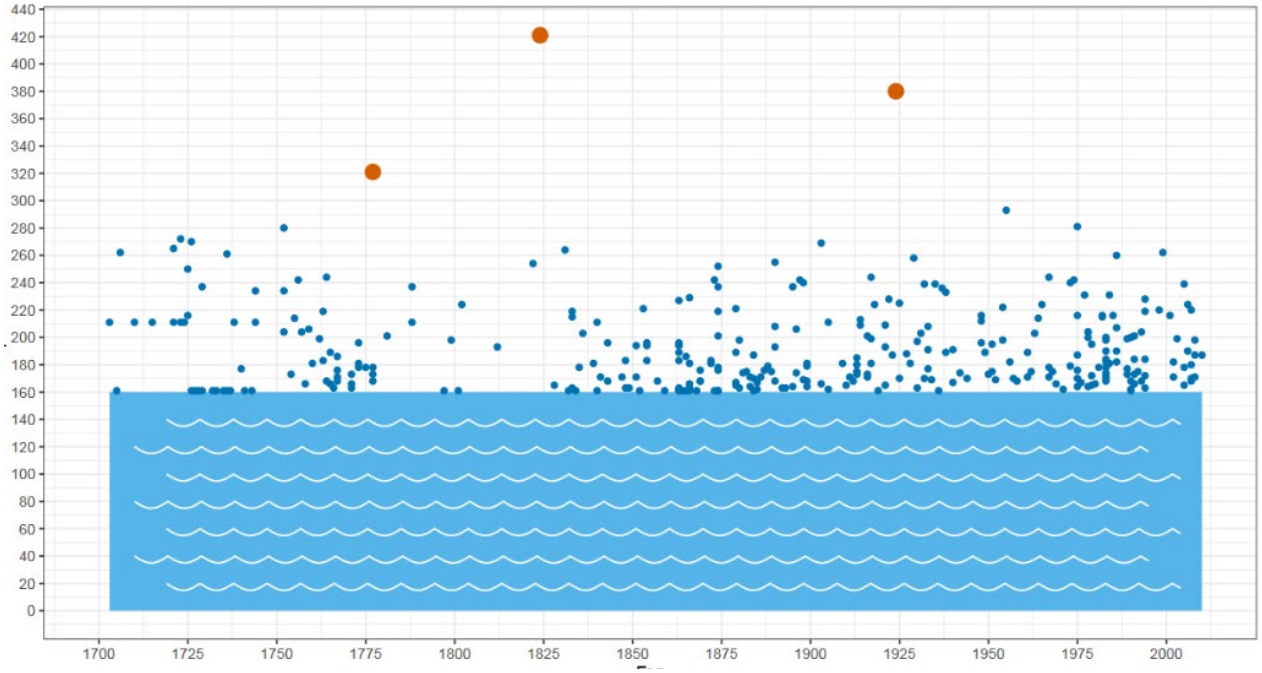


Рис. 3.12. Максимальный уровень воды в Неве (в сантиметрах от ординара) по годам (1703–2011). Источник [http://www.nevariver.ru/flood_list.php]

На рис. 3.12 показан график наводнений в г. Санкт-Петербург от основания города (1703 г.) до окончания строительства защитной дамбы (2011 г.). Если бы Петр I предвидел наводнения 1775 и 1824 гг., стал бы он закладывать город в дельте Невы? Это, конечно, вопрос риторический. Три самых крупных наводнения, выделенные на графике оранжевым цветом, говорят о том, что уровень воды в Неве до 2011 года описывался тяжелохвостым распределением со всеми его свойствами, главное из которых – следующее: каким бы ни был исторический максимум, всегда может случиться наводнение, которое поставит новый рекорд. В связи с этим в городе строили дома на мощных фундаментах, которые устоят во время любого наводнения (теперь дамба решает эту задачу, динамически понижая уровень нагонной волны по мере его подъема).

А теперь перенесемся в 2005 год: предприниматель хочет поставить на берегу Невы легкий торговый павильон, и ему требуется обоснование проекта. Тут можно переформулировать задачу: рассчитать риск, т.е. произведение издержек на строительство и вероятности наводнения такого уровня, которое этот павильон разрушит, а превышение риска заложить в страховые выплаты. В этом случае можно будет ограничиться 99-м квантилем, т.е. игнорировать самые высокие уровни воды, а для формирования выборки использовать интервал между ними (например, 1930–2000 г.), и для него оценить средний уровень подъема воды, на который должен быть рассчитан павильон. Согласятся ли на такую постановку задачи предприниматель и, главное, городские власти?

Таким образом, при выборе функции распределения дата-сайентист должен стремиться адекватно поставить задачу и максимально сузить ее контекст, привлекая заказчиков и экспертов предметной области. Но это не всегда возможно, и на нем остается огромная доля ответственности за неудачные решения.

3.3.2. Типовые распределения в DS

Из огромного числа распределений, существующих в мире чистой математики, в мире DS распространение получили лишь некоторые (рис. 3.13). В связи с ростом интереса к тяжелохвостым процессам к этой схеме можно добавить степенные и альфа-стабильные распределения, которые рассматриваются в следующем параграфе.

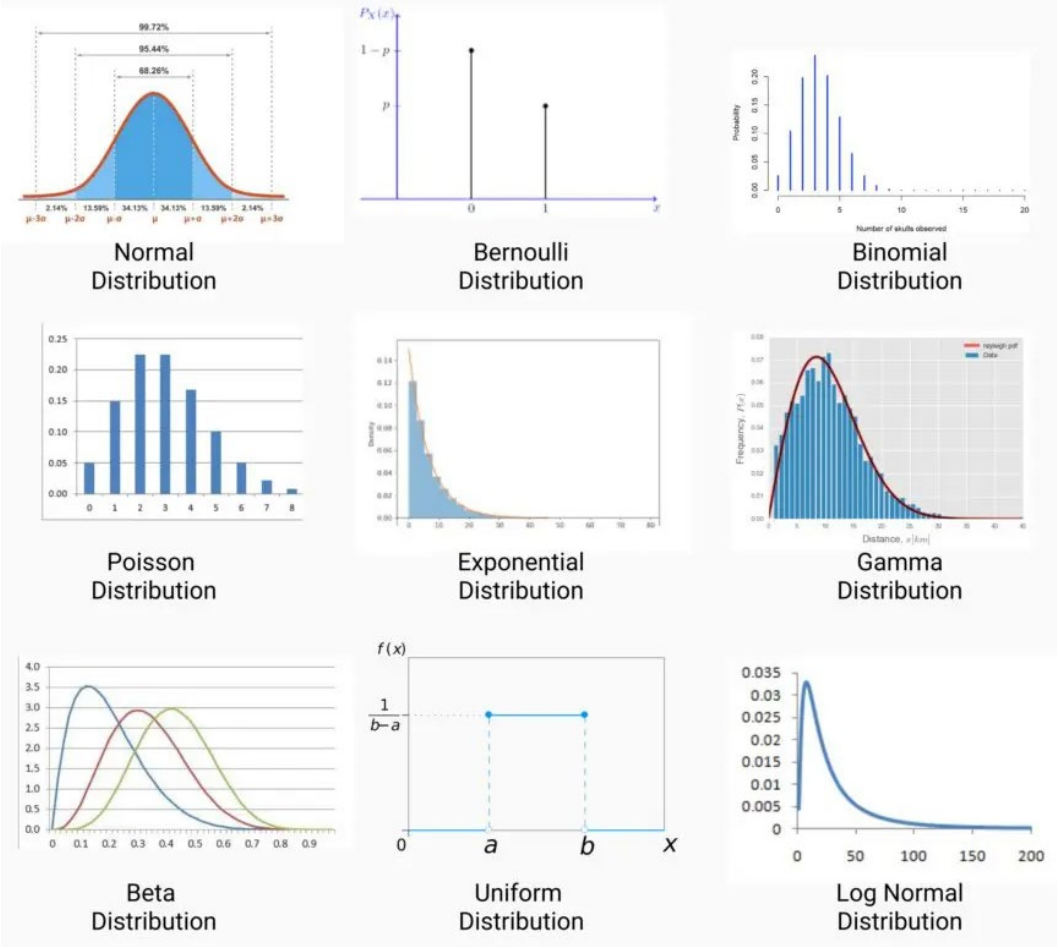


Рис. 3.13. Типовые распределения, используемые в DS (источник [https://datasciencedojo.com/blog/probability-distributions-in-data-science/])

В данном разделе упомянутые распределения описаны по единой схеме. Каждое распределение иллюстрируется примером ее функции плотности распределения (ФПР, probability density function, PDF). Горизонтальная ось каждого графика – набор возможных чисел-исходов x, вертикальная – вероятность каждого исхода $p(x)$. У дискретных распределений исходы являются целыми числами, а сами распределения выглядят как редкие линии, по одной на каждый исход, с высотой, соответствующей вероятности данного исхода. У непрерывных распределений исходы могут принять любое численное значение, и такие распределения представляются кривыми, причем площадь под участком кривой равна вероятности исхода в этой зоне.

К каждому распределению даются идеализированный математический сценарий его формирования, а также (по возможности) примеры адекватного применения в реальном мире.

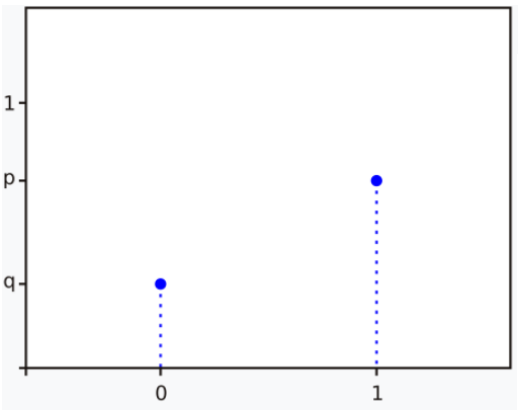
Распределение Бернулли. Случайная величина X имеет распределение Бернулли, если она принимает всего два значения, 1 и 0, с вероятностями p и $q \equiv 1-p$, соответственно. Если оба исхода равновероятны, то ФПР Бернулли содержит две линии одинаковой высоты, представляющие 2 равновероятных исхода, 0 и 1, соответственно. Распределение Бернулли может представлять и неравновероятные исходы, типа броска неправильной монетки (как на рис. 3.14).

Распределение Бернулли используется при контроле качества продукции. Важное условие – испытания должны быть независимыми друг от друга. Для этого, в частности, при проверке партии деталей уже проверенные детали должны возвращаться обратно в проверяемую партию.

Биномиальное распределение с параметрами n и p – распределение количества «успехов» в последовательности из n независимых случайных экспериментов, таких, что вероятность «успеха» в каждом из них постоянна и равна p. Киньте честную монету два раза – сколько раз будет орел? Ответ подчиняется биномиальному распределению. Оно возникает при подсчете количества успехов в однотипных испытаниях (как броски монеты), где каждый бросок не зависит от других и имеет одинаковую вероятность успеха.

Как биномиальное распределение используется в реальном мире [Кодкамп]?

- моделирование побочных эффектов от лекарств – найти вероятность того, что более определенного числа пациентов в случайной выборке из 100 будут испытывать негативные побочные эффекты, если их испытывают 5% больных:
  - P(X > 5 пациентов испытывают побочные эффекты) = 0,38400,
  - P(X > 10 пациентов испытывают побочные эффекты) = 0,01147.
- моделирование мошеннических транзакций – найти вероятность того, что в данный день произойдет более определенного количества мошеннических транзакций, если 2% всех транзакций по кредитным картам в определенном регионе являются мошенническими:
  - P(X > 1 мошенническая транзакция) = 0,26423,
  - P(X > 2 мошеннических транзакций) = 0,07843.

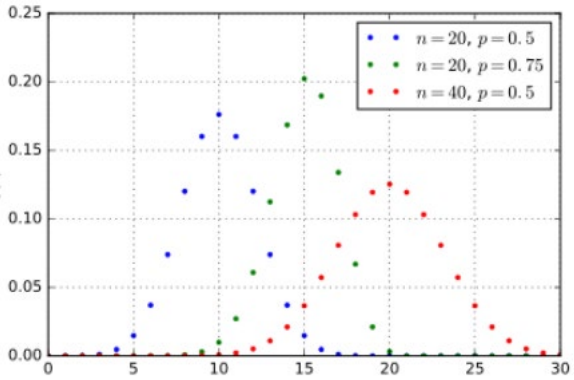


Функция вероятности	q	$k=0$
	p	$k=1$

Математическое ожидание	p
Мода	$\begin{cases} 0, & q>p \\ 0,1, & q=p \\ 1, & q<p \end{cases}$
Дисперсия	pq
Коэффициент асимметрии	$\frac{q-p}{\sqrt{pq}}$
Коэффициент эксцесса	$\frac{6p^2-6p+1}{p(1-p)}$

а б

Рис. 3.14. Распределение Бернулли: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/распределение_Бернулли])



Функция вероятности	$\binom{n}{k} p^k q^{n-k}$
Математическое ожидание	np
Медиана	одно из $\{\lfloor np \rfloor - 1, \lfloor np \rfloor, \lfloor np \rfloor + 1\}$
Мода	$\lfloor (n+1)\,p \rfloor$
Дисперсия	npq
Коэффициент асимметрии	$\frac{q-p}{\sqrt{npq}}$
Коэффициент эксцесса	$\frac{1-6pq}{npq}$

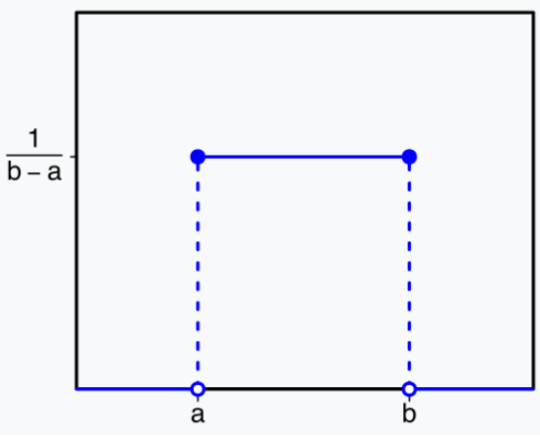
а б

Рис. 3.15. Биномиальное распределение: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Биномиальное_распределение])

Равномерное распределение – распределение случайной вещественной величины, принимающей значения, принадлежащие некоторому промежутку конечной длины, характеризующееся тем, что плотность вероятности на этом промежутке почти всюду постоянна. Оно может быть непрерывным (как на рис. 3.15) или дискретным – тогда оно описывает вероятность выпадения какой-то определенной грани «правильного кубика».

Где можно встретить равномерное распределение?

– Если бы вы подошли к случайному человеку на улице, вероятность того, что его день рождения приходится на определенную дату, будет иметь равномерное распределение.
– Если вы случайно выбираете карту из колоды, то вероятность того, что карта будет пиковой, червовой, трефовой или бубновой, распределяется равномерно, потому что каждая масть будет выбрана с одинаковой вероятностью.



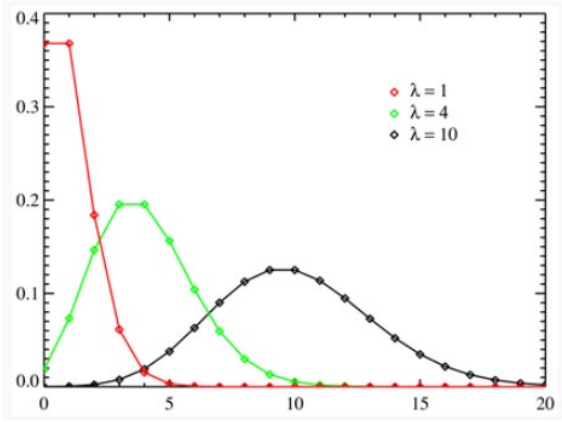
Плотность вероятности	$\frac{1}{b-a}$ $a \leqslant x \leqslant b$
	1 $x \geqslant b$
Математическое ожидание	$\frac{a+b}{2}$
Медиана	$\frac{a+b}{2}$
Мода	любое число из отрезка $[a, b]$
Дисперсия	$\frac{(b-a)^2}{12}$
Коэффициент асимметрии	0
Коэффициент эксцесса	$-\frac{6}{5}$

а б

Рис. 3.16. Равномерное распределение: ФПР (а) и числовые характеристики (б) [https://ru.wikipedia.org/wiki/Непрерывное_равномерное_распределение]

Распределение Пуассона представляет собой число событий, произошедших за фиксированное время, при условии, что данные события происходят с некоторой фиксированной средней интенсивностью и независимо друг от друга.

Функция вероятности	$\frac{e^{-\lambda}\lambda^k}{k!}$



Математическое ожидание	λ
Медиана	$\approx \lfloor \lambda + 1/3 - 0.02/\lambda \rfloor$
Мода	$\lfloor \lambda \rfloor, \lfloor \lambda \rfloor - 1$
Дисперсия	λ
Коэффициент эксцесса	λ^{-1}

а б

Рис. 3.16. Распределение Пуассона: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Распределение_Пуассона])

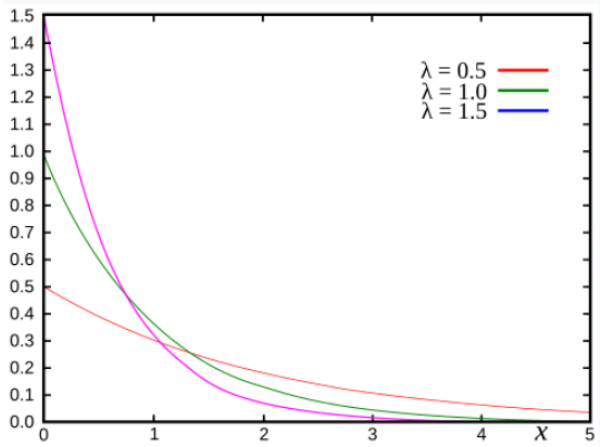
Так же, как и биномиальное, распределение Пуассона – это распределение количества раз того, как что-то произойдёт. Но, в отличие от биномиального, распределение Пуассона может описывать не только последовательные, но и накладывающиеся события – например, если в колл-центр звонят одновременно несколько клиентов. Поэтому оно параметризуется не вероятностью p и количеством испытаний n, но средней интенсивностью $\lambda=np$.

Распределение Пуассона описывает самые разнообразные потоки событий, например:

– количество ожидаемых клиентов – найти вероятность того, что в ресторан придут больше, чем определенное количество клиентов, если в среднем в день приходят 100 клиентов:
  - P(X > 110 клиентов) = 0,14714,
  - P(X > 120 клиентов) = 0,02267.
– количество ожидаемых сетевых сбоев – найти вероятность того, что компания столкнется с определенным количеством сбоев в сети за данную неделю, если в среднем происходит 1 сбой сети в неделю:
  - P(X = 0 отказов) = 0,36788,
  - P(X = 1 отказ) = 0,36788,
  - P(X = 2 отказа) = 0,18394.

Экспоненциальное распределение моделирует время между двумя последовательными свершениями одного и того же события.

Оно тесно связано с распределением Пуассона: если есть события, количество которых на единицу времени подчиняется распределению Пуассона, то время между ними подчиняется экспоненциальному распределению с тем же параметром λ.



Плотность вероятности	$\lambda e^{-\lambda x}$
Математическое ожидание	λ^{-1}
Медиана	$\ln(2)/\lambda$
Мода	0
Дисперсия	λ^{-2}
Коэффициент асимметрии	2
Коэффициент эксцесса	6

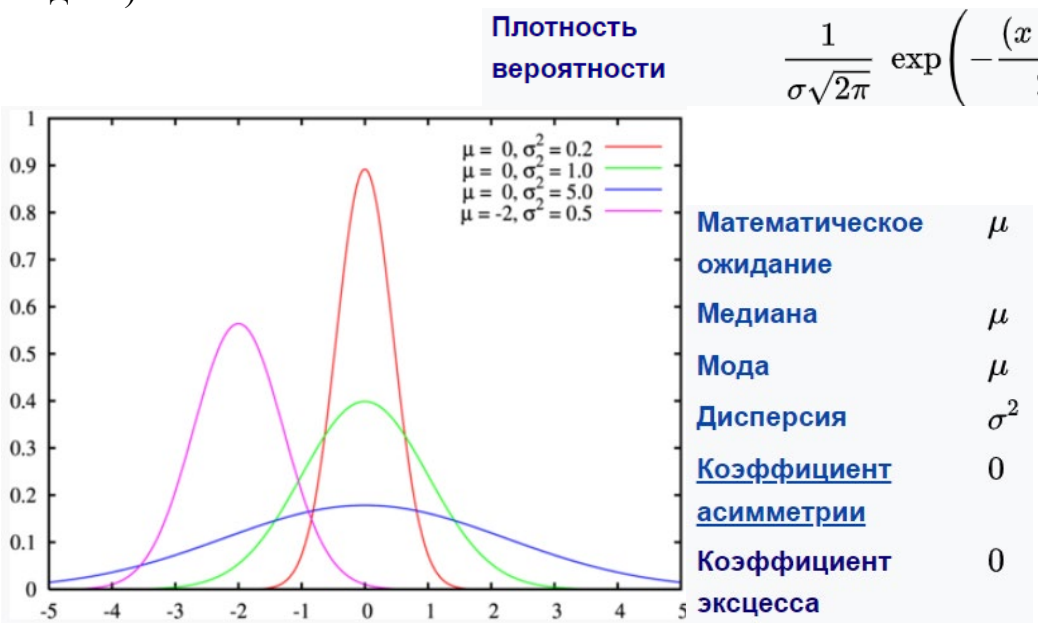
а б

Рис. 3.17. Экспоненциальное распределение: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Экспоненциальное_распределение])

Главное техническое применение экспоненциального распределения – оценка «наработки на отказ». Другие применения:
– время между землетрясениями – найти вероятность того, что следующее землетрясение произойдет не ранее, чем через 500 дней, если землетрясение в данном регионе происходит в среднем каждые 400 дней:
  - P(X ≤ 500 дней) = 0,7135,
  - P(X > 500 дней) = 1 – 0,7135 = 0,2865.
– время между вызовами в колл-центре – найти вероятность того, что следующий клиент позвонит в промежутке 10–15 мин., если клиенты звонят в среднем каждые 10 мин.:
  - P(10 < X ≤ 15) = (1 – exp(–0,1×15)) – (1 – exp(–0,1×10)) = 0,1448.

Нормальное распределение. Если величина является суммой (i) многих (ii) случайных (iii) слабо взаимозависимых величин, (iv) каждая из которых вносит малый вклад относительно общей суммы, то центрированное и нормированное распределение такой величины при достаточно большом числе слагаемых стремится к нормальному распределению.

Если процесс формирования случайной величины удовлетворяет четырем вышеперечисленным условиям, то для нее справедлива центральная предельная теорема [https://ru.wikipedia.org/wiki/Центральная_предельная_теорема], и сама величина распределена по нормальному закону. В качестве дополнительного условия можно добавить необходимость сопоставимого масштаба суммируемых распределений: если одно существенно доминирует надо остальными, то сходиться будет медленно (но с этим легко справиться небольшими преобразованиями задачи).



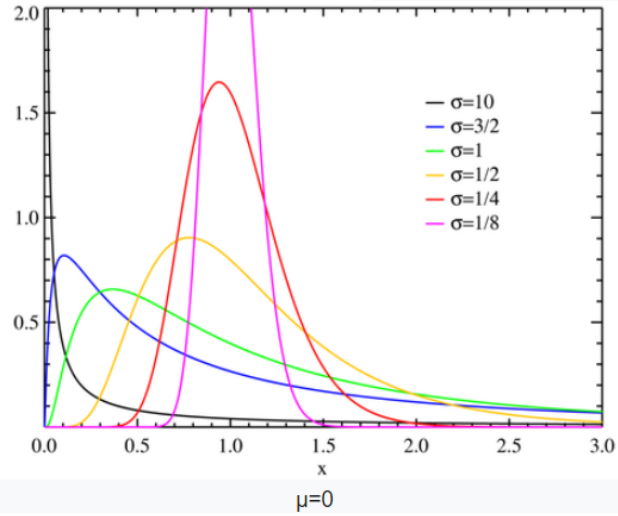
а б

Рис. 3.18. Нормальное распределение: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Нормальное_распределение])

Посмотрите на рис. 3.18, б – какое удобное во всех смыслах распределение: симметричное; все высшие моменты равны нулю; сумма нормально распределенных случайных величин остается нормально распределенной случайной величиной. Более того, нормальное распределение – предельный случай биномиального (с увеличением количества испытаний) и Пуассона (с увеличением параметра интенсивности λ).

Понятно желание дата-сайентиста использовать именно нормальное распределение в практических задачах. Но желание нужно сопоставлять с возможностями – а для этого надо содержательно проверять, выполняются ли в контексте задачи четыре вышеприведенных условия (позитивный пример – рис. 3.11, а негативный – рис. 3.12). Есть ряд формальных способов нормализации распределений, т.е. приведения их к нормальному виду (см. раздел 4), но к ним надо относиться с осторожностью.

Логнормальное распределение. Если случайная величина имеет логнормальное распределение, то ее логарифм имеет нормальное распределение. Из центральной предельной теоремы следует, что при определенных условиях логнормальное распределение является предельным распределением для произведения независимых положительных случайных величин.



Плотность вероятности	$\exp\left(-\left[\frac{\ln(x)-\mu}{\sigma}\right]^2 \Big/ 2\right) / (x\sigma\sqrt{2\pi})$
Математическое ожидание	$e^{\mu+\sigma^2/2}$
Медиана	e^{μ}
Мода	$e^{\mu-\sigma^2}$
Дисперсия	$(e^{\sigma^2}-1)e^{2\mu+\sigma^2}$
Коэффициент асимметрии	$(e^{\sigma^2}+2)\sqrt{e^{\sigma^2}-1}$
Коэффициент эксцесса	$e^{4\sigma^2}+2e^{3\sigma^2}+3e^{2\sigma^2}-6$

а б

Рис. 3.19. Логнормальное распределение: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Логнормальное_распределение])

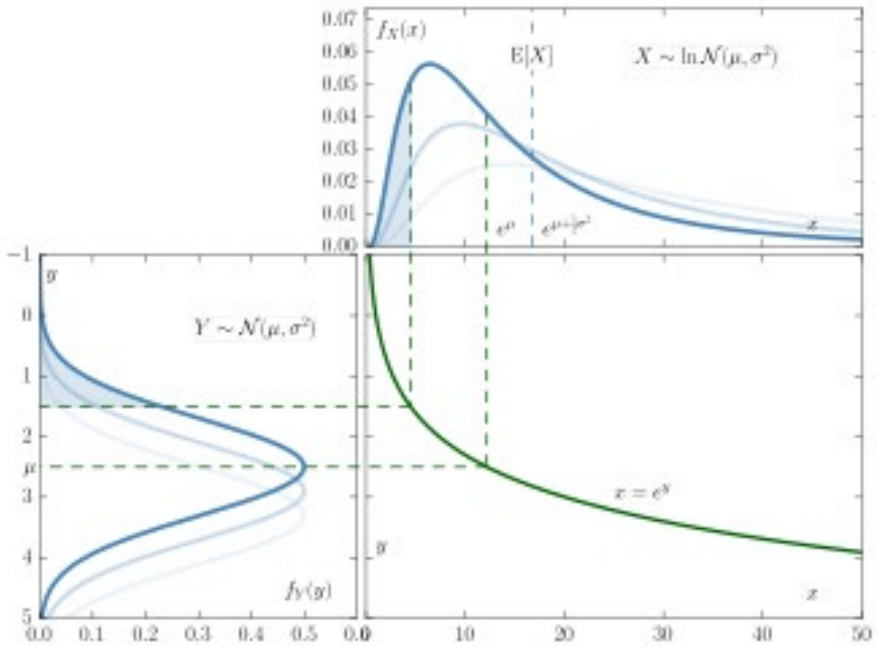


Рис. 3.20. Соотношение между нормальным и логнормальным распределениями [https://en.wikipedia.org/wiki/Log-normal_distribution#Generation_and_parameters]

Соотношение между нормальным и логнормальным распределениями показано на рис. 3.20: если случайная величина $Y=\mu+\sigma Z$ распределена нормально, то $X\sim\exp(Y)$ распределена логнормально. И наоборот: при логарифмическом

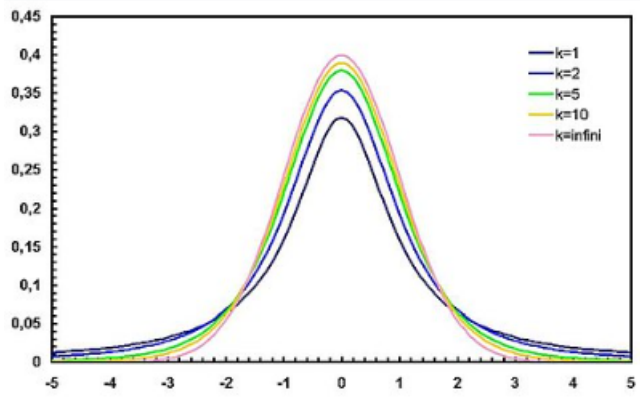
преобразовании переменной X (Y=ln(X)) мы получаем переменную Y, которая распределена нормально. Таким образом, эффективный способ анализа логарифмически нормально распределенных данных состоит в применении известных методов, основанных на нормальном распределении, к логарифмически преобразованным данным, а затем в обратном преобразовании результатов.

Логнормальное распределение часто возникает в природе. Например, в медицине оно описывает инкубационный период заболевания, в геологии – концентрацию редких элементов в горных породах, в лингвистике – количество слов в предложениях. Поскольку логнормальное распределение ограничено нулем с нижней стороны, в финансах оно подходит для моделирования цен активов, которые не могут принимать отрицательные значения. Нормальное распределение не может быть использовано для этой цели, поскольку оно симметрично.

Распределение Стьюдента – это непрерывное одномерное распределение с одним параметром n – количеством степеней свободы. Если Y_0, Y_1, …, Y_n – независимые стандартные нормальные случайные величины, такие что $Y_i \sim \mathcal{N}(0,1)$, i=0,1, …, n, то распределение случайной величины t,

$$t = \frac{Y_0}{\sqrt{\frac{1}{n}\sum_{i=1}^{n} Y_i^2}},$$

называется распределением Стьюдента с n степенями свободы (рис. 3.21). Оно является выборочным аналогом нормального распределения и при большом числе наблюдений, $n \to \infty$, практически переходит в нормальное.



а

Плотность вероятности	$\frac{\Gamma\left(\frac{n+1}{2}\right)\left(1+\frac{x^2}{n}\right)^{-\frac{n+1}{2}}}{\sqrt{n\pi}\,\Gamma\left(\frac{n}{2}\right)}$
Математическое ожидание	0, если $n > 1$
Медиана	0
Мода	0
Дисперсия	$\frac{n}{n-2}$, если $n > 2$
Коэффициент асимметрии	0, если $n > 3$
Коэффициент эксцесса	$\frac{6}{n-4}$, если $n > 4$

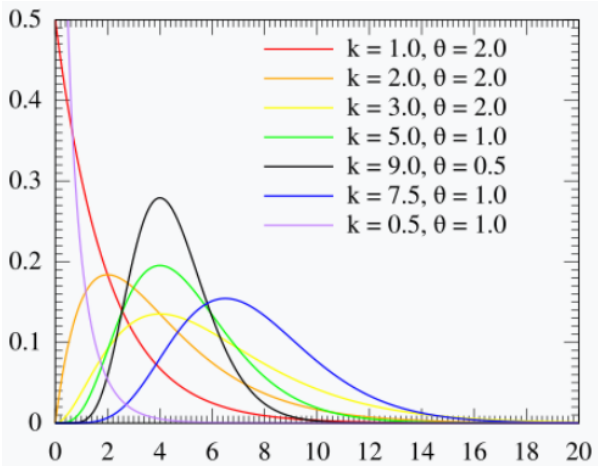
б

Рис. 3.21. Распределение Стьюдента: ФПР (а) и числовые характеристики (б); Г – гамма-функция Эйлера (источник [https://ru.wikipedia.org/wiki/Распределение_Стьюдента])

График функции плотности распределения Стьюдента симметричен, а его форма напоминает форму колокола, как у стандартного нормального распределения, но он ниже и шире (с более «тяжелыми» хвостами). По мере возрастания n кривая функции плотности все больше напоминает стандартное нормальное распределение.

Распределение Стьюдента – одно из наиболее известных распределений среди используемых при анализе реальных данных. Оно возникает в задачах проверки гипотез о среднем значении нормального распределения в случае неизвестной дисперсии. Его применяют при оценивании математического ожидания, прогнозного значения и других характеристик с помощью доверительных интервалов, по проверке гипотез о значениях математических ожиданий (гипотеза о неизвестном среднем статистической выборки из нормального распределения), коэффициентов регрессионной зависимости, гипотез однородности выборок и т.д.

Гамма-распределение, как и экспоненциальное, моделирует время между двумя последовательными свершениями одного и того же события, но позволяет делать это гораздо более гибко [Gamma]. Неформально говоря, гамма-распределение – это распределение суммы экспоненциально распределенных величин. Экспоненциальное распределение – частный случай гамма-распределения при k =1 (см. рис. 3.22).



а

Плотность вероятности	$\frac{1}{\Gamma(k)\theta^k} x^{k-1} e^{-\frac{x}{\theta}}$
Математическое ожидание	$k\theta$
Медиана	Отсутствует явное выражение в замкнутой форме
Мода	$(k-1)\theta$ при $k \geq 1$
Дисперсия	$k\theta^2$
Коэффициент асимметрии	$\frac{2}{\sqrt{k}}$
Коэффициент эксцесса	$\frac{6}{k}$

б

Рис. 3.22. Гамма-распределение: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Гамма-распределение])

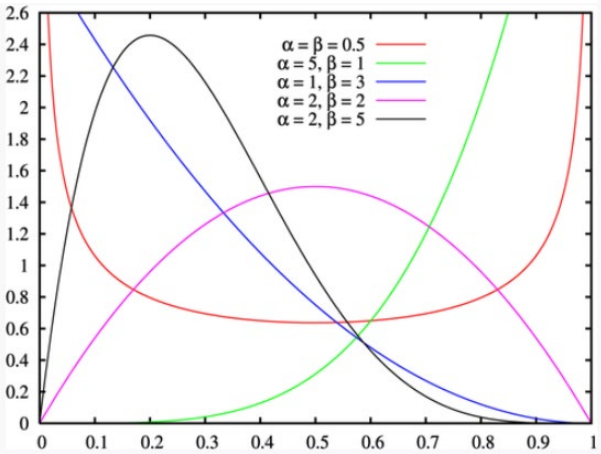
Например, экспоненциальное распределение моделирует поведение сущностей, которые выходят из строя с постоянной скоростью λ, независимо от накопленного «возраста». Хотя это свойство значительно упрощает анализ, оно оказывается слишком грубым в большинстве реальных приложений. Напротив, гамма-распределение, в отличие от экспоненциального с единственным параметром λ, имеет два настроечных параметра – параметр формы k и параметр масштаба θ – и поэтому обладает большей гибкостью описания.

Примеры применения гамма-распределения:

– в финансах – для оценки вероятности определенной нормы прибыли на инвестиции, что имеет важное значение в инвестиционных стратегиях и управлении портфелем, а также для моделирования волатильности базового актива в ценообразовании опционов,

– в медицине – для моделирования времени между обращениями пациентов в больницу, для моделирования времени ожидания, пока лекарство достигнет своего максимального эффекта в организме, для оценки вероятности выживания пациента по истечении определенного времени,
– в инженерии – для моделирования времени между отказами системы («наработки на отказ»), а также для моделирования прочности материалов (вероятности разрушения материала при определенных условиях),
– в экономике – для моделирования распределения доходов населения, а также продолжительности безработицы (оценки вероятности того, что человек найдет работу по истечении определенного времени).

Бета-распределение используется для описания случайных величин, значения которых ограничены конечным интервалом (как правило, [0, 1]). Форма графика плотности вероятности бета-распределения зависит от выбора параметров α и β и изменяется от строго выпуклого и уходящего в бесконечность на границах (красная кривая) до строго убывающего (синяя кривая). При α=1, β=1 график совпадает с графиком плотности стандартного непрерывного равномерного распределения (см. рис. 3.23).



а

Плотность вероятности	$\frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)}$
Математическое ожидание	$\frac{\alpha}{\alpha+\beta}$
Мода	$\frac{\alpha-1}{\alpha+\beta-2}$ для $\alpha>1,\beta>1$
Дисперсия	$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$
Коэффициент асимметрии	$\frac{2(\beta-\alpha)\sqrt{\alpha+\beta+1}}{(\alpha+\beta+2)\sqrt{\alpha\beta}}$
Коэффициент эксцесса	$6\frac{\alpha^3-\alpha^2(2\beta-1)+\beta^2(\beta+1)-2\alpha\beta(\beta+2)}{\alpha\beta(\alpha+\beta+2)(\alpha+\beta+3)}$

б

Рис. 3.23. Бета-распределение: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Бета-распределение])

Интуитивно [Averianova], бета-распределение – это распределение вероятностей по вероятностям, поэтому у него такая специальная область определения [0, 1]: ось *x* – это как раз вероятности. Содержательно α–1 связывается с количеством успешных исходов, а β–1 – с количеством неудач. В общем случае параметры α и β могут быть какими угодно, что и обеспечивает гибкость бета-распределения при моделировании.

Пример. Насколько вероятно, что кто-то согласится (т.е. вероятность успеха больше 50%) пойти с вами на свидание, если ваша ситуация соответствует бета-распределению с α = 2 и β = 8?

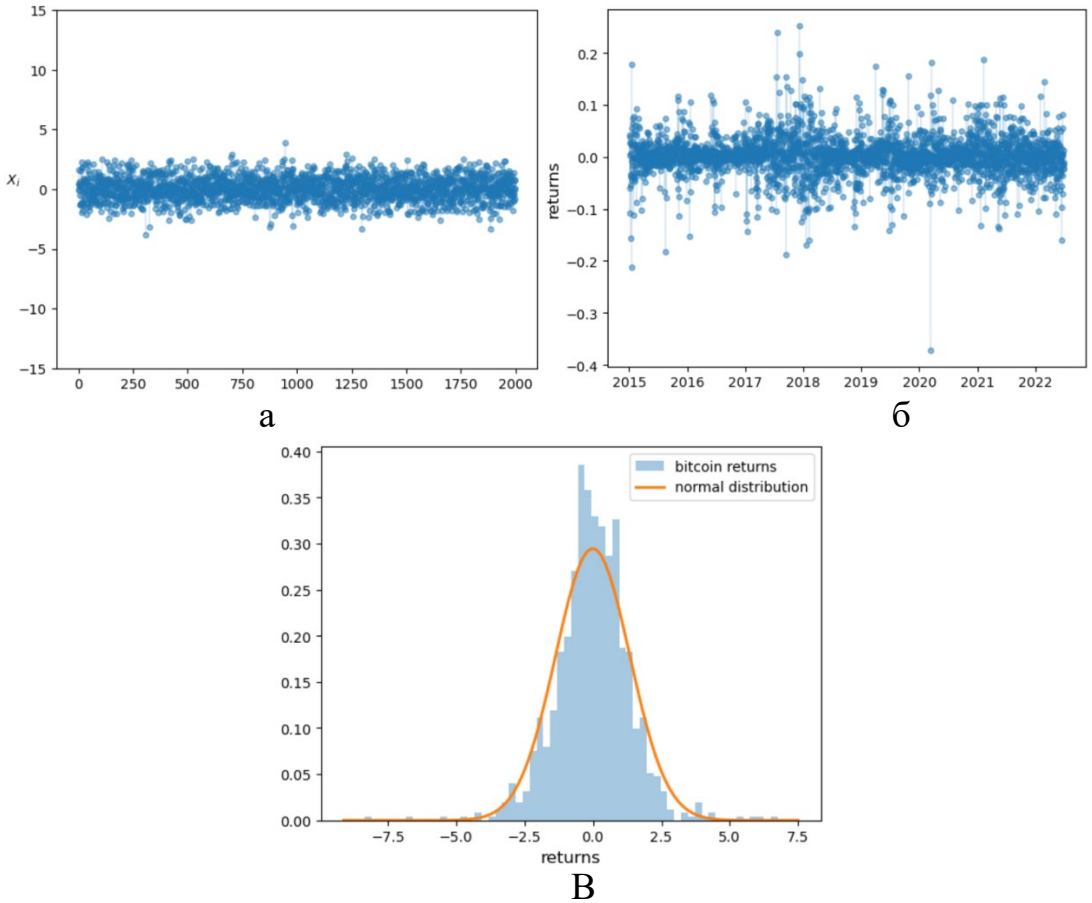
Ответ: $P(X>0.5)$ = 1– функция распределения(0.5) = 0.01953 – очень маленькая! Надо изменить контекст!

Бета-распределение можно использовать для моделирования вероятностей: рейтинг кликов вашей рекламы, коэффициент конверсии клиентов, фактически купивших что-то на вашем сайте, насколько вероятно, что посетители поставят лайки в вашем блоге, насколько вероятно избрание Трампа на очередной срок, 5-летний прогноз выживания женщин с раком груди и т.д.

Еще одно достоинство бета-распределения – оно широко применяется в расчетах на основе байесовской статистики.

3.3.3. Распределения с тяжелыми хвостами в DS

Что такое хвост распределения и почему он так важен в DS? Сравним два примера [Sargent].



а б

В

Рис. 3.24. Гистограммы роста людей (а) и доходов от биткоина (б), сопоставление функций распределения (в)

На рис. 3.24, а, показана гистограмма роста людей, нормированного на средний рост, на рис. 3.24, б, – динамика доходов по биткоину. Во втором случае присутствуют редкие, но значительные по размеру экстремальные наблюдения. Это отражается на функциях распределения (рис. 3.24, в): появляются всплески значений далеко за пределами интервала, характерного для нормального распределения.

Исследования последних лет [Талеб] показали, что случайные величины делятся на две группы:

– обычные – имеют некоторое типичное значение, от которого далеко не отклоняются; им соответствуют распределения с легким хвостом;

– необычные – в основном живут около типичного значения, но очень редко и при этом очень сильно отклоняются от него; им соответствуют распределения с тяжелым хвостом (fat- or heavy-tailed distributions).

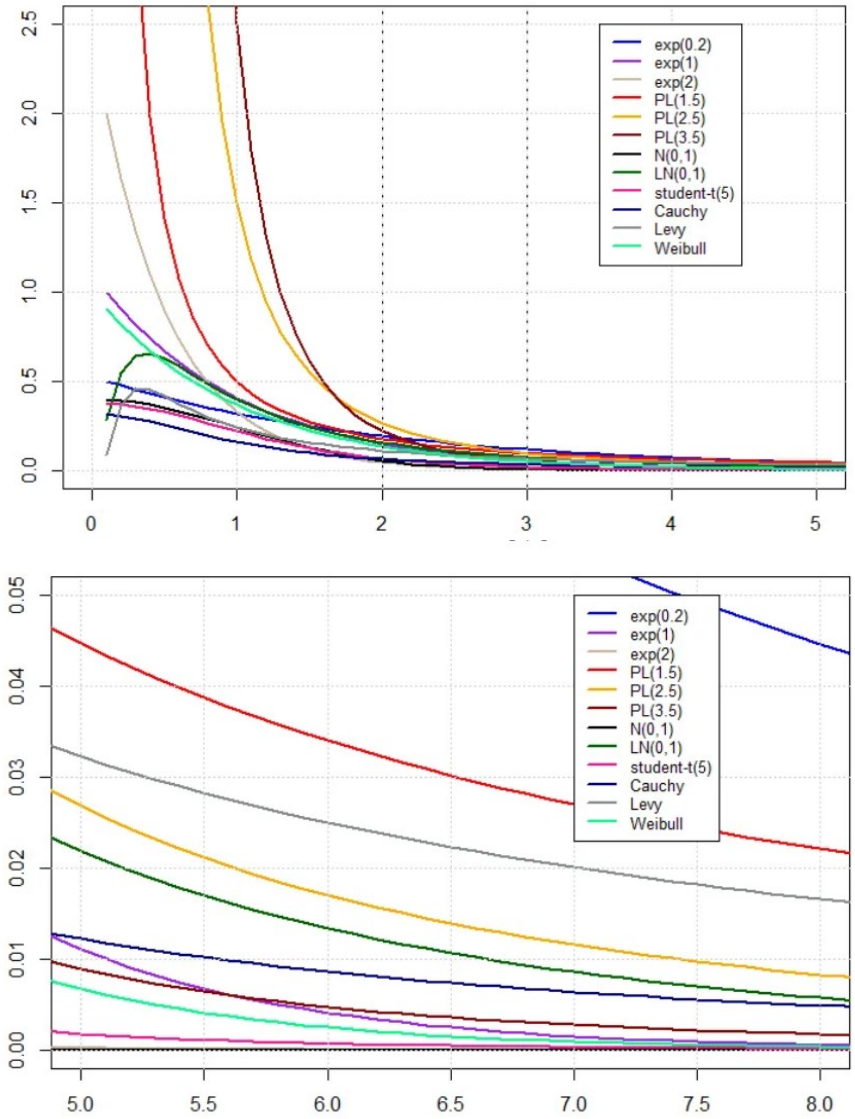


Рис. 3.25. Соотношение тяжелохвостых и легкохвостых распределений вблизи нуля (а) и на хвосте распределений (б). Обозначения: Exponential distribution (exp), Power-law distribution (PL), Normal distribution (N), Log-Normal distribution (LN), Student-t distribution, Cauchy distribution, Levy distribution, Weibull distribution (источник [https://towardsdatascience.com/journey-to-tempered-stable-distribution-part-1-fat-tailed-distribution-958d28bc20c])

Как работать в процессе, если он имеет такую непредсказуемость? Это – вопрос, на который важно и даже страшно отвечать. Например, в финансовой аналитике вплоть до 2010-х гг. его старались игнорировать (вспомним пример с оценками риска инвестиционного портфеля из параграфа 3.3.1).

Математическим водоразделом между light- и heavy-tailed distributions принято считать экспоненциальное распределение: если функция распределения убывает быстрее экспоненциальной, то она – легкохвостая, если медленнее – тяжелохвостая (рис. 3.25).

Степенной закон. Степенной закон выражает степенную зависимость между переменными:

$$f(x)=ax^{-k}.$$

Очевидно, что в двойной логарифмической шкале (log–log) он выглядит как прямая линия (однако обратное неверно: многие распределения в log–log шкале выглядят как прямые, но не подчиняются степенному закону).

Как показывают исследования, многие физические, биологические и искусственные явления имеют функции распределения, приблизительно соответствующие степенному закону в различных масштабах. Ограничимся тремя примерами.

Частота употребления слов в большинстве языков убывает по степенному закону (рис. 3.26): если все слова языка или просто достаточно длинного текста упорядочить по убыванию частотности их использования (по рангу), то второе по используемости слово встречается примерно в два раза реже, чем первое, третье – в три раза реже, чем первое, и т.д. Это явление называется законом Ципфа и математически описывается распределением Парето, которое является частным случаем степенного закона.

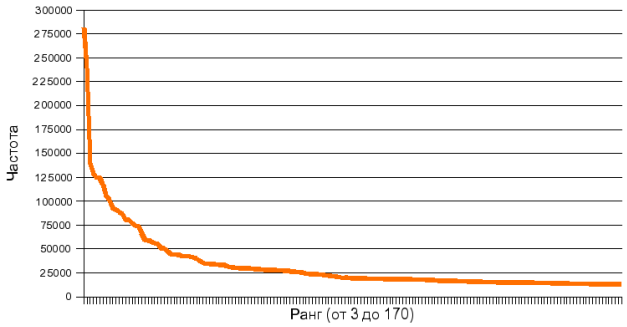


Рис. 3.26. Частотность слов из статей русской Википедии с рангами от 3 до 170 (источник [https://ru.wikipedia.org/wiki/Закон_Ципфа])

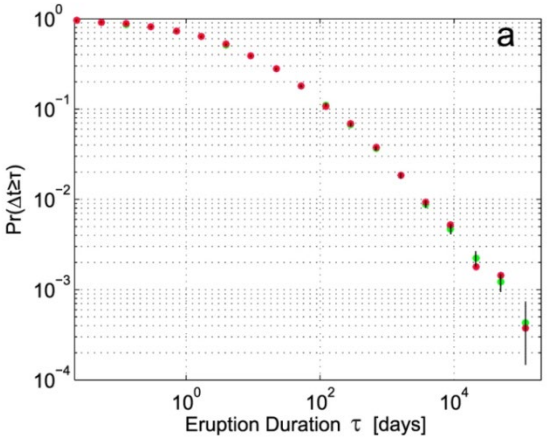


Рис. 3.27. Распределение длительности извержений вулканов

Авторы [Cannavò] на основе анализа большинства известных вулканических извержений показали, что динамика продолжительности извержений соответствует распределению (рис. 3.27). Правая часть графика, построенного в

двойной логарифмической шкале, отлично описывается прямой линией, т.е. налицо степенной закон.

Г. Фехнер в 1860 г. сформулировал «основной психофизический закон» восприятия, устанавливающий логарифмическую зависимость между интенсивностью раздражителя и величиной субъективного ощущения:

$$S = K \ln J + C,$$

где S – субъективная величина ощущения, J – величина (интенсивность) раздражителя (стимула), K и C – константы. Логарифмическая кривая ограничена снизу порогом чувствительности, а сверху – болевым порогом органа чувств. Закон справедлив для любых раздражителей – звука, света, температуры, вкусовых ощущений и тому подобного.

Только в 2017 году было найдено нейрофизиологическое объяснение этого закона. В работе [Scheler] показано, что распределение возбудимости нейронов, которые передают первичный сигнал от органа чувств к зоне принятия решения в мозге, имеет тяжелый хвост, точнее, имеет логнормальную форму. Другими словами, нейроны мозга логарифмируют все сигналы от органов чувств, что позволяет человеку существовать в мире с колоссальными динамическими диапазонами внешних раздражителей (например, перепад освещенности от Солнца в безлунную ночь и в яркий день составляет от 0,001–0,002 лк до 100 000 лк, т.е. 9 порядков).

Механизмы появления степенного закона весьма разнообразны [Farmer, Hilbert], но в целом они свидетельствуют о неравновесных и нелинейных явлениях, имеющих место в анализируемом процессе (например, в бизнесе).

Распределение Парето. Распределение Парето является частным случаем степенного закона. Оно имеет два параметра – параметр формы x_m и параметр масштаба α. Его функция распределения (Cumulative distribution function, CDF) соответствует степенному закону:

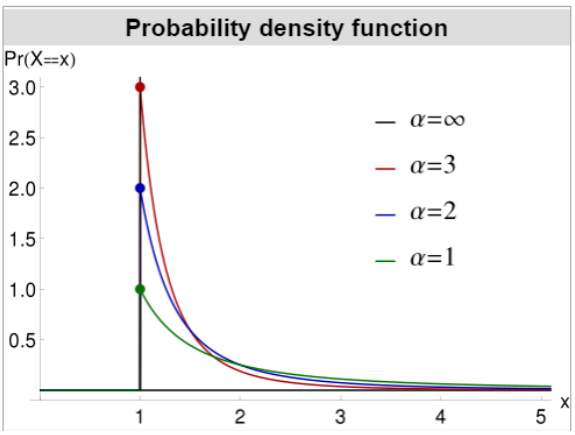
$$CDF = 1 - \left(\frac{x_m}{x}\right)^{\alpha},$$

а вытекающие из него «особенные» свойства распределения показаны на рисунке 3.28.

Свойства распределения Парето:

- при $\alpha \to \infty$ распределение сужается до δ-функции Дирака $\delta(x - x_m)$;
- при малых α моменты распределения не существуют, а следовательно, не существуют и соответствующие числовые характеристики: для $\alpha \le 2$ дисперсия$\to\infty$, для $\alpha \le 2$ коэффициент асимметрии$\to\infty$, для $\alpha \le 2$ эксцесс$\to\infty$. С точки зрения теории это означает, что расходятся соответствующие интегралы. С точки зрения практики это означает, что при вычислении выборочных статистик не выполняется закон больших чисел: с ростом выборки значение числовой характеристики (например, дисперсии) не устанавливается на каком-то уровне, а продолжает сильно колебаться или далеко отходит от ожидаемого теоретического значения.

Заметим, что такое поведение выборочных статистик должно насторожить дата-сайентиста – видимо, анализируемый процесс начинает отклоняться от предсказуемого хода.



а

PDF	$\frac{\alpha x_{\mathrm{m}}^{\alpha}}{x^{\alpha+1}}$
Mean	$\begin{cases} \infty & \text{for } \alpha \le 1 \\ \frac{\alpha x_{\mathrm{m}}}{\alpha - 1} & \text{for } \alpha > 1 \end{cases}$
Median	$x_{\mathrm{m}} \sqrt[\alpha]{2}$
Mode	x_{m}
Variance	$\begin{cases} \infty & \text{for } \alpha \le 2 \\ \frac{x_{\mathrm{m}}^2 \alpha}{(\alpha-1)^2(\alpha-2)} & \text{for } \alpha > 2 \end{cases}$
Skewness	$\frac{2(1+\alpha)}{\alpha-3}\sqrt{\frac{\alpha-2}{\alpha}}$ for $\alpha > 3$
Excess kurtosis	$\frac{6(\alpha^3+\alpha^2-6\alpha-2)}{\alpha(\alpha-3)(\alpha-4)}$ for $\alpha > 4$

б

Рис. 3.28. Распределение Парето: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Распределение_Парето])

Несмотря на столь странные свойства, дата-сайентисту приходится постоянно работать с распределением Парето, так как ему подчиняются многие явления [Reed], например:

– размеры населенных пунктов (мало крупных городов, много деревень/поселков);
– размеры файлов интернет-трафика, использующего протокол TCP (много мелких файлов, несколько больших);
– частота ошибок жесткого диска (много мелких проблем и мало существенных сбоев);
– запасы нефти на нефтяных месторождениях (несколько крупных месторождений и много мелких).

Распределение Коши. Альфа-стабильные распределения. Еще хуже «ведет себя» уже упомянутое в параграфе 3.3.1 распределение Коши (рис. 3.29).

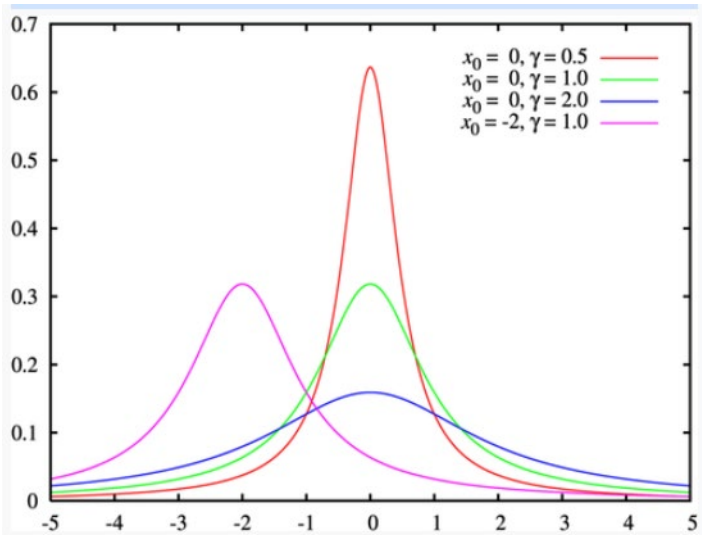
И математическое ожидание, и дисперсия распределения Коши не определены. Это очень хорошо видно при построении выборочных оценок (рис. 3.30). Типичная траектория выборочных средних (рис. 3.30, а) выглядит как длительные периоды медленной сходимости к нулю, прерываемые большими скачками от нуля, но никогда не уходящие слишком далеко. Типичная траектория выборочных дисперсий (рис. 3.30, б) выглядит похоже, но скачки накапливаются быстрее, чем спад, расходясь до бесконечности.

С другой стороны, распределение Коши имеет важное свойство – оно принадлежит к классу альфа-стабильных.

Стабильные распределения удовлетворяют обобщенной центральной предельной теореме, т.е. с точностью до масштаба и сдвига сохраняют свой вид

при суммировании. Свойство сохранять вид функции распределения и называется стабильностью, или устойчивостью. Класс стабильных распределений описывается четырьмя параметрами, два из которых – сдвиг и масштаб, как в нормальном распределении, третий – параметр асимметрии, а четвертый параметр α, $0<\alpha\leq 2$, характеризует степень «тяжести хвоста» распределения. При $\alpha=2$ имеем частный случай – нормальное распределение; при $\alpha=1$ имеем распределение Коши. Для промежуточных значений α функции распределения не выражаются аналитически, но их можно получить численно.

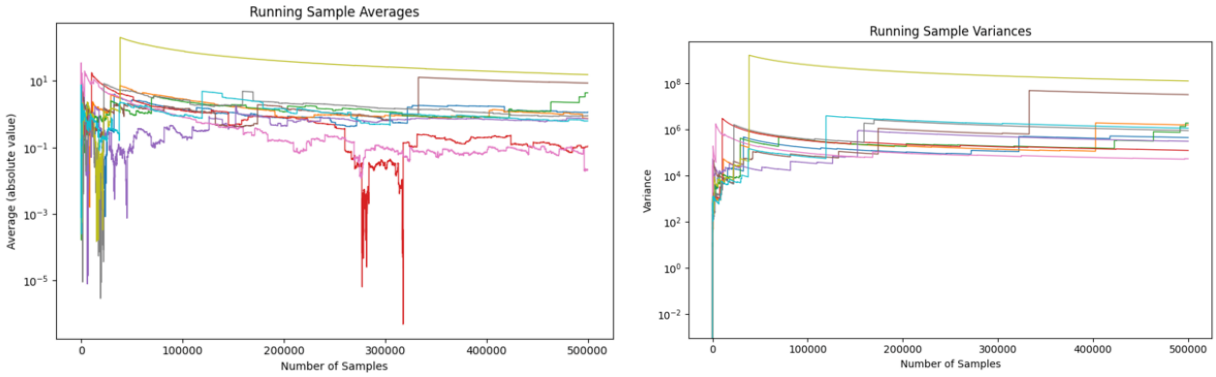
Плотность вероятности	$\frac{1}{\pi\gamma\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]}$



Математическое ожидание	не существует
Медиана	x_0
Мода	x_0
Дисперсия	не существует
Коэффициент асимметрии	не существует
Коэффициент эксцесса	не существует

а б

Рис. 3.29. Распределение Коши: ФПР (а) и числовые характеристики (б) (источник [https://ru.wikipedia.org/wiki/Распределение_Коши])



а б

Рис. 3.30. Изменение выборочных оценок математического ожидания (а) и дисперсии (б) распределения Коши с ростом выборки (источник [https://en.wikipedia.org/wiki/Cauchy_distribution])

Именно это свойство распределения Коши используется в DS: оно моделирует предельно «патологическое» поведение анализируемого бизнес-процесса.

Изменяя α от 2 до 1, мы можем на модели распределения, при сохранении ее колоколообразной формы, проследить, как ведет себя бизнес-процесс при переходе от нормальных ко все более нестабильным условиям.

Как распознать/оценить/предсказать тяжелохвостый процесс? Тяжелые хвосты всегда приносят очень много проблем, так как порождают экстремальные выбросы, при этом исторических данных не всегда достаточно для их оценки.

Печальный, но реальный пример: можно стабильно зарабатывать в течение многих лет, а затем в один день потерять все состояние из-за очередного финансового кризиса, которого никто не ждал.

Более конструктивный пример [Описательная]. В онлайн магазинах основную прибыль приносят продажи в особенные дни («черная пятница», Новый год), и критически важно для бизнеса, чтобы в эти дни сервера работали без сбоев. С другой стороны, покупать все доступные вычислительные мощности с запасом – заведомо потерять прибыль. Так к какой же нагрузке готовиться?

Математическую основу для ответа на подобные вопросы дает теория экстремальных значений (extreme value theory) Фишера-Типпета-Гнеденко. Ее современное развитие представлено в статье [Lee], практическое приложение к анализу интернет-торговли – в статье [Wang H.]. Разработан ряд пакетов [Dutang] с реализацией методов теории экстремальных значений, каждый из которых имеет собственное вѝдение того, как ее правильно применять, поэтому так важен предварительный содержательный анализ ситуации в бизнес-процессе.

Формальные методы оценки параметров распределений с тяжелыми хвостами – например, сравнительной оценки «тяжести» хвоста у двух распределений – являются предметом активных исследований [Акимов, Hilbert], общепризнанных здесь нет. Однако существуют содержательные признаки-предвестники того, что в анализируемом бизнесе может возникнуть (или уже возник) тяжелохвостый процесс. Перечислим некоторые из них:

- Случайная величина, характеризующая бизнес, подчиняется закону Парето («закону 80:20»):
  - состояние 1% населения Земли превышает состояние остальных 99%;
  - по данным Microsoft, 20% багов являются причиной 80% ошибок системы, а 1% багов вызывает половину всех ошибок;
  - 60% населения РФ живет в 5% городов.
- Случайная величина, характеризующая бизнес, формируется не по принципу заговора, а по принципу катастроф:
  - принцип заговора: большое значение суммы является следствием чуть большего, чем обычно, значения у большого числа слагаемых.
  - принцип катастроф: большое значение суммы является следствием большого значения ровно одного из слагаемых.

Например [Николаев], в 2010 году на Земле произошло 2136 землетрясений магнитудой 5 и выше с суммарным числом погибших 164627 чел., из которых около 160000 человек погибли вследствие одного землетрясения на Гаити; в 2011 году произошло 2495 землетрясений магнитудой 5 и выше с суммарным

числом погибших 20927 чел., из которых 19647 чел. явились жертвами одного землетрясения (Тохоку, Япония).

- Случайная величина, характеризующая бизнес, имеет убывающую функцию риска.

Содержательно функция риска $\lambda(t)$ – это мера правдоподобия того, что время ожидания закончится прямо сейчас, в момент t, при условии, что мы уже прождали какое-то время. Например, опыт медицины показывает, что применительно к продолжительности жизни человека функция риска выглядит так:

– возрастающая – за счет естественного старения;
– постоянная – при наличии хронического заболевания;
– убывающая – в послеоперационном периоде.

Математическая оценка функции риска для характерных распределений:

– Парето – $\lambda(t) = \frac{a}{t}$;
– экспоненциальное – $\lambda(t)=1$;
– нормальное – $\lambda(t)>t$.

- Для случайной величины, характеризующей бизнес, перестают выполняться условия центральной предельной теоремы (см. раздел 3.3.2).
- Случайная величина, характеризующая бизнес, формируется в результате мультипликативного, а не аддитивного процесса.

В мультипликативных процессах рост результирующей величины происходит пропорционально ее текущему значению:

– доход на инвестиции («богатые становятся богаче»);
– число подписчиков в социальных сетях («теория предпочтительного присоединения» – вероятность присоединиться к новой вершине графа пропорциональна числу уже присоединенных вершин).

- Выборочные оценки числовых характеристик распределения начинают вести себя странно: перестают сходиться, сходятся не к типичному значению и т.д. В первую очередь это может проявляться на старших моментах (в коэффициенте эксцесса и асимметрии).

При появлении любого из таких признаков дата-сайентисту целесообразно как минимум начать более углубленные исследования ситуации.

Вопросы для самопроверки

1. В чем различие между теорией вероятностей и статистикой в изучении DS?
2. Что такое репрезентативность выборки, какая она бывает, как оценивается?
3. Можно ли сразу сформировать репрезентативную выборку и какими методами?
4. какие методы улучшения репрезентативности выборки входят в набор процедур подготовки данных?
5. Дайте определение дисперсии и среднего квадратического отклонения?
6. Какие виды связей бывают между переменными?
7. Чему равен коэффициент корреляции между независимыми случайными величинами?
8. От чего зависит выбор конкретного функционала качества в задаче DS?

9. Для каких переменных используются непараметрические критерии?
10. Какие метрики качества применятся в задачах классификации, регрессии, кластеризации?
11. Какими метриками можно оценивать эффективность сегментации клиентов?
12. К чему может привезти неправильный подбор функции распределения?
13. Приведите типовые распределения в DS и примеры их использования?
14. Где применяются гамма- и бетта-распределения?
15. Перечислите основные свойства распределения Парето?
16. Какие существуют содержательные признаки-предвестники распределения с тяжелым хвостом в DS?

4. РАЗВЕДОЧНЫЙ АНАЛИЗ В НАУКЕ О ДАННЫХ

4.1. Источники данных

Как правило, начиная бизнес-проект по анализу данных, компания уже располагает некоторым набором собственных данных. Однако в ходе выполнения проекта часто обнаруживается, что имеющихся данных недостаточно, и необходимо их расширить, например, закупить сторонние данные или организовать сбор новых данных. Хотя, согласно методологии CRISP-DM, операция сбора данных входит в фазу 2. Анализ данных, чаще всего необходимость в добавлении новых данных осознается в фазе 3. Подготовка данных, что соответствует итеративному характеру методологии.

Собираемые данные могут быть классифицированы:

- по структуре;
- по происхождению;
- по способу формирования.

Структурированные и неструктурированные данные. Структурированные данные имеют установленный порядок и шаблон и чаще всего являются количественными. Их сбор быстрее и проще в реализации, чем неструктурированные методы сбора данных, и является наилучшим вариантом для крупномасштабных исследований. Однако им часто не хватает гибкости неструктурированных данных.

Неструктурированные данные не имеют установленного порядка или шаблона. Примерами неструктурированных данных являются записи в блогах, комментарии в социальных сетях, электронные письма, формы обратной связи и опросы. Для их обработки могут использоваться средства искусственного интеллекта (ИИ) и обработки естественного языка (NLP).

Во многих ситуациях наилучшим подходом является сочетание структурированных и неструктурированных методов сбора данных.

Первичные и вторичные данные. Данные, которые являются необработанными, оригинальными и извлеченными непосредственно из официальных источников, называются первичными данными. Этот тип данных собирается напрямую с помощью таких методов, как анкетирование, интервью и опросы. Собранные данные должны соответствовать спросу и требованиям целевой аудитории, в отношении которой проводится анализ, в противном случае обработка данных будет затруднена. Несколько методов сбора первичных данных:

- Метод интервью. Данные, собранные в ходе этого процесса, собираются посредством опроса целевой аудитории лицом, называемым интервьюером, а лицо, отвечающее на вопросы интервью, называется интервьюируемым. Задаются некоторые основные вопросы, связанные с бизнесом или продуктом, и записываются в виде заметок, аудио- или видеозаписей, а затем эти данные сохраняются для обработки. Они могут быть как структурированными, так и неструктурированными, например, личные интервью или формальные интервью по телефону, лично, по электронной почте и т. д.

– Метод опроса – это процесс исследования, в ходе которого задается список соответствующих вопросов и ответы записываются в виде текста, аудио или видео. Метод опроса можно получить как в режиме онлайн, так и офлайн, например, с помощью форм на сайте и по электронной почте. Затем ответы на опросы сохраняются для анализа данных. Примерами являются онлайн-опросы или опросы в социальных сетях.
– Метод наблюдения – это метод сбора данных, при котором исследователь внимательно наблюдает за поведением и практикой целевой аудитории, используя какой-либо инструмент сбора данных, и сохраняет полученные данные в виде текста, аудио, видео или любых других необработанных форматов. При этом методе данные собираются напрямую путем отправки нескольких вопросов участникам. Например, наблюдение за группой покупателей и их поведением по отношению к продуктам. Полученные данные будут отправлены на обработку.
– Экспериментальный метод – это процесс сбора данных путем проведения экспериментов, исследований и изысканий. В этом случае очень важно использовать методы планирования эксперимента [Григорьев], которые позволяют минимизировать число необходимых испытаний, установить рациональный порядок и условия проведения исследований в зависимости от их вида и требуемой точности результатов.

Вторичными называются данные, которые уже были собраны и повторно использованы для какой-либо обоснованной цели. Этот тип данных предварительно регистрируется из первичных данных и имеет два типа источников – внутренние и внешние.

Внутренние данные можно найти внутри организации. Это, например, рыночные данные, данные о продажах, транзакциях, данные о клиентах, бухгалтерские ресурсы и т.д. Часто такие данные являются проприетарными. При использовании внутренних источников сокращаются затраты и время на поиск данных.

Данные, которые невозможно найти во внутренних организациях и которые можно получить через внешние сторонние ресурсы, являются внешними данными. Примерами внешних источников являются правительственные публикации, новостные издания. Другими примерами внешних данных могут служить:

– данные датчиков. С развитием устройств Интернета вещей датчики этих устройств собирают данные, которые можно использовать для анализа данных датчиков, чтобы отслеживать производительность и использование продуктов;
– спутниковые данные. Спутники ежедневно собирают огромное количество изображений и данных в терабайтах с помощью камер наблюдения, которые можно использовать для сбора полезной информации;
– веб-трафик. Благодаря быстрым и дешевым возможностям Интернета многие форматы данных, загружаемых пользователями на различные платформы, могут быть спрогнозированы и собраны с их разрешения для анализа

данных. Поисковые системы также предоставляют свои данные через ключевые слова и запросы, по которым чаще всего осуществляется поиск.

Активные и пассивные методы сбора данных. Активный сбор данных требует, чтобы кто-то искал данные, необходимые для своего исследования. Этот метод часто предпочтительнее пассивных методов, поскольку он позволяет дата-сайентисту строить целенаправленное исследование. Кроме того, он помогает избегать смещений выборки, распространенных при пассивных методах сбора данных. Примерами активных методов сбора данных являются опросы, эксперименты, фокус-группы и наблюдения.

Примерами пассивных методов сбора данных являются использование журнала сервера для отслеживания трафика веб-сайта, использование Google Analytics для отслеживания демографических данных посетителей веб-сайта или установка программного обеспечения на компьютеры компании для отслеживания производительности труда сотрудников.

Пассивные методы часто являются самым простым способом получения данных. Пассивные методы сбора данных могут быть автоматизированы. Однако они могут не давать самых релевантных результатов.

4.2. Что такое разведочный анализ данных и зачем он нужен

Статистика, как известно – наука точная, и к ней как никогда применим принцип «garbage in – garbage out» («мусор на входе – мусор на выходе»). Чтобы получить верные и полноценные статистические выводы, перед использованием статистических пакетов или алгоритмов машинного обучения анализируемые данные нужно «очистить». А это, как явствует из предыдущих глав, – далеко не тривиальная процедура.

Например, общепринятая процедура – удалять выбросы из входных данных. А если это вовсе не выбросы, а вполне информативные сэмплы, свидетельствующие о потенциальной «длиннохвостости» распределения? А если говорить о распределении, то из раздела 3.1.4 мы знаем, что одна и та же выборка может с различной вероятностью принадлежать любому закону распределения; проще говоря, имеющуюся выборку можно «подвести» под любое желаемое распределение – так под какое же?

Список таких вопросов можно продолжить. Поэтому привычный для всех подтверждающий статистический анализ (confirmatory data analysis, CDA) имеет смысл предварять ориентировочными и даже интуитивными оценками данных для выявления в них базовых закономерностей. Комплекс процедур для таких оценок под названием «разведочный анализ данных» (exploratory data analysis, EDA) предложил в 1961 году Джон Тьюки (John Tukey).

Несмотря на предварительный характер, разведочный анализ данных имеет надежную формальную основу в таких разделах математики, как непараметрическая и робастная статистика. «Третьим китом» разведочного анализа данных является предложенный Тьюки специализированный аппарат визуализации, способствующий интегральному представлению данных и выявлению закономерностей в них.

EDA – это не жесткий список обязательных процедур, это, скорее, способ осмысления данных и понимания их базовой структуры, которые необходимы аналитику данных для дальнейшей работы. В литературе предлагаются различные списки процедур EDA, но, как правило, в программу-минимум EDA входят следующие позиции:

- описательная статистика по каждой из переменных входного датасета, в том числе выявление экстремальных значений, пропусков и т.п.;
- общая оценка распределения и выявление потенциальных проблем (мультимодальность, скосы, возможное наличие тяжелых хвостов, аномалии в данных и обоснование для их оценки, и пр.);
- обоснованный отбор одного (возможно, несколько) распределений-кандидатов, которые могут обеспечить адекватную и, в то же время, экономичную модель данных.

EDA – это, по сути, творческий процесс, и для его поддержки исключительно важную роль играет визуализация данных, для которой применяются специализированные инструменты.

4.3. Описательная статистика

В статистике различаются два подхода к описанию имеющегося набора данных [Елисеева]. Первый подход – описательная, или дескриптивная, статистика (descriptive statistics) – фиксирует основные закономерности этого конкретного набора данных, не делая предположений о том, из какой генеральной совокупности он мог быть получен. Второй подход – индуктивная статистика, или статистический вывод (inferential statistics, or inductive statistics) – напротив, стремится обобщить информацию из этого набора данных на всю генеральную совокупность.

Это различие сказывается в применяемом аппарате. Индуктивная статистика в полной мере использует математический аппарат теории вероятностей, в то время как описательная статистика является, как правило, непараметрической и использует выборочные оценки для расчета статистических показателей. Для представления результатов описательной статистики, кроме набора статистических показателей, широко используется визуализация, в первую очередь таблицы и графики, которые помогают обобщенно оценить набор данных и выявить его закономерности.

В реальной исследовательской практике оба подхода используются в паре. Любой датасет, используемый в ИТ, сопровождается своей дескриптивной статистикой. В статьях, содержащих экспериментальные исследования, обязательно представляются дескриптивные данные датасета, такие как общий размер выборки, размеры выборки в важных подгруппах (например, в медицинских исследованиях – для каждой группы лечения или воздействия), а также специфические характеристики по каждой подгруппе (например, средний возраст, доля субъектов каждого пола, доля субъектов с сопутствующими заболеваниями и т.д.).

В дескриптивную статистику входят две группы мер – меры центральной тенденции и меры изменчивости или дисперсии:

- меры центральной тенденции (меры положения)
  - среднее значение,
  - медиана,
  - мода
- квантильные оценки
- меры вариации (меры разброса)
  - дисперсия,
  - размах,
  - стандартное отклонение.

Все метрики рассчитываются для каждой переменной входного датасета.

4.3.1. Меры центральной тенденции

Среднее арифметическое и медиана. *Среднее арифметическое* – это сумма всех чисел в наборе данных, деленная на количество чисел. *Медиана* – это число, которое находится в центре набора данных, упорядоченного по возрастанию. Если в наборе данных четное число элементов, то медиана рассчитывается как среднее двух серединных значений.

Пример 1. Для набора данных {3, 7, 12, 16, 19} среднее арифметическое равно 11,4, а медиана равна 12.

*Пример 2.*Для набора данных {3, 7, 12, 15, 16, 19} среднее арифметическое равно 12, а медиана равна (12+15)/2=13,5.

Пример 3. Для набора данных {3, 7, 12, 15, 16, 190} среднее арифметическое равно 40,5, а медиана равна (12+15)/2=13,5.

Сравнение примеров 2 и 3 наглядно показывает, что среднее арифметическое, в отличие от медианы, не устойчиво (не робастно) по отношению к выбросам. Но у медианы есть свой недостаток – ее величина на подвыборках из одного и того же набора данных будут иметь больший разброс, т.е. ее статистическая эффективность хуже, чем у среднего арифметического.

Поясним, что термин «статистически эффективный» содержательно означает «самый лучший по минимуму дисперсии» [Новицкий].

Компромиссным вариантом является *усеченное среднее*. В этом случае из набора данных удаляются заранее определенное количество самых больших и самых маленьких значений, а для оставшихся по стандартной процедуре рассчитывается среднее арифметическое. Этот способ более робастный, чем обычное среднее, и более эффективный, чем медиана, однако его проблема – как определить, сколько значений нужно выбрасывать?

Содержательное обоснование имеет альтернативный подход: нужно удалять из набора данных только экстремальные значения, т.е. выбросы. Существует несколько десятков методов обнаружения выбросов в наборе данных (некоторые из них рассматриваются в разделе 4.3), но все они так или иначе основываются на свойствах распределения, к которому принадлежит набор данных. А именно его-то мы и хотим выявить посредством описательной статистики! Получается своего рода «порочный круг», последствия которого продемонстрированы на примерах в разделе 3. Поэтому лучше удалять выбросы только в

тех случаях, когда есть очень большая уверенность в том, что это – именно выбросы, т.е. значения, не характерные для выборки в целом.

Кроме средних и медиан есть и другие метрики, которые оценивают центральную тенденцию. В частности, удачную комбинацию эффективности и робастности демонстрирует оценка Ходжеса–Лемана. Она строится следующим образом: нужно рассмотреть все пары чисел из набора данных, для каждой пары посчитать среднее арифметическое, а затем из полученных средних выбрать медиану:

$$HL = \underset{i<j}{\text{median}} \left(\frac{x_i + x_j}{2}\right).$$

В таблице 4.1. приведены статистических характеристики различных метрик центральной тенденции [Описательная]. Точка перелома показывает процент значений выборки, который можно заменить на произвольно большие, не ухудшив робастность метрики.

Таблица 4.1. Статистические характеристики различных метрик центральной тенденции

	Среднее	Медиана	Ходжес–Леман
Эффективность	100%	64%	96%
Точка перелома	0%	50%	29%

Что мы видим для оценки Ходжеса–Лемана? С одной стороны, отличную эффективность – почти как у среднего арифметического, т.е. на всех подвыборках из набора данных мы получим очень близкие ее значения. С другой стороны, оценка сохранится, если мы до 29% чисел в наборе данных отнесем к выбросам и, соответственно, заменим. Все это делает оценку Ходжеса–Лемана достойной заменой среднему арифметическому как на околонормальных распределениях, так и на распределениях произвольной формы.

Мода и другие метрики центральной тенденции. *Мода* – это самое часто встречающееся значение в наборе данных.

Пример 4. В наборе данных {3, 7, 12, 16, 19} моды нет.

Пример 5. В наборе данных {3, 7, 12, 12, 19} мода равна 12.

Пример 6. В наборе данных {7, 7, 12, 12, 19} две моды – 7 и 12.

Как правило, мода используется для получения наиболее репрезентативного значения в нечисловых рядах. Популярные цвета сезона, хиты продаж, рейтинги фильмов и музыки, лучшие кафе и закусочные определяются именно модой.

Геометрическое среднее

$$\mu_{geom} = \sqrt[n]{a_1 a_2 \dots a_n},$$

наиболее часто используется для того, чтобы сосчитать среднее значение темпов роста, доходности и т.п. В таких случаях между собой данные взаимодействуют не путем сложения (как при вычислении среднего арифметического), а именно путём умножения (ожидаемая доходность, объём или площадь фигуры вычисляются с помощью умножения), и это меняет подход к выявлению и смыслу средних значений. Например, в финансах при помощи геометрического

среднего считаются средние темпы роста прибыли, выручки, доходность фондов и т.п.

Гармоническое среднее

$$\mu_{harm} = \frac{n}{a_1^{-1}+a_2^{-1}+\cdots+a_n^{-1}}.$$

помогает вычислить среднее арифметическое в рядах чисел, заданных обратными значениями. Это бывает чаще, чем можно подумать. Например, гармоническое среднее используется тогда, когда один и тот же объём работы выполняется с разной производительностью.

Общие рекомендации. Выбор метрики центральной тенденции зависит от целого ряда факторов:

- как взаимодействуют элементы в том бизнес-процессе, из которого взят набор данных (складываются? умножаются? становятся обратными величинами? просто выбираются?);
- каковы статистические характеристики (хотя бы прикидочные) распределений, которые можно использовать для этого бизнес-процесса;
- какие статистические характеристики самих метрик являются самыми важными (эффективность, робастность или что-то еще).

В целом, на уровне первого приближения, общепринятым считается такой подход:

если данные распределены нормально и нет ярких выбросов, то используем среднее арифметическое;

если работаем с ненормальным распределением или видим выбросы, то используем медиану.

Но, когда метрика выбрана, сам расчет не представляет особого труда. В статистических пакетах и библиотеках статистического анализа уже есть готовые функции, которые считают описательные статистики. Например, в библиотеке pandas для Python есть функция describe, которая сразу выведет несколько статистик для одной или всех переменных датасета.

4.3.2. Квантильные оценки

Квантиль – значение, которое заданная случайная величина не превышает с фиксированной вероятностью [Фролов]. Если вероятность задана в процентах, то квантиль называется процентилем или перцентилем.

Например, фраза «90-й процентиль массы тела у новорожденных мальчиков составляет 4 кг» означает, что у 90 % вес меньше либо равен 4 кг, а у 10 % вес больше 4 кг.

Самая популярная квантильная оценка – *медиана*: она делит распределение на две равные части. *Квартили* ($Q_{0,25}$, $Q_{0,5}$ и $Q_{0,75}$) делят распределение на три равные части, *децили* – на 10 равных частей, *процентили* – на 100 равных частей. К стандартным трем квартилям, которые также являются 25-м, 50-м и 75-м процентилями, часто добавляют так называемые нулевой и сотый процентили, которые соответствуют минимуму и максимуму.

В реальной практике мы делим не все множество возможных значений, а только некоторую выборку из *n* значений, поэтому получаем не истинные кван-

тили, а квантильные оценки. Например, для расчета квартильных оценок имеем:

- порядковый номер медианы = (n+1)/2;
- порядковый номер квартиля = номер квартиля*(n+1)/4.

Пример 7. В наборе данных {1, 3, 5, 7, 9, 11, 13}:

- порядковый номер медианы равен (7+1)/2= 4, сама медиана $Q_{0,5}$=7;
- порядковый номер первого квартиля равен 2, сам квартиль $Q_{0,25}$=3;
- порядковый номер третьего квартиля равен 6, сам квартиль $Q_{0,75}$=11.

Если вычисленный таким образом порядковый номер оказывается нецелым (например, 6,3), то значение квартиля рассчитывается как сумма шестого элемента в последовательности и 30% от седьмого элемента.

Предложены разнообразные варианты квантильных оценок, основанные либо на одной порядковой статистике, либо на линейной комбинации двух последовательных порядковых статистик (*i*-й порядковой статистикой называется *i*-й элемент в отсортированной выборке) [Описательная, Фролов]. Они реализованы в большинстве современных библиотек и статистических пакетов.

Межквартильным размахом (англ. Interquartile range, IQR) называется разность между третьим и первым квартилями, то есть $Q_{0,75} – Q_{0,25}$.

Пример 8. В наборе данных {1, 3, 5, 7, 9, 11, 13} межквартильный размах $Q_{0,75} – Q_{0,25}$ = 11–3=8.

Межквартильный размах – характеристика разброса распределения величины. Он является робастным аналогом дисперсии, что особенно важно в случае распределений с большими выбросами.

4.3.3. Меры вариации

Вначале несколько слов о терминологии. В русском и английском языках принята противоположная терминология относительно вариации:

- термин «вариация» по-русски представляет общее понятие для разных метрик, которые описывают разброс значений в распределении; по-английски ему соответствует термин «dispersion»;
- термин «дисперсия» по-русски означает конкретную оценку вариации, равную квадрату стандартного отклонения (StdDev); по-английски ему соответствует термин «variance (Var)».

Стандартное отклонение – одна из самых популярных оценок вариации:

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

Здесь σ – величина стандартного отклонения, n – количество элементов в выборке, $\bar{x}$ – среднее по выборке.

Общеизвестно, что пара «среднее значение + стандартное отклонение» является исчерпывающей характеристикой распределения, если оно – нормальное. А если нет? Эту ситуацию иллюстрирует рис. 4.1.

На рис. 4.1 показаны три распределения, очень похожих на нормальное. Среди них: A – стандартное нормальное распределение с единичным стандартным отклонением; B – сумма стандартного нормального распределения (95%) и

нормального распределения со стандартным отклонением 49 (5%); C – сумма стандартного нормального распределения (90%) и из нормального распределения со стандартным отклонением 9 (5%). Соответственно,

σ_A=1;

$\sigma_B = \sqrt{0.95 \cdot 1^2 + 0.05 \cdot 49^2}$=11;

$\sigma_C = \sqrt{0.9 \cdot 1^2 + 0.1 \cdot 9^2}$=3.

Заметим, что такие смеси – полезный подход для моделирования выбросов и экстремальных значений в выборке. Еще один подход – использование альфа-стабильных распределений – описан в разделе 3.3.3.

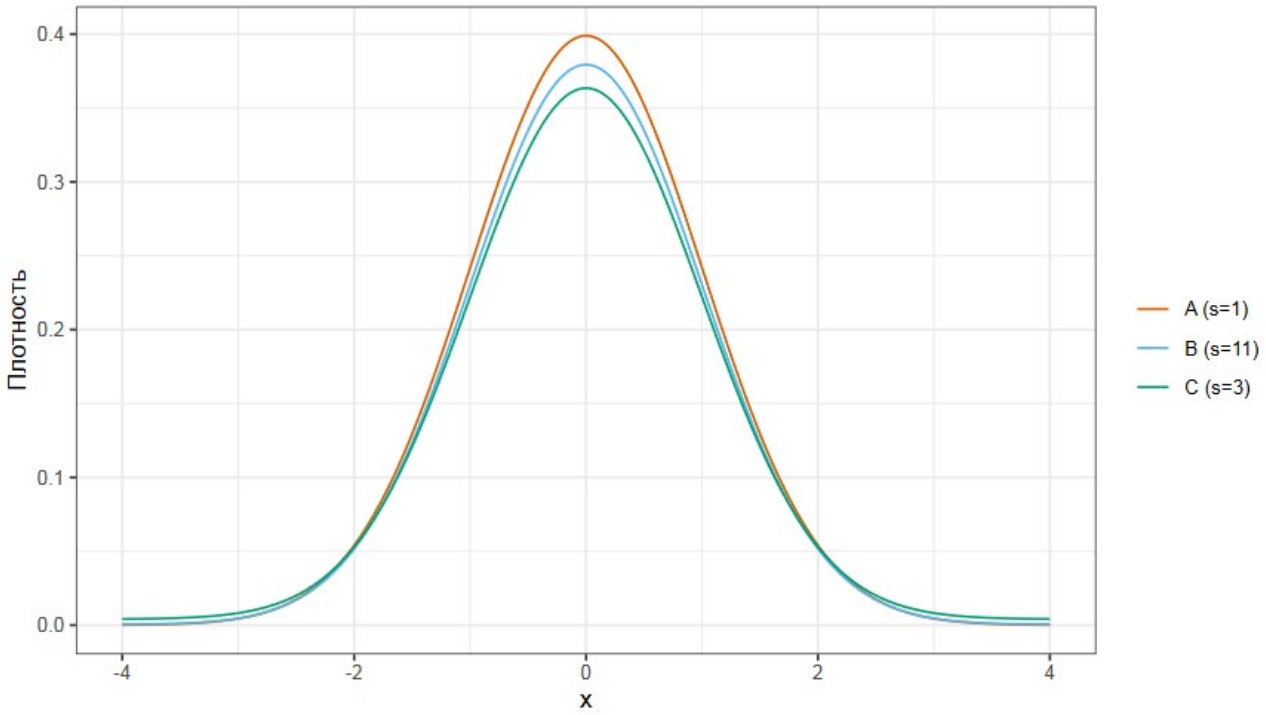


Рис. 4.1. К оценке стандартного отклонения
(источник [https://habr.com/ru/companies/jugru/articles/722342/])

Еще одна проблема стандартного отклонения – высокая чувствительность к поведению распределения на хвостах. Как и среднее арифметическое, оно не является робастным, т.е. может сильно измениться при наличии выбросов.

Робастной альтернативой стандартному отклонению может служить *медианное абсолютное отклонение*:

$$MAD = C_n \cdot median(|x_i - median(x)|).$$

О робастности медианных характеристик уже говорилось в разделе 4.2.1. Здесь подчеркнем только ясный физический смысл метрики MAD, который хорошо виден из рисунка 4.2: интервал «медиана плюс-минус MAD» всегда покрывает ровно 50% распределения.

Чтобы значение MAD соответствовало стандартному отклонению нормального распределения, вводится нормировочный множитель C_n, который, как видно из обозначения, зависит от объема выборки n. Если размер выборки

очень большой, т.е. при $n \to \infty$, C_n=1.48, в других случаях нужна коррекция этой константы [Akinshin].

Есть другие робастные оценки стандартного отклонения, среди которых выделяются оценки, идеологически похожие на оценку Ходжеса-Леманна для центральной тенденции. Например, оценка Q_n [Rousseeuw] строится следующим образом: рассчитываются абсолютные попарные разницы всех элементов выборке, и из них выбирается первый квартиль:

$$Q_n = C_n \cdot Q\left(\left|x_i - x_j\right|_{i<j}, 0.25\right),\ C_\infty = 2.2191.$$

В [Rousseeuw] показано, что для сравнительно больших выборок (n>30) эта оценка является лучшей по соотношению робастности и эффективности среди всех мер вариации. При n>20 целесообразно пользоваться другими оценками.

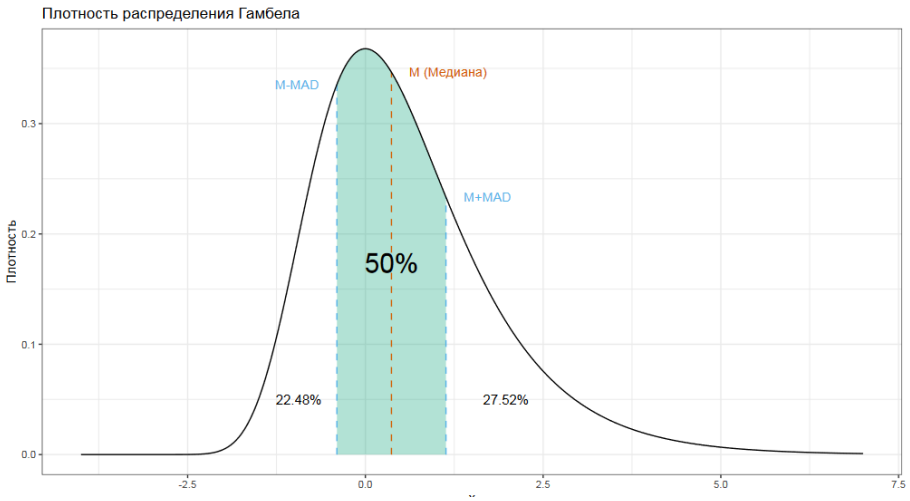


Рис. 4.2. Зона покрытия MAD
(источник [https://habr.com/ru/companies/jugru/articles/722342/])

4.4. Выявление аномалий

4.4.1. Виды аномалий в данных

«Выпадающие» точки, или аномалии, в наборе данных являются одной из самых опасных проблем, с которой сталкиваются почти все проекты по работе с данными. Большинство датасетов, которые выкладываются в открытый доступ для бенчмарка, не содержат выбросов. Однако реальные наборы данных всегда содержат некоторые «нетиповые» сэмплы. Задача дата-сайентиста – обнаружить их и соответствующим образом обработать.

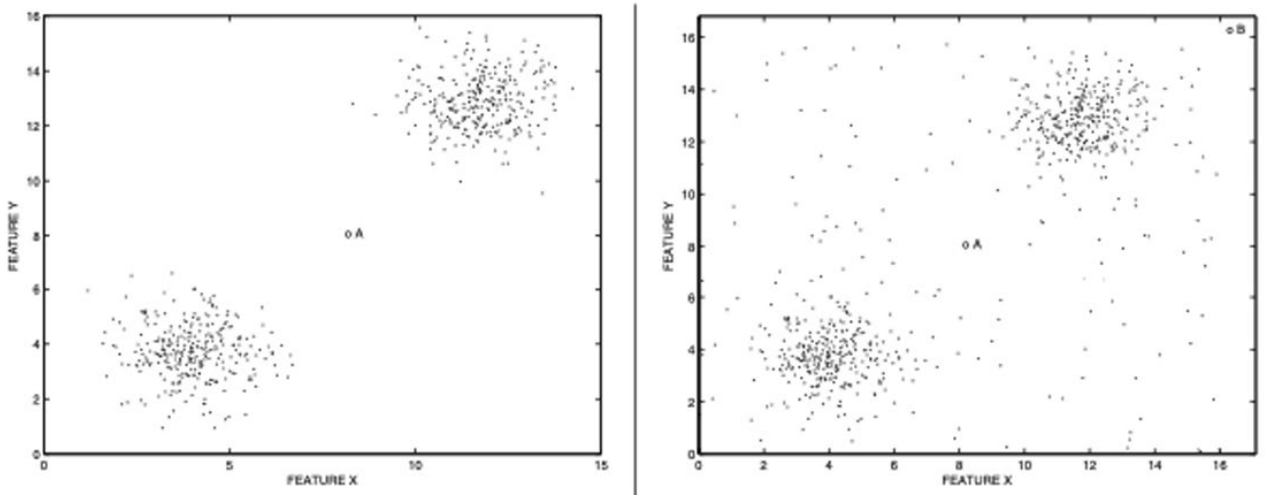
В науке о данных аномалии содержательно определяются как значительно отклоняющиеся или не соответствующие друг другу точки из набора данных. Однако перевести это определение в формальное гораздо сложнее, что иллюстрирует рис. 4.3.

На рис. 4.3, а, б, показаны идентичные кластеры точек в двумерном пространстве признаков. Точка данных A кажется отличной от остальных точек данных. Но эта точка на рис. 4.3, а, явно является выбросом, а на рис. 4.3, б, она окружена шумом, и уже довольно сложно сказать, шум это или выброс.

При обнаружении аномалий возникают вопросы:

– на каком основании выделять аномалии, т.е. что такое «значительное отклонение от нормы»?
– обнаружила ли модель все аномалии в конкретном наборе данных?

Получить исчерпывающий ответ на эти вопросы формальными средствами невозможно. Здесь требуется учитывать механизм возникновения аномалий, контекст задачи и конкретный сценарий ее исполнения, а также во многом опираться на интуицию дата-сайентиста.



а б

Рис. 4.3. Является ли точка А аномалией? (источник [Sikder])

Выделяются аномалии двух типов – выбросы (outliers) и новинки (novelties). Выбросы – это аномальные или экстремальные точки данных, которые существуют в обучающих данных. Напротив, новинки – это новые или ранее не встречавшиеся случаи по сравнению с исходными (обучающими) данными.

Проиллюстрируем эту разницу на примере данных суточной температуры в городе [Tuychlev]. Большую часть дней температура колеблется от 20°C до 30°C. Однодневный скачок до экстремального значения 40°C является выбросом, поскольку он значительно отклоняется от обычного дневного диапазона температур.

Теперь представьте, что в городе устанавливают новую, более точную систему мониторинга погоды. В результате в наборе данных начинают постоянно фиксироваться немного более высокие температуры в диапазоне от 25°C до 35°C. Это устойчивое повышение температуры является новинкой, представляющей новую закономерность, введенную улучшенной системой мониторинга.

Термин «аномалии» можно использовать для определения любого аномального случая в любом контексте. Определение типа аномалий имеет решающее значение в плане того, что с ними делать дальше: с выбросами нужно бороться, т.е. выявлять их с целью устранения из набора данных, в то время как новинки – это ценный материал для анализа, и их нужно выделять в датасете

для дальнейшего исследования. Однако для обнаружения аномалий любого типа, как правило, используются одни и те же алгоритмы; поэтому в контексте обнаружения термины «аномалии» и «выбросы» часто используются как синонимы.

В количественном выражении проблема сводится к оценке уровня загрязнения (contamination level), т.е. процента выбросов, в наборе данных. Из-за нее мы не можем надежно измерить производительность классификаторов выбросов, а также не можем проверить их результаты.

Однако существует целый ряд приемов и алгоритмов, позволяющих дать приближенные оценки. Например, в специализированной библиотеке для оценки выбросов Python Outlier Detection (PyOD) имеется параметр contamination, который по умолчанию установлен на 0,1. В модели IsolationForest из Sklearn имеется внутренний алгоритм для автоматического определения уровня загрязнения.

Еще одна проблема обнаружения аномалий – это дисбаланс данных. Аномалии встречаются существенно реже по сравнению с обычными экземплярами, что приводит к дисбалансу наборов данных и сильно затрудняет выявление аномалий типовыми средствами машинного обучения.

Метрики, используемые при выявлении аномалий, также весьма разнообразны. С практикой выбора таких метрик для временных рядов можно познакомиться в [Кацер].

К настоящему времени предложены десятки, если не сотни алгоритмов выявления аномалий. В литературе [Sikder, Wang, Pimentel] принято делить их на базовые категории, которые кратко охарактеризованы ниже. Следует подчеркнуть, что они во многом пересекаются, поэтому один и тот же алгоритм в разных источниках может быть отнесен к разным категориям. В данном пособии в основном используется наиболее современная категоризация, предложенная в [Sikder].

4.4.2. Статистические и вероятностные алгоритмы выявления аномалий

Предполагается, что данные генерируются из некоторого базового распределения вероятностей D, которое может быть оценено с использованием имеющегося набора данных. Эта оценка $\widehat{D}$ выполняется вероятностными или статистическими методами (см. раздел 3.3.4) и обычно рассматривается как модель обычных данных (без выбросов). Затем для некоторых параметров $\widehat{D}$ устанавливается порог новизны, который в вероятностном смысле разделял обычные данные и выбросы.

Простейшим в этой категории является *тест Граббса* («правило трех сигм»). В первоначальном виде он применяется для одномерного нормального распределения. Параметром отсечения является стандартное отклонение: все точки, выходящие за пределы утроенной величины стандартного отклонения от среднего значения, объявляются выбросами. Разработана модификация этого теста для многомерного случая [Pimentel]. Несмотря на свою простоту, тест Граббса остается мощным инструментом выделения выбросов, особенно для небольших датасетов. Однако многократное применение теста Граббса к одно-

му и тому же набору данных (после удаления выбросов) может увеличить вероятность ошибки типа I (неправильного отклонения нулевой гипотезы). Поэтому важно соответствующим образом скорректировать уровень значимости или использовать альтернативные методы, подходящие для обнаружения множественных выбросов.

Не менее популярным является *метод Тьюки*. В отличие от теста Граббса, который фокусируется на самых экстремальных значениях, метод Тьюки рассматривает разброс данных с использованием межквартильного размаха (*IQR*) между первым (*Q*1) и третьим (*Q*1) квартилями, т.е. нацелен не на поиск одного или нескольких абнормальных значений, а на оценку структуры данных в целом. Выбросами объявляются точки датасета, лежащие ниже *Q*1–1.5**IQR* или выше *Q*3 + 1.5**IQR*. Коэффициент 1,5 может быть скорректирован для конкретных приложений и с соответствующими допущениями. Обычно для «мягких» выбросов используется значение 1.5, а для «экстремальных» – значение 3.

Метод Тьюки часто представляется в графической форме, в виде диаграммы «ящик с усами» (рис. 4.4).

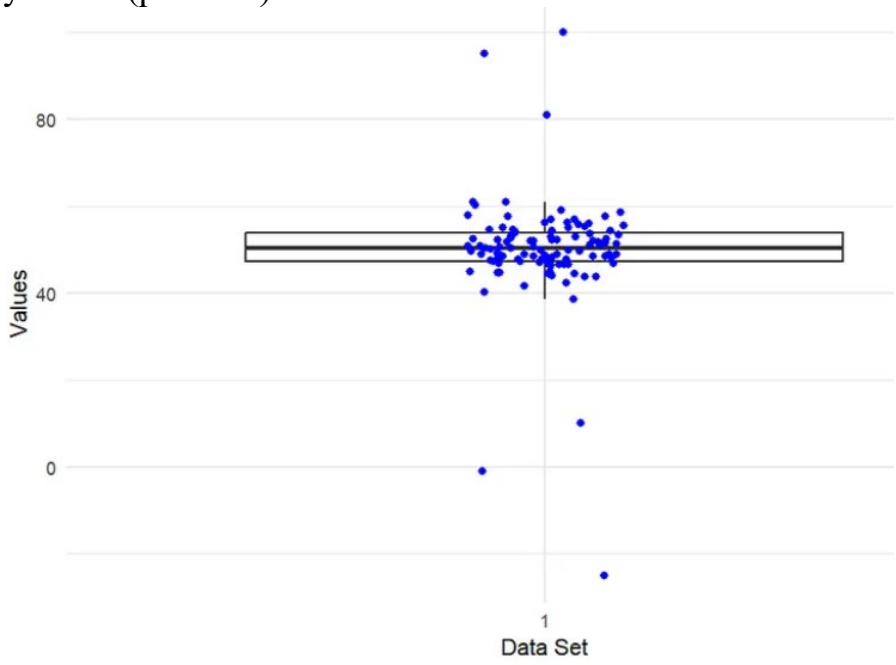


Рис. 4.4. Определение аномалий по методу Тьюки
(источник [https://medium.com/data-and-beyond/outlier-detection-in-r-tukey-method-or-why-you-need-box-and-whiskers-3c35d9ad8fb3])

Однако в реальной практике подобрать удачное значение отсекающего коэффициента крайне сложно. Чаще всего выбросами помечаются обычные нормальные элементы выборки, что уменьшает эффективность подхода. Но если ослабить условия интервала, то мы начнем пропускать важные выбросы, что поломает робастность метода.

В качестве средства оценки функции плотности распределения может выступать гистограмма (см. раздел 3.3.4). *Гистограммный метод* выявления выбросов (Histogram-based Outlier Score, HBOS) предложен в [Goldstein]. Идея метода проиллюстрирована на рис. 4.5. Для каждой фичи набора данных строится

отдельная гистограмма, и относительное количество точек p, попавших в каждый столбик гистограммы, используется в качестве оценки плотности. Показатель HBOS для каждой точки вычисляется как произведение обратных логарифмов рассчитанной плотности p:

$$HBOS(p) = \sum_{i=0}^{d} \left(\log \frac{1}{hist_i(p)}\right).$$

Метод HBOS плохо ищет локальные выбросы, но обладает линейной, O(n), т.е. очень высокой скоростью вычислений и поэтому может быть успешно использован для больших объемов датасетов и даже для расчетов в реальном времени [Wang B.].

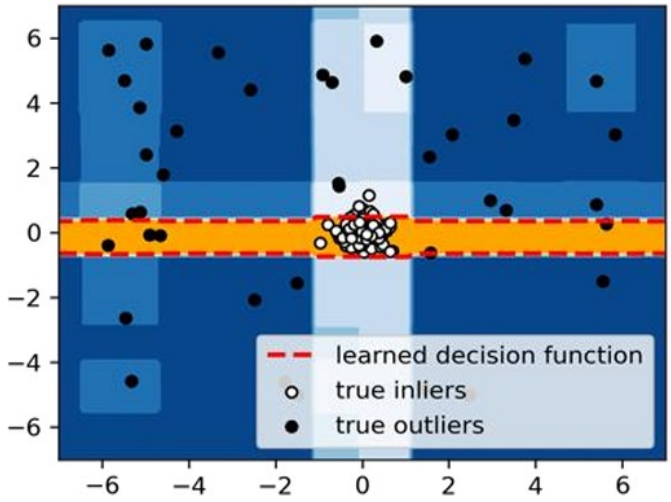


Рис. 4.5. Определение выбросов по гистограммному методу (источник [Sikder])

Методы Тьюки и HBOS являются представителями подкатегории непараметрических методов обнаружения аномалий. Эта подкатегория весьма обширна: для каждого метода непараметрической оценки плотности вероятности (см. раздел 3.3.4) предложен целый «веер» методов обнаружения аномалий. Есть также методы обнаружения аномалий, использующие параметрическую оценку плотности вероятности, в первую очередь на основе гауссовской смеси моделей и на основе построения регрессии. Со всеми методами можно более подробно познакомиться по обзорам [Sikder, Wang, Pimentel].

Вероятностные и статистические методы обнаружения выбросов отличаются своей надежной математической основой и простотой имплементации. Среди недостатков отмечаются плохая работа на реальных данных, в особенности на небольших датасетах, что связано с трудностями оценки плотности распределения, и высокие вычислительные затраты при переходе к многомерным данным. В целом в этой категории методов наиболее привлекательными считаются непараметрические методы, в первую очередь – основанные на ядерной оценке плотности распределения.

4.4.3. Алгоритмы выявления аномалий на основе оценки расстояния

Методы на основе расстояний обнаруживают выбросы путем вычисления расстояний между точками. Точка из набора данных, которая находится на большом расстоянии от своего ближайшего соседа, считается выбросом. В рам-

ках этой концепции могут быть введены различные определения выбросов, в том числе [Wang]:

- в наборе данных объект O является дистанционно-определяемым (Distant-Based, DB) (p, D)-выбросом, если незначительная доля p объектов в наборе данных лежит за пределами расстояния D от O;
- (p, D)-выбросы – это точки с менее чем p различными образцами в пределах расстояния D;
- выбросы – это верхние n примеров, расстояние до k-го ближайшего соседа которых наибольшее;
- выбросы – это верхние n примеров, среднее расстояние до k-го ближайшего соседа которых наибольшее.

Метод k-ближайших соседей (k-NN) (рис. 4.6) основан на предположении, что нормальные точки имеют близких соседей в «нормальном» наборе данных, в то время как выбросы расположены далеко от этих точек. Точка объявляется выбросом, если она расположена далеко от своих соседей.

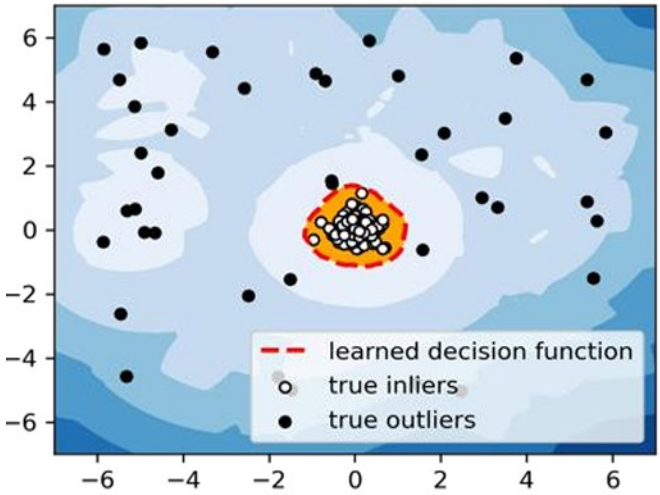


Рис. 4.6. Определение выбросов по методу k-NN (источник [Sikder])

Классический алгоритм kNN строится следующим образом:

1. Вычисляется расстояние между тестовым и всеми обучающими образцами.
2. Из них выбирается k ближайших образцов (соседей), где число k задаётся заранее.
3. Итоговым прогнозом среди выбранных k ближайших образцов будет мода в случае классификации и среднее арифметическое в случае регрессии.
4. Предыдущие шаги повторяются для всех тестовых образцов.

Для одномерных и многомерных (n-мерных) непрерывных атрибутов чаще всего используется евклидово расстояние

$$d(a,b) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2},$$

но могут использоваться и другие меры, такие как манхэттенское расстояние

$$d(a,b) = \sum_{i=1}^{n}|a_i - b_i|.$$

Классический алгоритм kNN отличается простотой в реализации и интерпретации, а также высокой точностью прогнозов при правильном подборе k и метрики расстояния. Однако он использует для поиска ближайших соседей метод полного перебора, что делает его слишком медленным при работе с большим объемом данных.

Для решения данной проблемы в реализации scikit-learn предусмотрены более продвинутые методы, основанные на бинарных деревьях, что позволяет получить значительный прирост в производительности.

Кроме того, специально для поиска выбросов разработаны различные подходы, которые используют *прунинг* (прореживание данных), т.е. сокращают поиск, игнорируя данные, которые нельзя считать выбросами.

Например, прунинг реализован в алгоритме ORCA. После нахождения ближайших соседей для точки порог, основанный на оценке самого слабого выброса (т. е. выброса, который находится ближе к точке), найденного до сих пор, устанавливается для любой новой точки данных. Поэтому близкие точки отбрасываются алгоритмом. В результате для достаточно рандомизированных данных средняя сложность поиска ближайшего соседа будет приблизительно линейной. Еще более эффективные алгоритмы на основе прунинга представлены в литературе [Sikder, Pimentel].

В целом методы выявления аномалий, основанные на расстоянии, широко распространены, поскольку они имеют сильную теоретическую основу и вычислительно эффективны. Одним из критических недостатков большинства методов, основанных на расстоянии, является их неспособность хорошо масштабироваться для очень многомерных наборов данных.

4.4.4. Алгоритмы выявления аномалий на основе кластеризации

Методы выявления аномалий на основе кластеризации идеологически близки к предыдущей группе методов, поэтому в некоторых обзорах [Sikder] они рассматриваются как единая категория.

Идея этой категории методов заключается в следующем. В результате кластеризации заданного набора данных он разделится на более и менее разреженные подзоны (популяции). Более плотные популяции образуют кластеры, разреженные подзоны, как правило, содержат большинство выбросов. Другими словами, кластеры меньшего размера, которые содержат значительно меньше точек данных, чем другие кластеры, помечаются как выбросы. Поэтому большинство алгоритмов кластеризации получают выбросы как побочный продукт своего анализа.

Оценка выброса может быть рассчитана с использованием расстояния между точкой данных и ближайшим центроидом кластера. Из-за различной формы кластеров удачной мерой расстояния может служить расстояние Махаланобиса, которое хорошо масштабируется для кластеров. Математически расстояние Махаланобиса от точки данных X до распределения кластера с центроидом μ и ковариационной матрицей Σ равно

$$MB(X, \mu, \Sigma)^2 = (X - \mu)\Sigma^{-1}(X - \mu)^T .$$

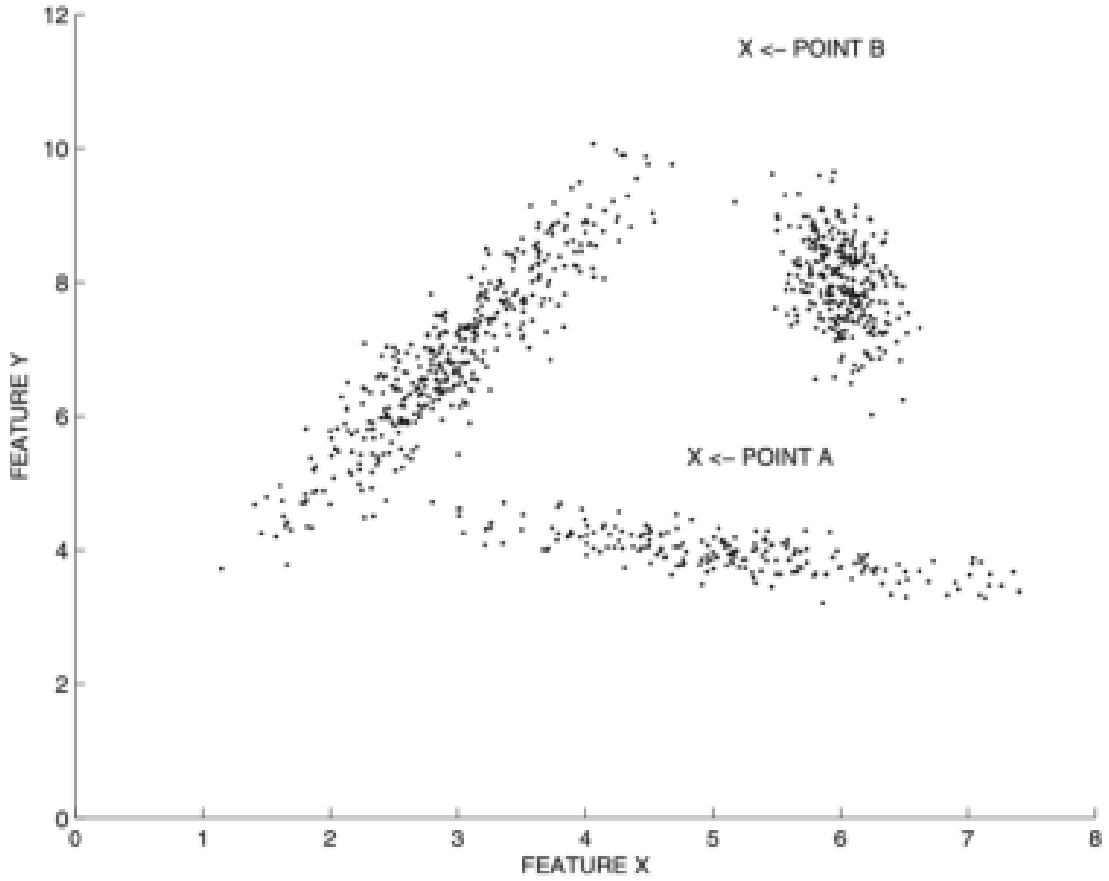


Рис. 4.7. Определение выбросов на основе кластеризации (источник [Sikder])

Рис. 4.7 иллюстрирует эффект выбора расстояния при идентификации выбросов. Евклидово расстояние будет считать точку «A» выбросом по сравнению с точкой «B». Однако расстояние Махаланобиса, учитывая локальность данных, делает точку «B» более аномальной, чем точку «A. Таким образом, удачное определение надлежащего количества кластеров и подходящей меры расстояния является определяющим при определении выбросов путем кластеризации.

Так как кластер – формально не определимое понятие, то для построения кластеров используются различные эвристические приемы, каждый из которых порождает свою подгруппу методов определения аномалий [Sikder, Wang], в том числе:

- кластеризация путем разбиения (partitioning) – основана на вычислении расстояния между точками набора данных;
- кластеризация на основе центроидов – примером является алгоритм k-means;
- кластеризация на основе плотности – примером является алгоритм DBSCAN;
- кластеризация на основе иерархии – алгоритмы разбивают кластер на различные уровни, структурированные как дерево и др.

Методы выявления аномалий на основе кластеризации имеют целый ряд преимуществ. Для их реализации не требуется никаких предварительных дан-

ных о типе распределения. После обучения на кластерах можно вставить дополнительные новые точки, а затем проверить их на выбросы. Они применимы для разных типов данных. Они хорошо масштабируются. Иерархические методы позволяют сформировать неизотропные кластеры, а также создают несколько вложенных разделов, которые дают пользователям возможность выбирать разные части в соответствии с их уровнем сходства. Все это позволяет использовать кластерные методы выявления аномалий в режиме реального времени и на потоках данных.

Недостатки кластерных методов можно разделить на три группы:

- невозможно сформировать количественную оценку уровня выброса (насколько важным он является);
- в процесс кластеризации невозможно внести текущие изменения. В большинстве методов кластеризации необходимо заранее задать ожидаемое число кластеров, а часто – и ожидаемую форму кластеров. Еще одним недостатком является неопределенность критериев остановки;
- кластерные методы демонстрируют серьезное ухудшение эффективности в многомерных пространствах.

Несмотря на проблемы и недостатки, методы, основанные на кластеризации, полезны для обнаружения выбросов, особенно в потоках данных.

4.4.5. Алгоритмы выявления аномалий на основе плотности

Выявление аномалий на основе плотности является одним из самых популярных и распространенных методов. Основной принцип заключается в том, что точка выброса находятся в разреженной области, тогда как нормальные точки –в более плотной области. На рис. 4.8 представлен двумерный набор данных, где помеченные точки «A» и «B» значительно отделены от остальной части «густонаселенных» кластеров, поэтому являются точками выброса в этом наборе данных.

В [Breunig] предложен *метод* обнаружения выбросов на основе вычисления *локального фактора выбросов* (Local Outlier Factor, *LOF*). Для каждой точки набора данных рассчитывается ее kNN-окрестность $kNN(p)$, затем определяется плотность локальной достижимости (local reachability density, lrd):

$$lrd(p) = \frac{1}{\frac{\sum_{o \in kNN(p)} reach\text{-}dist_k(p \leftarrow o)}{|kNN(p)|}},$$

и показатель LOF (Local Outlier Factor):

$$LOF_k(p) = \frac{1}{|kNN(p)|} \sum_{o \in kNN(p)} \frac{lrd_k(o)}{lrd_k(p)}.$$

LOF точки (рис. 4.7) оценивает соотношение локальной плотности области вокруг точки и локальных плотностей ее соседей. Размер окрестности точки определяется областью, содержащей указанное пользователем минимальное количество точек. LOF принимает высокие значения для выбросов, поскольку он количественно определяет, насколько изолирована точка относительно плотности ее окрестности. LOF ранжирует точки, учитывая только плотность окрестностей точек, и поэтому он может пропустить потенциальные выбросы, плотности которых близки к плотности их соседей.

Многочисленные модификации метода LOF широко представлены в литературе [Sikder, Wang, Pimentel].

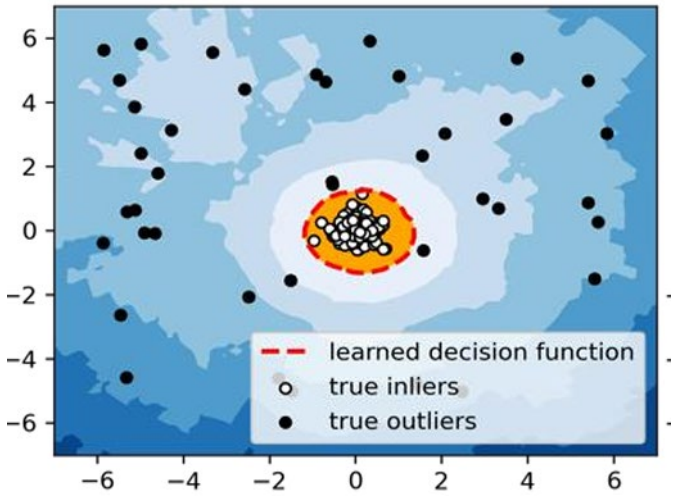


Рис. 4.8. Определение выбросов по методу LOF (источник [Sikder])

Алгоритмы оценки выбросов на основе плотности являются непараметрическими, т.е. не предполагают какой-либо предопределенной модели распределения. Это делает алгоритм простым в настройке. Фактически, только настройка одного гиперпараметра дает хорошие результаты. Они могут как определять локальные, так и глобальные выбросы, что делает их полезными для применения в реальном мире. В этом плане они часто превосходят другие статистические алгоритмы.

В то же время алгоритмы на основе плотности являются вычислительно затратными, что особенно проявляется при работе с данными высокой размерности и в реальном времени. Этот недостаток можно преодолеть, исключая из датасета точки, которые заведомо не являются выбросами [Su].

4.4.6. Методы выявления аномалий на основе машинного обучения

Использование подпространства. Если датасет имеет высокую размерность, то для выявления выбросов может оказаться эффективным переход к обучению в подпространстве. Основная идея заключается в выявлении разнородных подмножеств с разной размерностью. Алгоритмы проецируют наборы данных с высокой размерностью на разреженное и низкоразмерное подпространство. Выбросами считаются те точки, которые находятся в разреженном подпространстве, поскольку они характеризуются более низкой плотностью.

Проецирование пространства с высокой размерностью на разреженное подпространство занимает много времени. Для повышения его эффективности предложены различные способы. Например, *метод Subspace OD* (SOD) [Huang] рассчитывает корреляцию каждого объекта с его общим ближайшим соседом; вместо того, чтобы учитывать расстояние от объектов до его соседей, модель учитывает ранги каждого объекта, которые находятся близко к объекту. Краткий обзор других методов можно найти в [Sikder].

Несмотря на предложенные модификации, методы перехода в подпространства остаются вычислительно дорогими.

Применение активного обучения. Активное обучение реализует концепцию human-in-the-loop, в которой алгоритм обучения может выборочно запрашивать у аналитика-человека метки входных экземпляров для повышения точности прогнозирования. Общая цель алгоритмов активного обучения – минимизировать количество запросов для достижения целевой производительности.

Аналитик-человек предоставляет алгоритму обратную связь по истинным меткам (номинальным или аномальным), в соответствии с которой алгоритм обновляет параметры своей модели (рис. 4.9). Применение активного обучения для обнаружения аномалий позволяет даже при значительном увеличении объема данных сократить долю ложноположительных результатов, что исключительно важно в реальных условиях.

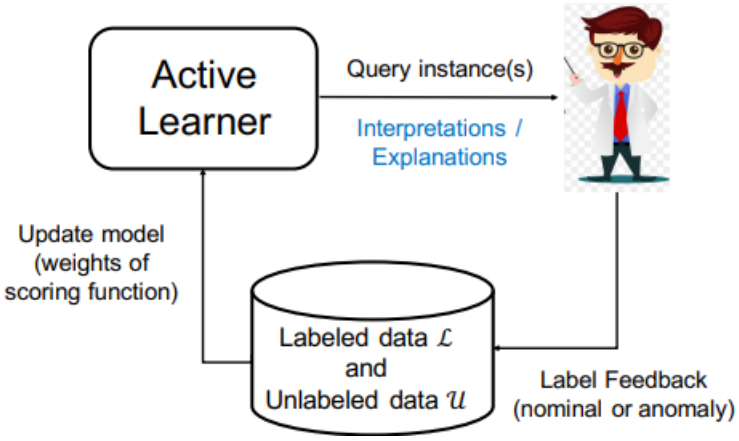


Рис. 4.9. Активное обучение при поиске аномалий (источник [Das])

Алгоритм леса изоляции (Isolation Forest, IFOR) состоит из ансамбля деревьев изоляции. Каждое дерево разбивает исходное пространство признаков случайным образом. В каждом узле дерева сначала случайным образом выбирается признак, а затем точка разделения для этого признака выбирается равномерно случайным образом (рис. 4.10). Эта операция разбиения выполняется до тех пор, пока каждый экземпляр не достигнет своего собственного листового узла.

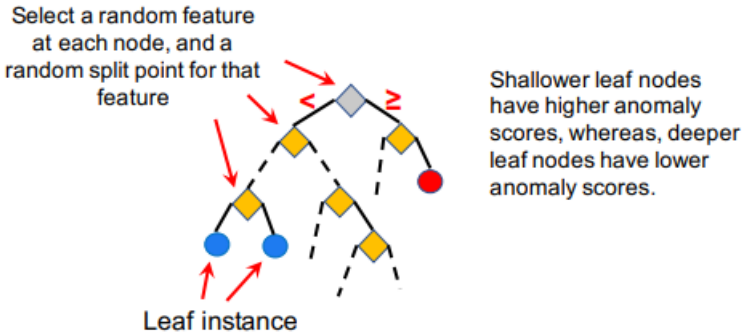


Рис. 4.10. Алгоритм деревьев изоляции (источник [Das])

Основная идея алгоритма заключается в том, что аномальные экземпляры, которые обычно изолированы в пространстве признаков, с помощью этой стратегии разбиения достигают листовых узлов быстрее, чем нормальные экзем-

пляры, которые принадлежат более плотным областям. Следовательно, путь от корневого узла короче до листьев аномальных экземпляров по сравнению с листьями номинальных экземпляров. В результате аномальные экземпляры получают в среднем более высокие оценки, чем номинальные экземпляры. Поскольку каждый экземпляр принадлежит только нескольким листовым узлам, векторы оценок разрежены, что приводит к низким затратам памяти и вычислений. В рамках леса обратная связь по любому отдельному экземпляру передается многим другим экземплярам, что сокращает требуемое количество вмешательств человека-оператора.

Алгоритм IFOR использует взвешенную линейную комбинацию оценок аномалий от членов ансамбля. Этот подход хорошо работает, когда ансамбль содержит большое количество членов, которые сами по себе сильно локализованы, например, конечные узлы детекторов на основе дерева.

Алгоритм GLocalized Anomaly Detection (GLAD). Если в ансамбле небольшое количество детекторов, то весьма вероятно, что отдельные детекторы неверны, по крайней мере в некоторых локальных частях входного пространства признаков.

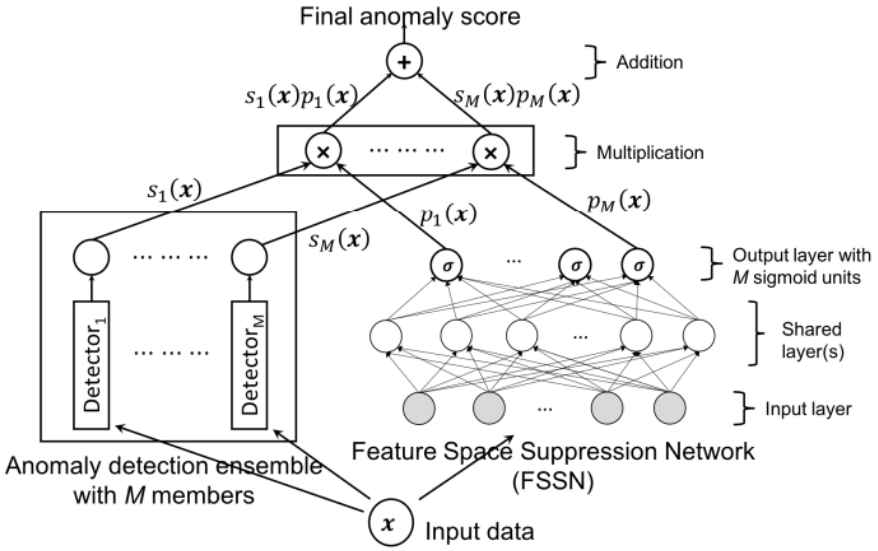


Рис. 4.11. Алгоритм GLAD (источник [Das])

Алгоритм GLAD (рис. 4.11) изучает локальную релевантность каждого члена ансамбля в пространстве признаков с помощью нейронной сети, используя обратную связь по меткам от аналитика-человека. В каждой итерации выбирается один непомеченный экземпляр для запроса и обновляются веса нейронной сети (FSSN), чтобы скорректировать локальную релевантность каждого члена ансамбля по всем помеченным экземплярам.

Чтобы уменьшить усилия аналитика-человека, алгоритм настраивается так, чтобы вероятность обнаружения аномалий была очень высокой с первой же итерации обратной связи.

Алгоритмы выявления аномалий на основе глубокого обучения получили исключительное развитие в последние несколько лет [Chalapathy]. Эти методы обеспечивают высокую точность при обнаружении выбросов в критически важной инфраструктуре, здравоохранении и обороне, для обнаружения мошеннических транзакций и в многих других областях. Они эффективны для боль-

ших размерных наборов данных, при иерархической структуре признаков, а также во временных рядах.

В моделях используются большинство известных методов и архитектур глубокого обучения, в том числе автоэнкодеры, сети LSTM, графовые сети, рекуррентные нейронные сети, обучение с подкреплением и т.д. [Sikder].

4.4.7. Инструментарий для выявления аномалий

Существует множество библиотек и инструментов для выявления аномалий. Среди множества инструментов самыми популярными, согласно [Sikder], являются следующие:

- Scikit-learn (Python): включает такие популярные алгоритмы, как Isolation Forest и LOF;
- Python Outlier Detection (PyOD) (Python): включает ансамблевые методы и методы глубокого обучения;
- ELKI (Java): open-source платформа, обеспечивает бенчмаркинг и простые тесты эффективности алгоритмов выявления аномалий;
- Python Streaming Anomaly Detection (PySAD) (Python): open-source библиотека для поточной идентификации выбросов; содержит более 15 алгоритмов;
- Scalable Unsupervised OD (SUOD) (Python): работает поверх PyOD, обеспечивает акселерацию для больших объемов данных;
- Rapid Miner (Java): поддерживает алгоритмы семейства LOF;
- MATLAB: коммерческий продукт, поддерживает много алгоритмов;
- Time-series Outlier Detection System (TODS) (Python): специализирован для работы в потоках многомерных данных;
- Skyline (Python): работает в почти реальном времени;
- Telemanom (Python): использует архитектуру Long Short-Term Memory в многомерных временных рядах;
- DeepADoTS (Python): коллекция алгоритмов глубокого обучения для бенчмаркинга алгоритмов поиска аномалий во временных рядах;
- Numerical Anomaly Benchmark (NAB) (Python): бенчмаркинг алгоритмов поиска аномалий в реальном времени и в потоках;
- Datastream.io (Python): open-source инструмент для выявления выбросов в реальном времени.

4.5. Оценка распределений

4.5.1. Статистические методы восстановления распределений

Задача, которую решает дата-сайентист в этом случае, является базовой задачей математической статистики [Справочник]: имея конечную выборку из генеральной совокупности, подобрать функцию распределения, адекватно описывающую всю генеральную совокупность. Формально говоря, требуется оценить плотность распределения $p(x)$ по выборке $X_l = \{x_i\}_{i=1}^m$ независимых случайных векторов, распределенных по этому закону.

В общем случае эта задача однозначного решения не имеет: одна и та же выборка может с различной вероятностью принадлежать любому закону распределения (простейший пример приведен на рис. 3.11). Поэтому, как отмечалось в разделе 3.3.1, в реальной практике можно искать ее частное решение для конкретной ситуации на домене, например:

- в [Peng] решается задача выбора распределения для сильно разреженных датасетов, для этого используется теория доказательств (evidence theory) Демпстера-Шафера;
- в [Тырсин] предложен метод выбора закона распределения непрерывной случайной величины из заданного множества моделей распределений. Идея метода состоит в непрерывном отображении эмпирического выборочного распределения на эталонную прямую, которая задает отношение порядка на любом конечном множестве модельных функций распределения. В качестве наиболее вероятного закона для исходной выборки выбирается тот, для которого среднеквадратическая ошибка отображения будет минимальной.

Таких частных методов уже предложено огромное количество, и поток публикаций на эту тему не иссякает. Для ориентации в этом потоке большую помощь оказывает базовая классификация [Оценивание], согласно которой существуют следующие основные подходы к оцениванию плотности распределения:

- Непараметрическое восстановление плотности
  - Гистограммный метод оценивания,
  - Методы локального оценивания,
  - Метод оценивания с помощью аппроксимации функции плотности.
- Параметрическое восстановление плотности
  - Метод максимального правдоподобия,
  - Метод моментов
- Восстановление смесей распределений
  - EM-алгоритм

Непараметрический подход. Предположения о плотности распределения $p(x)$ не делаются, и аппроксимирующая функция $\tilde{p}(x)$ строится из самых общих соображений.

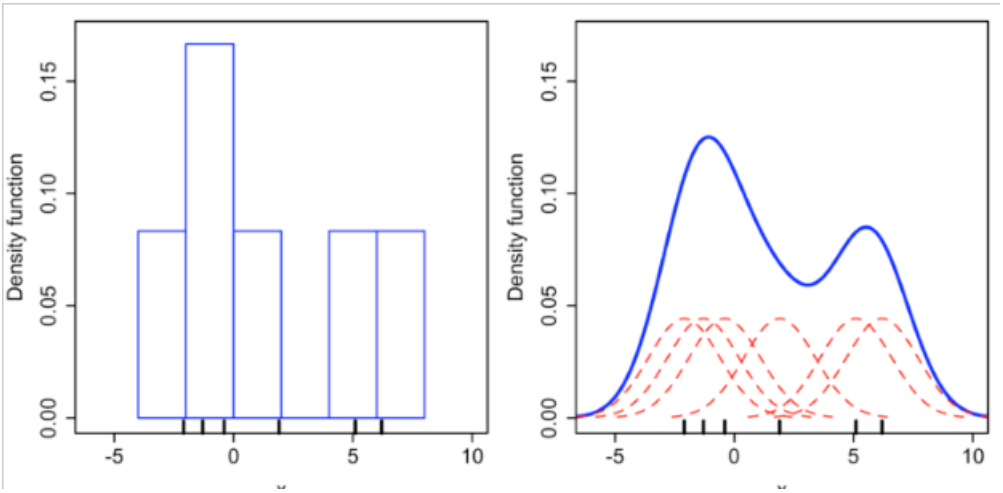
Простейший вариант непараметрического оценивания – гистограмма (рисунок 4.12, б). Для построения гистограммы область определения имеющейся выборки делится на равные интервалы, подсчитывается число точек выборки, попадающих в каждый интервал, и над каждым интервалом помещается столбик, высота которого пропорциональна числу точек в нем.

Более гладкий вид аппроксимирующей функции $\tilde{p}(x)$ дает ядерный метод непараметрического оценивания (рис. 4.12, в). Другое название – метод парзеновского окна. В этом случае над каждой точкой данных x_i помещается ядро, имеющее форму гауссоиды (показаны красными пунктирными линиями). Ядра суммируются, давая ядерную оценку плотности (сплошная синяя кривая). Выбирая форму ядра (например, величину дисперсии в случае гауссовских ядер), можно регулировать степень гладкости получаемой аппроксимации $\tilde{p}(x)$. Легко

видеть, что гистограммный метод можно рассматривать как частный случай ядерного метода с прямоугольным окном.

1	2	3	4	5	6
-2,1	-1,3	-0,4	1,9	5,1	6,2

а



б в

Рис. 4.12. а – исходная выборка; б – гистограмма; в – ядерная оценка плотности (источник [https://en.wikipedia.org/wiki/Kernel_density_estimation])

Третью группу методов непараметрического оценивания составляют различные аппроксимации функции плотности. Как правило, используют представление искомой функции $p(x)$ через ортогональную систему базисных функций, таких как многочлены Лежандра, Чебышева, Эрмита, Лагранжа, Лагерра и т.п., при этом вместо бесконечного ряда используют сумму первых k членов. Соответствующие программы можно найти в типовых статистических пакетах, например, в пакете R [Delignette-Muller].

Параметрический подход. Общий вид функции $p(x)$ известен с точностью до набора параметров, которые можно оценить по обучающей выборке.

Например, если есть основания предполагать, что оцениваемое распределение является нормальным, то оно полностью определяется своими двумя параметрами (математическим ожиданием и дисперсией), которые можно оценить по выборке X_l методом максимального правдоподобия:

$$\widetilde{\mu} = \frac{1}{l}\sum_{i=1}^{l} x_i; \quad \widetilde{\Sigma} = \frac{1}{l}\sum_{i=1}^{l} (x_i - \widetilde{\mu})(x_i - \widetilde{\mu})^T.$$

Многие сложные распределения (см. раздел 3.2) можно также описать через выборочные оценки их моментов m-го порядка:

$$\widetilde{E}[\xi^m] = \frac{1}{N}\sum_{j=1}^{N} x_j^m.$$

Восстановление смесей распределений. Если форма получающейся эмпирической плотности распределения имеет достаточно сложный вид (например, является двугорбой), то можно предположить, что она является смесью двух распределений. Для ее восстановления можно использовать EM-алгоритм [Оценивание].

Типовые методы восстановления плотности распределения по выборке имеют программную поддержку практически во всех статистических пакетах.

4.5.2. Оценка распределений на основе числовых характеристик

В реальной практике оценки плотности распределения дата-сайентист часто сталкивается с проблемами, которые в основном связаны с малым объемом имеющейся выборки и потенциально возможными отклонениями от нормального закона в сторону степенных и других длиннохвостых распределений. В этих случаях статистические методы оценки распределений работают хуже и при неадекватном применении могут своими результатами ввести в заблуждение.

Альтернативный подход заключается в идентификации закона распределения по набору числовых характеристик распределений, которые можно оценить по имеющимся выборочным данным. Конечно, такая идентификация не позволяет получить точную (в вероятностном смысле) оценку плотности распределения, но позволяет оценить их форму и тем самым сузить зону возможных распределений-кандидатов и их параметров.

Чаще всего с этой целью используются моменты распределений. По поведению первого и второго моментов можно судить о близости исследуемого распределения к нормальному. Многое о форме распределения могут сказать старшие моменты – коэффициенты асимметрии и эксцесса (см. раздел 3.3.2).

Во многих практических задачах DS оценок первых четырех моментов бывает достаточно для принятия дальнейших решений (см. раздел 4.4.7).

Очень помогает в такой оценке привлечение внешней информации об анализируемом процессе. Например, если исследуется временной ряд, характеризующий наработку оборудования на отказ, то при выборе кандидатов на распределение целесообразно принимать во внимание физику отказа: нормальный закон характеризует отказы за счет износа и процесс старения, закон распределения Вейбулла – приработочные, усталостные и внезапные отказы, а экспоненциальный закон – внезапные отказы. Это позволит существенно сократить объем выборки, необходимой для желаемой точности оценки.

В целом необходимо использовать этот подход с осторожностью, так как он имеет ряд принципиальных недостатков [Тырсин]:

– низкая точность оценок центральных моментов высоких порядков, в том числе в условиях выбросов;

– чувствительность оценок к выбираемой длине интервала при группировании данных, в частности, к параметрам гистограммы.

4.5.3. Проверка нормальности распределения

Критерии нормальности – это выделенный частный случай критериев согласия. Нормально распределённые величины часто встречаются в прикладных задачах, что обусловлено действием закона больших чисел. Если про выборки

заранее известно, что они подчиняются нормальному распределению, то к ним можно применять более мощные параметрические критерии. Проверка нормальности часто выполняется на первом шаге анализа выборки, чтобы решить, какие методы использовать далее – параметрические или непараметрические. В справочнике [Кобзарь, 2006] приведена сравнительная таблица мощности для 21 критерия нормальности.

Например, с помощью критерия асимметрии и эксцесса можно проверить гипотезу H_0: случайная величина имеет распределение, отличное от нормального. Если распределение нормально, то его коэффициент асимметрии $\alpha_3 = 0$ и коэффициент эксцесса $\alpha_4 = 0$. Так как значения $\alpha_3 = 0$ и $\alpha_4 = 0$ могут иметь место и для распределений, отличных от нормального, то этот критерий следует воспринимать как критерий установления отклонения от нормальности распределения, но не установления нормальности. При $n > 200$ можно использовать грубый критерий: если выборочная оценка коэффициента асимметрии

$$\alpha_3 = \frac{1}{ns^3}\sum_{i=0}^{n}(x_i - \tilde{x})^3 \geq \frac{6}{n},$$

то нормальность распределения отвергается. Для получения более точных оценок, а также для проверки нормальности по коэффициенту эксцесса уже требуется обращение к специальным нормализующим таблицам [Кобзарь, 2006].

4.5.4. Методы нормализации распределения

На практике часто приходится иметь дело со статистическими данными, которые по тем или иным причинам не проходят тест на нормальность. В этой ситуации есть два выхода: либо обратиться к непараметрическим методам, либо воспользоваться специальными методами, позволяющими преобразовать исходную «ненормальную статистику» в «нормальную». В этом случае вы наблюдаете за данными $Z(\mathbf{s})$ и применяете некоторое преобразование $Y(\mathbf{s}) = t(Z(\mathbf{s}))$. Обычно вы хотите найти такое преобразование, чтобы $Y(\mathbf{s})$ было нормально распределено. Часто такое преобразование также дает данные, которые имеют постоянную дисперсию в изучаемой области.

Среди множества таких методов преобразований (при неизвестном типе распределения) одним из лучших считается преобразование Бокса–Кокса:

$$Y(s) = \left(Z(s)^{\lambda} - 1\right)/\lambda \text{ при } \lambda \neq 0.$$

Например, ваши данные получены в результате фиксации величины некоторого явления по некоторой территории. Если у вас мало отсчетов в одной части изучаемой территории, дисперсия в этом регионе будет меньше, чем в другом регионе, где отсчетов выше. В этом случае целесообразно применить преобразование по методу квадратного корня:

$$Y(s) = 2\left(\sqrt{Z(s)} - 1\right),$$

которое может сделать дисперсию более постоянной на изучаемой территории, а также часто приводит данные к нормальному распределению. Преобразование по методу квадратного корня – это частный случай преобразования Бокса–Кокса с $\lambda = ½$. В целом такое (или кубическое) преобразование рекомендуется использовать при смещении данных влево.

Когда данные смещены в положительном направлении (рис. 3.30) и присутствует мало очень больших значений, то часто используется логарифмическое преобразование:

$$Y(s) = lnZ(s),$$

которое является частным случаем преобразования Бокса-Кокса с $\lambda = 0$. Если эти большие значения расположены в области наблюдения, логарифмическое преобразование поможет сделать дисперсию более постоянной и привести данные к нормальному распределению.

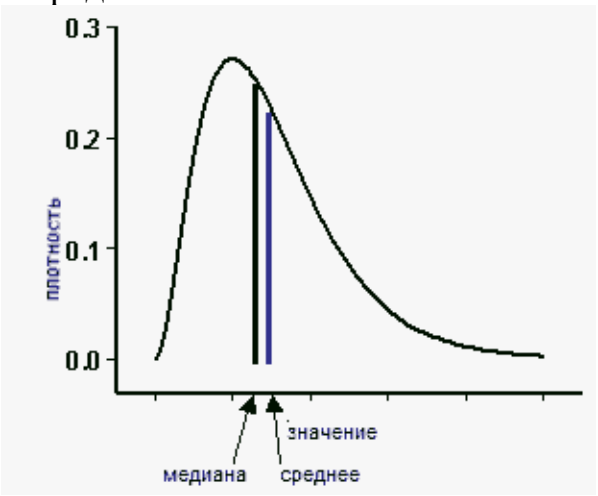


Рис. 4.13. Использование логарифмического преобразования исходных данных

В общем случае значение λ нужно подбирать таким образом, чтобы оно обеспечивало наибольшую нормальность преобразованных данных. Выбор оптимального значения параметра (лямбда) при использовании преобразования Бокса–Кокса может быть выполнен с использованием различных методов:

- Метод максимального правдоподобия. В этом подходе подбирается значение (лямбда), которое максимизирует правдоподобие модели. Это можно сделать с помощью численных методов оптимизации, таких как метод Ньютона–Рафсона или метод Брента;
- Критерии информационного критерия. Можно использовать информационные критерии, такие как критерий Акаике (AIC) или критерий Шварца (BIC);
- Кросс-валидация. при этом данные разбиваются на обучающую и проверочную выборки, и производится оценка преобразования Бокса–Кокса для различных значений (лямбда) на обучающей выборке. Затем оцениваются результаты на проверочной выборке и выбирается лучшее значение.

Необходимо помнить об ограничениях преобразования Бокса–Кокса:

- оно плохо подходит для отрицательных значений, так как требует возведения отрицательных значений в степень;
- используемые данные должны быть непрерывными.

4.5.5. Графические методы оценки распределений, отличных от нормального

Работа с распределениями, отличными от нормального, требует повышенного внимания дата-сайентиста. Как отмечалось в разделе 3.3.3, «особо опасными» в этом смысле являются распределения, близкие к степенному закону. Практически всегда обнаружение и характеристика степенных законов осложнены большими флуктуациями данных в хвосте распределения (где происходят крупные, но редкие события), и трудностью определения диапазона, в котором степенной закон сохраняется. Обычные методы анализа, такие как подгонка методом наименьших квадратов, для степенного закона распределений могут давать существенно неточные оценки параметров. Более того, они, как правило, не дают указаний на то, подчиняются ли данные степенному закону вообще.

Поэтому дата-сайентисту в работе с распределениями, потенциально близкими к степенному закону, нужно быть аккуратным и внимательно читать литературу, например, [Cruasett]. Для первых прикидок традиционно используют графические методы – квантиль-квантильный график (Q–Q plot) и график в двойном логарифмическом масштабе.

График квантиль-квантиль позволяет проверить следующие гипотезы:

- для заданной случайной величины X, имеющей среднее и стандартное отклонение, – является ли она гауссовой или нет;
- для любых двух случайных величин – имеют ли они одинаковое распределение или нет.

Чтобы создать квантильный график для проверки первой гипотезы, нужно:

- расположить оцениваемые данные в порядке возрастания;
- вычислить их квантили, расположить их по оси Y;
- вычислить теоретические квантили гауссовского распределения, расположить их по оси X.
- построить график зависимости соответствующих квантилей.

Если точки построенного графика приблизительно соответствуют прямой линию, то экспериментальные данные распределены по Гауссу. S-образная кривая указывает на тяжелые хвосты, а перевернутая S-образная кривая – указывает на легкие хвосты.

В идеальном случае, т.е. если исследуемый датасет достаточно большой и данные не зашумлены, полученные графики могут выглядеть так, как показано на рис. 4.14. В реальных случаях зашумленных данных и малых выборок графики становятся менее выраженными (рис. 4.15). Соотношение типичных квантиль-квантильных графиков и соответствующих функций распределения проиллюстрировано на рис. 4.16, где trunknorm – усеченное нормальное распределение с «обрезанными» хвостами.

Проверка эквивалентности двух распределений, т.е. справедливость второй гипотезы, выполняется аналогично, только вместо нормального распределения используется одно из сравниваемых распределений.

Средства построения квантиль-квантильных графиков представлены практически во всех статистических пакетах.

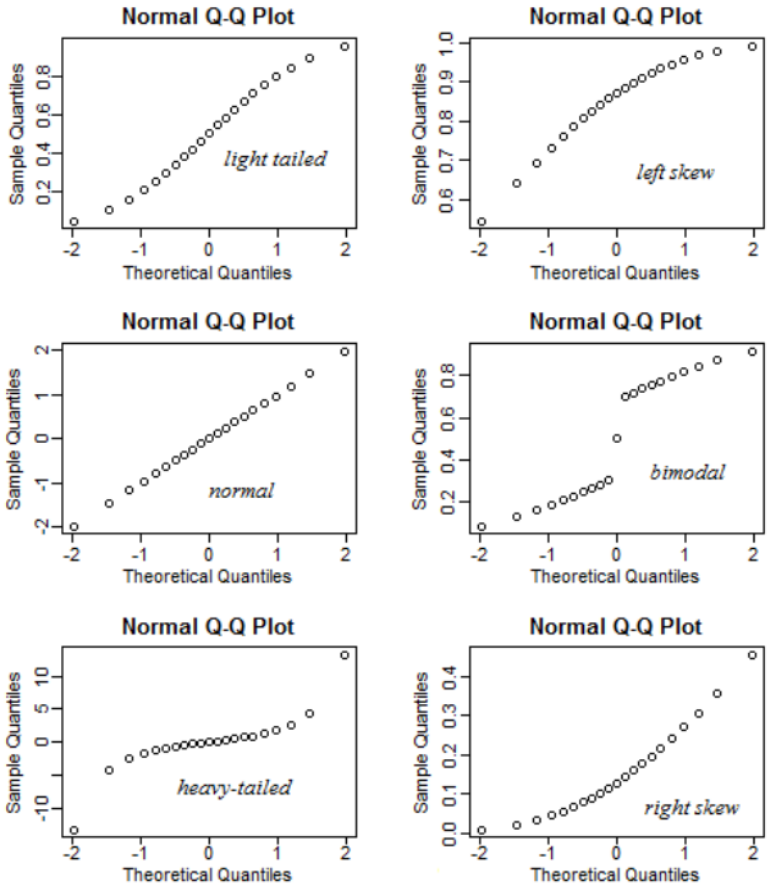


Рис. 4.14. Квантиль-квантильные графики для идеального случая

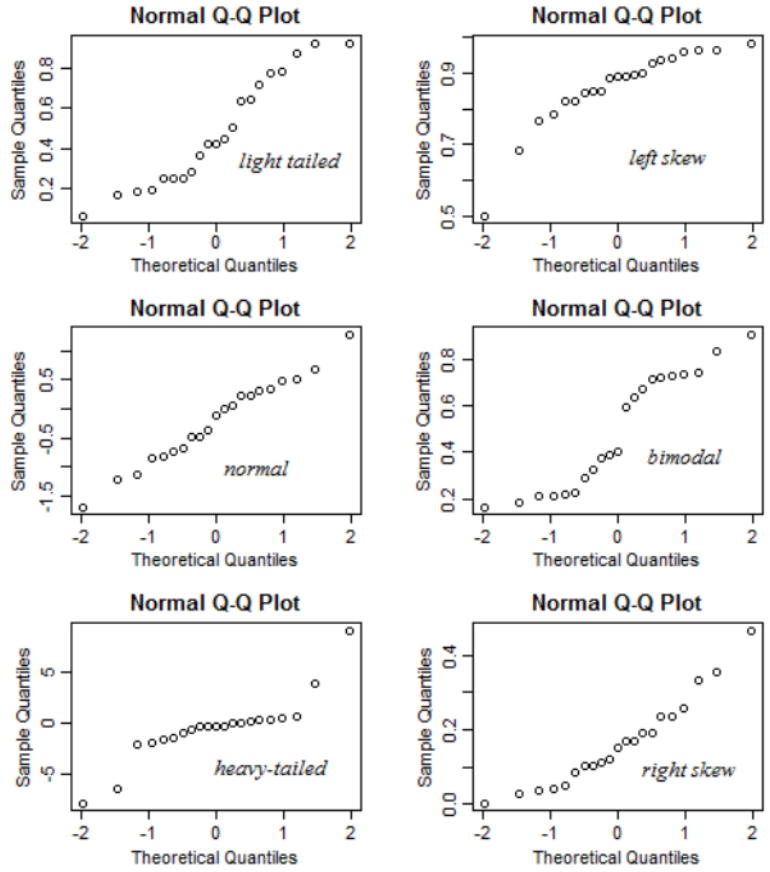


Рис. 4.15. Квантиль-квантильные графики для реального случая

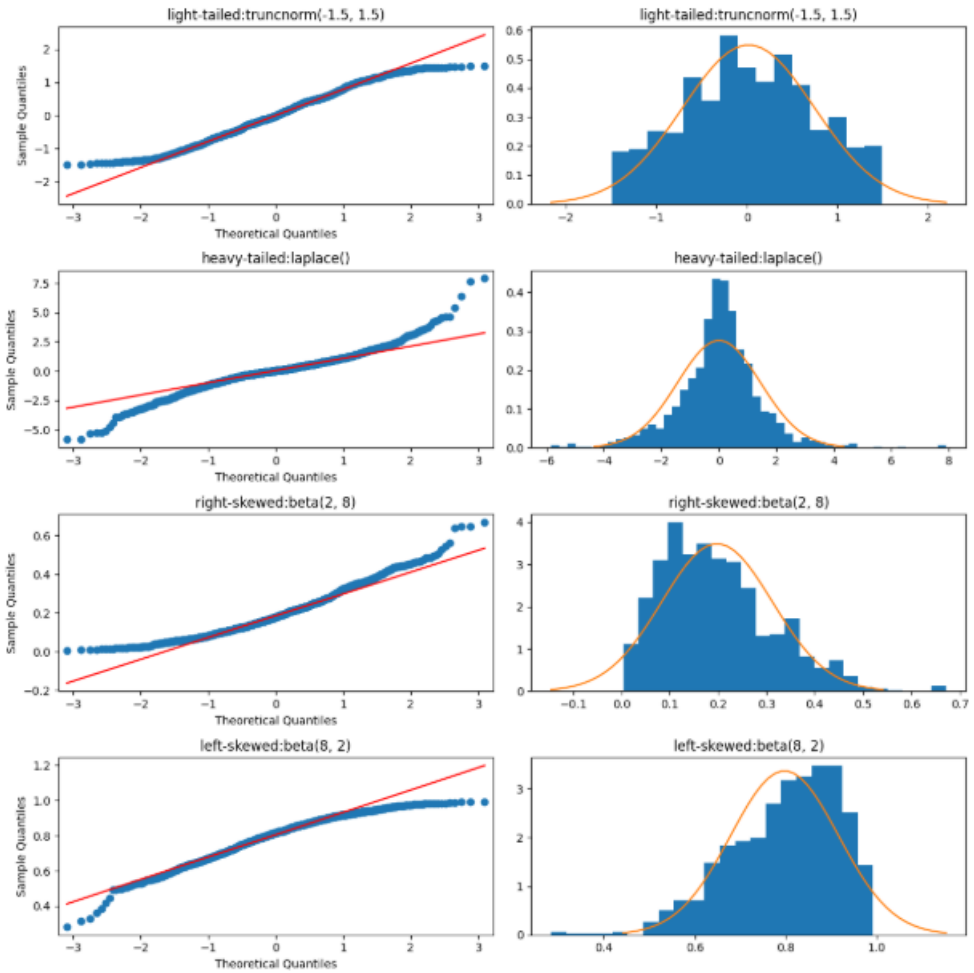


Рис. 4.16. Соотношение функций распределения и квантиль-квантильных графиков

Логарифмические графики направлены непосредственно на выявление степенной зависимости. Если взять логарифм от степенной функции

$$f(x) = ax^k,$$

то получим выражение

$$logf(x) = log\ a\ + k\ logx,$$

график которого есть в двойном логарифмическом масштабе (рис. 4.17) представляет собою прямую линию с наклоном k.

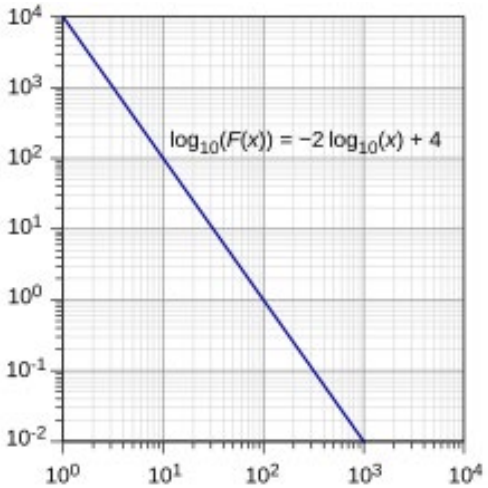


Рис. 4.17. Двойной логарифмический график для степенного закона

К сожалению, прямолинейный характер этого графика является необходимым, но недостаточным доказательством степенной зависимости, поскольку многие нестепенные распределения также отображаются в виде прямых линий на логарифмическом графике. Недостатками логарифмических графиков является то, что они требуют больших объемов данных для надежной оценки характера распределения, а также подходят только для дискретных (или сгруппированных) данных.

4.5.6. Типовая практика работы с распределениями

В практике работы дата-сайентиста необходимость восстановления функции распределения возникает в различных контекстах. Иногда (хотя и достаточно редко) она выступает как самостоятельная задача – например, если это требуется в статистическом отчете. Иногда оценку распределений формируют как предварительную задачу, чтобы, например, обосновать выбранный статистический тест. Иногда требуется оценить не полное распределение, а его характерные фрагменты (например, оценить тяжесть хвоста или сопоставить два распределения). Для каждого такого случая дата-сайентисту необходимо выбирать адекватные подходы и метрики.

Примеры выбора типа критерия для решения задачи о репрезентативности выборки, приведенные в разделе 3.1.2, показывают, что не существует общепризнанного подхода для решения конкретной задачи – хотя, как показывает опыт применения критерия p-value в медицине, это бывает очень трудно доказать заказчику! Поэтому главное для дата-сайентиста при выборе подхода к оценке распределения – тщательно анализировать задачу на качественном уровне:

- что именно мы хотим получить – вид распределения, средние значения, дисперсии иди др.;
- какие типы данных мы обрабатываем – непрерывные, дискретные, категориальные иди др.;
- каковы механизмы появления распределения; например, при наличии в анализируемом процессе неравновесных и нелинейных явлений можно ожидать степенной закон распределения [Farmer, Hilbert];
- к какой группе может принадлежать распределение (например, нормальное – скошенное – с длинным / тяжелым хвостом; степенное – экспоненциальное; альфа-семейство; и др.), какие характеристики можно оценивать для него (например, для многих распределений не существуют моменты определенного типа – см. раздел 3.3);
- что мы знаем о форме распределения, например:
    - исследуемый показатель слишком часто принимает минимальное или максимальное значение – тогда распределение скошено вправо или влево;
    - из теоретических знаний об исследуемом показателе следует, что его распределение должно быть нормальным;

- есть ли основания ожидать выбросов, могут ли они указывать на наличие тяжелого хвоста;

и т.п.

4.6. Визуализация в разведочном анализе

4.6.1. Цели и особенности визуализации данных

Визуализация данных – это процесс графического представления данных. Перевод данных в визуальный контекст обеспечивает их интегральное представление, которое помогает человеку в анализе данных:

– упрощает восприятие больших массивов данных со сложной структурой;
– иллюстрирует корреляции и связи, помогает сравнивать и находить различия, выявлять тренды и аномалии;
– акцентирует внимание на нужных аспектах;
– удерживает внимание потребителя информации.

Можно выделить следующие особенности визуализации данных.

Визуализация данных используется для облегчения распознавания закономерностей или тенденций в данных. Отсюда очевидно, что визуализация данных должна выполняться в расчете на конкретного адресата: то, что просто для понимания десятилетним ребенком, может быть не таким уж простым для обладателя докторской степени (и наоборот!); потребителю целесообразно донести совсем не ту информацию о продукте, которая нужна его разработчику.

Визуализация данных используется для поддержки эффективного общения, так как создает единый контекст для общения. Эта особенность является продолжением первой, но применительно ко всей целевой группе одновременно работающих относительно тех, кто может их просматривать. Генеральный директор может предпочесть просматривать идеи, которые предоставляют действенные шаги, а команда машинного обучения может предпочесть просматривать идеи о том, как работают их модели.

Целью визуализации данных может быть либо собственно исследование данных, либо объяснение результатов заказчику, и для реализации каждой цели используются свои средства. Исследовательская графика создается, как правило, «для внутреннего пользования» командой специалистов и призвана поддержать их креативные решения. Презентационная графика, напротив, служит «демонстрации товара»: ее целью является подведение ЛПР к нужным выводам в задаче. Здесь, помимо собственно информационной составляющей, важно построение сюжета (story-telling) всей презентации и ее эстетическое оформление.

4.6.2. Основные методы визуализации данных в разведочном анализе

Линейные графики (line chart, curve chart) (рис. 4.18) – один из наиболее часто используемых типов визуализаций. Линейные графики отлично подходят для отслеживания эволюции переменной с течением времени. Обычно временная переменная располагается по оси x, а анализируемая переменная – по оси y.

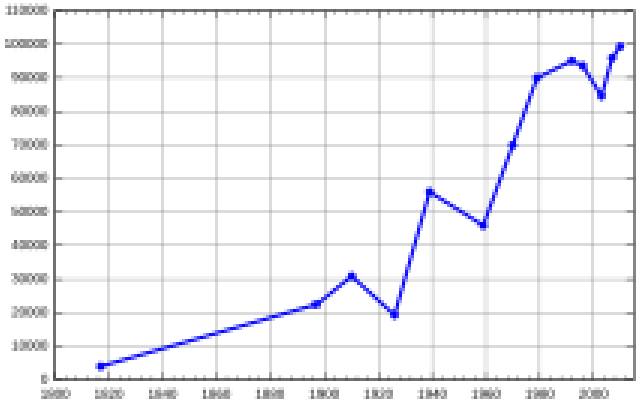


Рис. 4.18. Линейный график: рост населения г. Пушкин (Санкт-Петербург) [https://en.wikipedia.org/wiki/Line_chart]

Столбчатая диаграмма (bar plot, bar chart) (рис. 4.19) ранжирует данные по категориям. Она состоит из прямоугольников, длина которых пропорциональна значению каждой категории. Существует несколько типов столбчатых диаграмм, каждый из которых подходит для определенной цели, включая вертикальные столбчатые диаграммы, горизонтальные столбчатые диаграммы и кластеризованные столбчатые диаграммы.

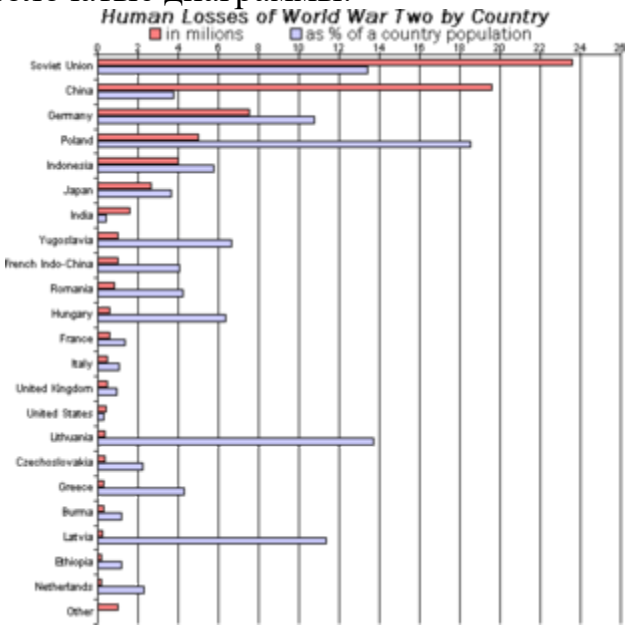


Рис. 4.19. Столбчатая диаграмма: потери населения во Второй мировой войне [https://en.wikipedia.org/wiki/ Bar_chart]

Гистограмма (histogram) (рис. 4.20) – это визуальное представление распределения количественных данных. Для построения гистограммы первым шагом является «бинирование» (или «разбиение на сегменты») диапазона значений, т.е. разделение всего диапазона значений на ряд интервалов, и затем подсчет количества значений, попадающих в каждый интервал. Бины (интервалы) являются смежными и обычно (но не обязательно) имеют одинаковый размер. Гистограммы дают грубое представление о плотности базового распределения данных и часто используются для оценки плотности вероятности базовой переменной.

Интервал	Число значений
–3.5…–2.51	9
–2.5…–1.51	32
–1.5…–0.51	109
–0.5…0.49	180
0.5…1.49	132
1.5…2.49	34
2.5…3.49	4

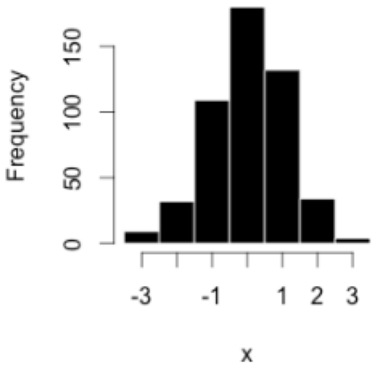


Рис. 4.20. Гистограмма: слева – распределение данных по интервалам, справа – гистограмма [https://en.wikipedia.org/wiki/ Bar_chart]

Диаграммы «ящик с усами» (Box and whisker plots) (рис. 4.21) наглядно представляют квантильное описание распределения величин в наборе данных:

- медиана (50% данных меньше медианы и 50% данных больше медианы);
- верхний квартиль (75% данных меньше верхнего квартиля, а 25% данных больше верхнего квартиля);
- нижний квартиль (25% данных меньше нижнего квартиля, а 75% выше нижнего квартиля);
- межквартильный размах (верхний квартиль минус нижний квартиль);
- «максимум» (верхний квартиль плюс 1,5-кратный межквартильный размах);
- «минимум» (нижний квартиль минус 1,5-кратный межквартильный размах);
- выбросы (любые значения выше «максимума» или ниже «минимума»).

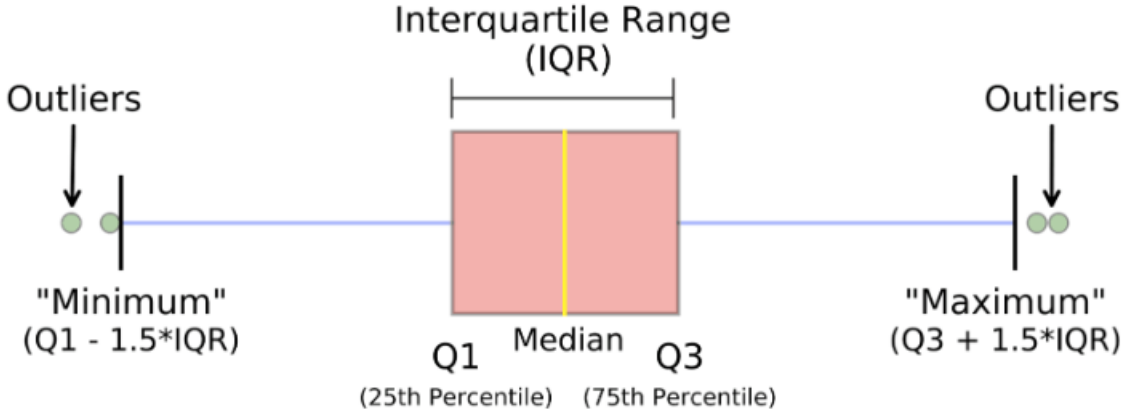


Рис. 4.21. Схема построения диаграммы «ящик с усами» [https://www.datacamp.com/blog/data-visualization-techniques]

Диаграммы рассеяния (scatter plots) (рис. 4.22) используются для визуализации взаимосвязи между двумя непрерывными переменными. Каждая точка на диаграмме представляет собой отдельный экземпляр данных с значениями признаков *x* и *y* соответственно. Диаграммы рассеяния используются для быстрого выявления потенциальных корреляций.

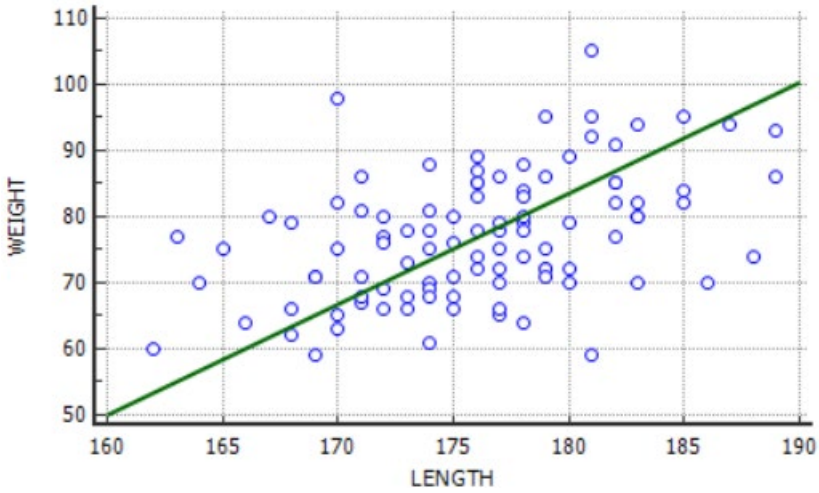


Рис. 4.22. Диаграмма рассеяния для соотношения роста и веса в группе больных; показана главная линия регрессии [https://www.medcalc.org/manual/scatter-diagram.php]

Пузырьковая диаграмма (Bubble plot) (рис. 4.23) расширяет возможности диаграммы рассеяния за счет добавления новых координат – цвета и размера точек.

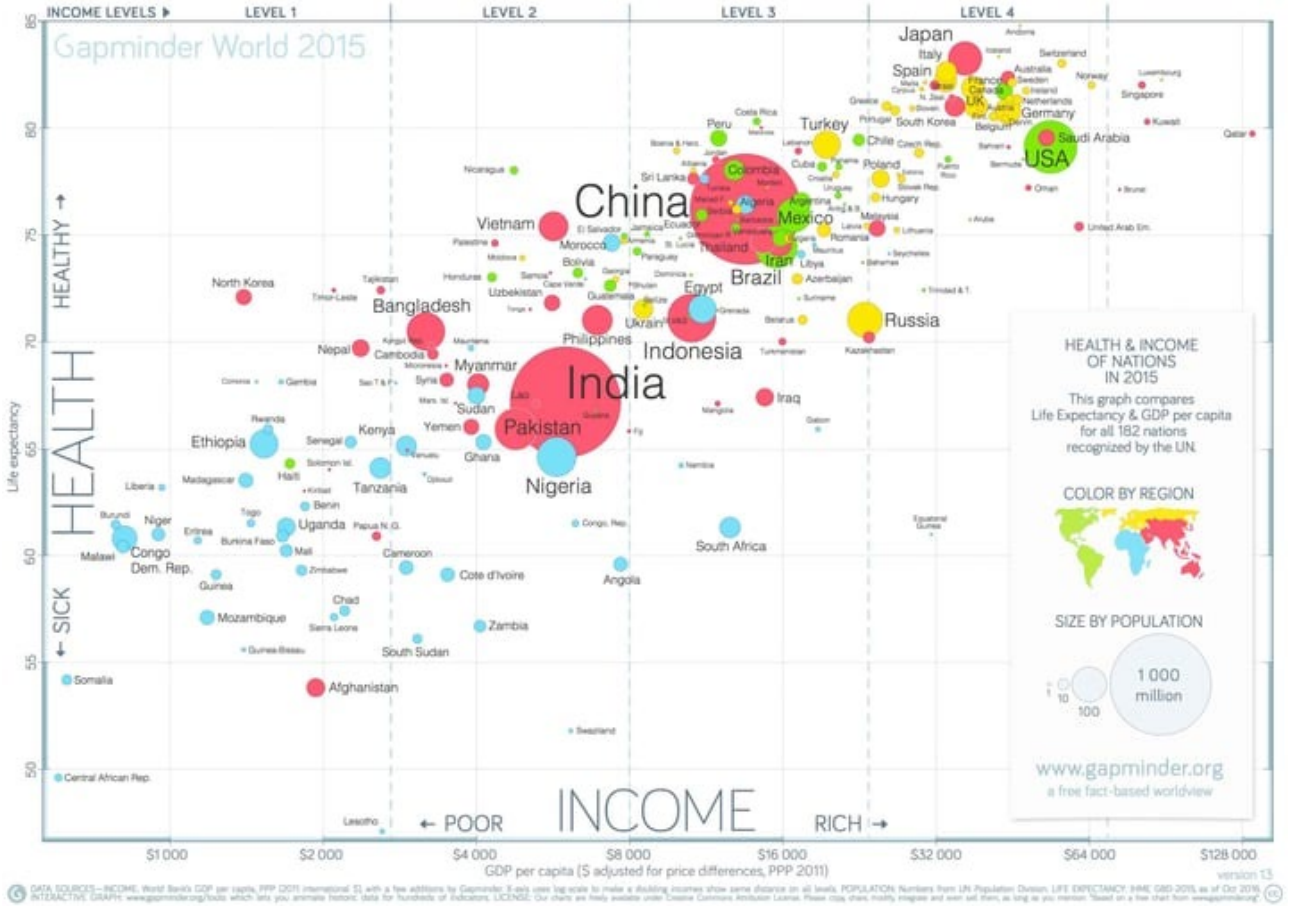


Рис. 4.23. Пузырьковая диаграмма, иллюстрирующая соотношение здоровья и доходов населения стран мира [https://www.gapminder.org/tag/gapminder-world/]

Древовидные карты (treemaps), **круговые диаграммы** (pie charts) (рис. 4.24) используются для отображения отношений «часть-целое» в данных.

Они представляют иерархические данные в виде набора фрагментов (прямоугольников или секторов), причем площадь каждого фрагмента пропорциональна размеру этой категории данных. Древовидные карты считаются более интуитивными и предпочтительными по сравнению с круговыми диаграммами.

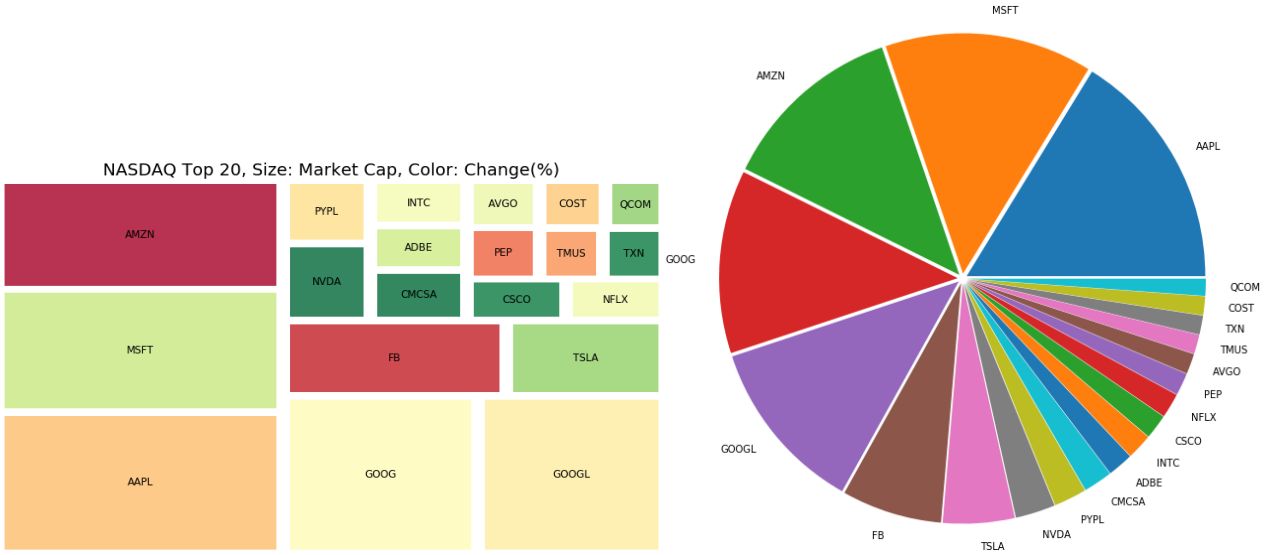


Рис. 4.24. Доля крупнейших компаний в биржевом индексе NASDAQ [https://medium.com/nerd-for-tech/how-treemaps-are-better-than-pie-chart-5f8709057fbc]

Тепловые карты (heatmaps) отображают степень связи между двумя переменными в виде цветового кода. Например, визуализация дифференциальной экспрессии генов (рис. 4.25) показывает, насколько каждый ген увеличивает или уменьшает свою активность в раковой ткани по сравнению со здоровой тканью. Дендрограммы по бокам показывают результаты кластеризации тканей и генов на основе их сходства, что дополнительно упрощает анализ.

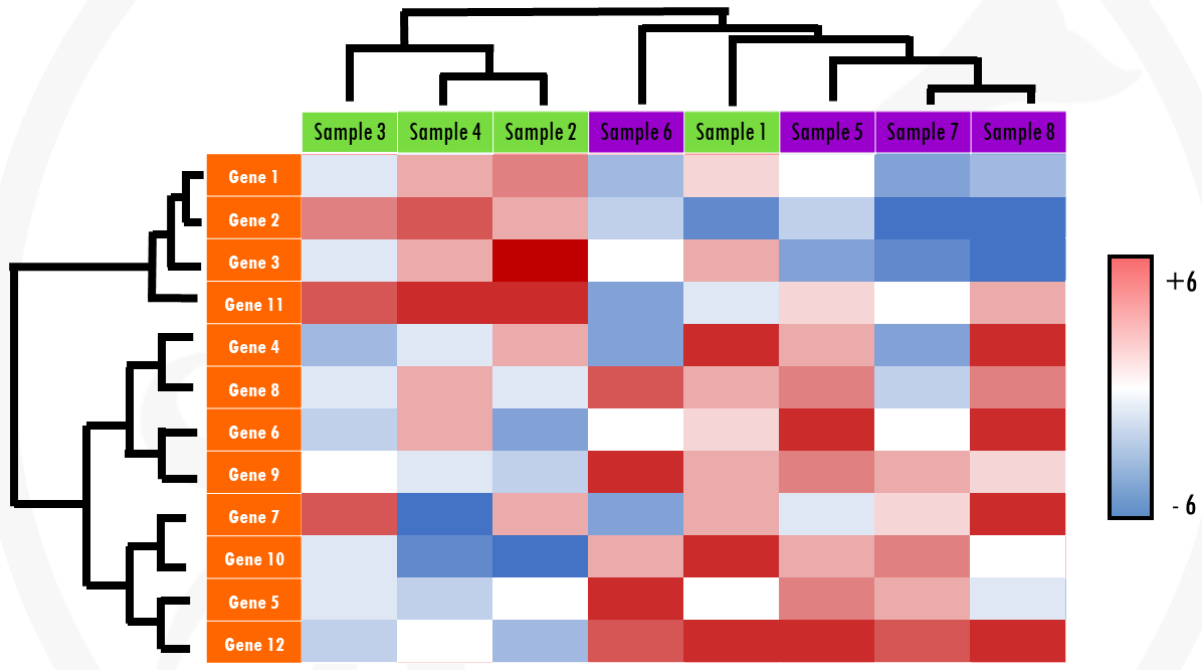


Рис. 4.25. Тепловая карта экспрессии генов в здоровых и больных тканях (источник [https://biostatsquid.com/heatmaps-simply-explained/])

Облака слов (word clouds) компактно показывают самые частые слова в тексте или наборе данных. Они считаются более визуально привлекательными, но более сложными для интерпретации по сравнению с обычными столбчатыми диаграммами. Облака слов полезны в презентационных целях, так как позволяют наглядно донести наиболее важные темы и ключевые концепции, а также показать общее настроение текста. Например, на рис. 4.26 показано облако слов, построенное по результатам оценки наиболее значимых потребительских характеристик вин из более чем 40 стран мира.

Рис. 4.26. Потребительские характеристики вин (по результатам [https://www.kaggle.com/datasets/zynicide/wine-reviews/data])

Карты (maps) – широко известный способ отображения данных с привязкой на местности. Современные карты могут нести богатую геопространственную информацию: например, карты высокой четкости [Xiao] (рис. 4.27) содержат не только точное геометрическое описание элементов дороги, но и семантическую информацию более высокого уровня, такую как топология дороги, тип полосы движения, ограничение скорости и т.д., представленную облаками точек.

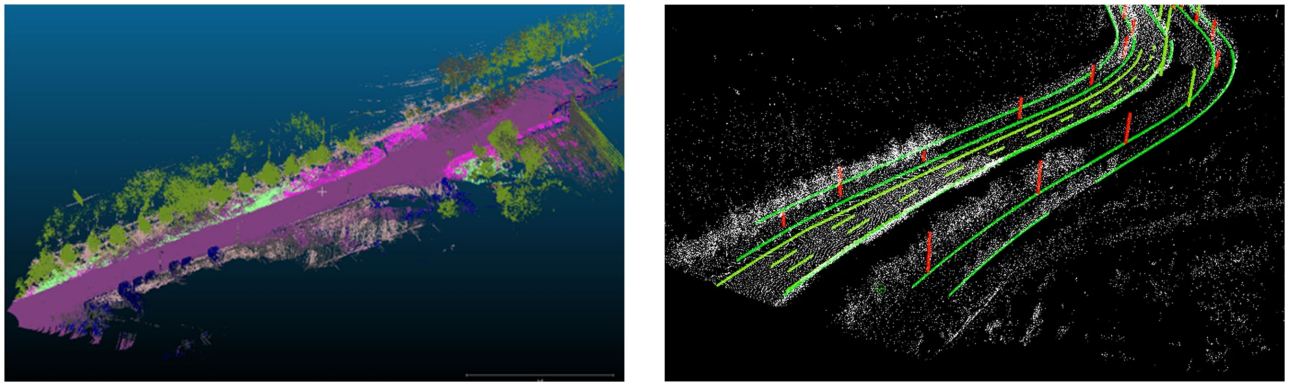


Рис. 4.27. Различные типы карт высокой четкости (HD) — (a) карта облака точек с семантическими метками; (b) векторная HD-карта типичной городской дороги

Сетевые диаграммы (network diagrams), представляя данные в виде графов, обеспечивают богатые возможности для их интерпретации. Например, граф на рис. 4.28 [Cui] иллюстрирует механизм конкуренции между различными моделями автомобилей с точки зрения поведения потребителей: какие автомобили рассматриваются вместе и какие автомобили в конечном итоге покупа-

ются клиентами. Направленная связь от узла u к узлу v означает, что существуют клиенты, которые рассматривали автомобили u и v вместе, но в итоге купили v вместо u. Толщина связи между двумя узлами пропорциональна ее силе (т.е. количеству клиентов, которые рассматривают или выбирают продукт), а размер узла пропорционален популярности продукта. Датасет построен по опросам клиентов с 2012 по 2016 гг. на рынке Китая (более 40 000 респондентов каждый год), показан фрагмент графа для спортивно-утилитарных (SUV) и аналогичных автомобилей.

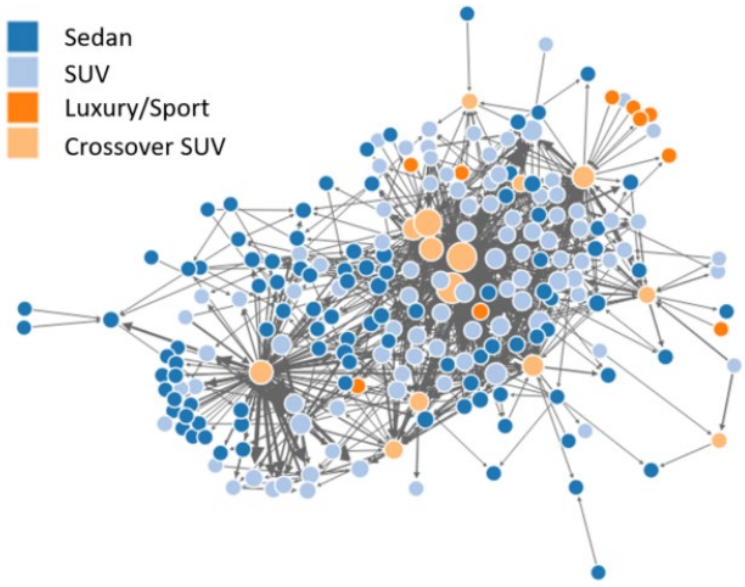
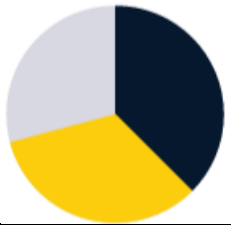


Рис. 4.28. Сетевая диаграмма конкуренции на рынке автомобилей

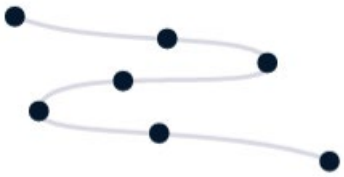
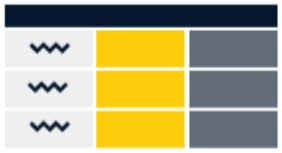
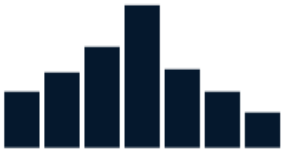
4.6.3. Выбор метода визуализации данных в разведочном анализе

Нижеследующая таблица 4.6 [Cotton] дает примеры применения инструментов визуализации в различных задачах DS.

Таблица 4.6. Примеры применения инструментов визуализации в задачах DS

Тип диаграммы	Сфера применения	Примеры
Способы отображения отношения «часть-целое»		
Круговая диаграмма 	Один из наиболее распространенных способов отображения данных «часть к целому». Также обычно используется с процентами	Предпочтение избирателей по возрастным группам Рыночная доля облачных провайдеров
Круговая диаграмма «пончик» 	Вариант круговой диаграммы, имеет в центре отверстие для удобства чтения	Доля рынка ОС Android Ежемесячные продажи по каналам

Тепловые карты 	Двухмерные диаграммы, которые используют цветовую заливку для представления тенденций данных	Средние месячные температуры в течение года Отделы с наибольшей текучестью кадров с течением времени
Столбчатая диаграмма с накоплением 	Лучше всего подходит для сравнения подкатегорий в категориальных данных. Также может использоваться для сравнения процентов	Квартальные продажи по регионам Общие продажи автомобилей по производителям
Диаграммы «древовидная карта» 	Двумерные прямоугольники, размер которых пропорционален измеряемому значению и может использоваться для отображения иерархически структурированных данных	Количество продаж продуктов питания с категорией Сравнение цен на акции по отраслям и компаниям
Способы фиксации тренда		
Линейная диаграмма 	Самый простой способ зафиксировать, как числовая переменная изменяется с течением времени	Потребление энергии в кВтч с течением времени Поисковые запросы с течением времени
Мультилинейная диаграмма 	Захватывает несколько числовых переменных с течением времени. Может включать несколько осей, позволяя сравнивать различные единицы и диапазоны масштабов	Сравнение стоимости акций Apple и Amazon с течением времени Поисковые запросы Lebron и Steph Curry с течением времени
Диаграмма областей 	Показывает, как изменяется числовое значение, закрашивая область между линией и осью X.	Всего продаж за все время Активные пользователи за все время
Диаграмма с накоплением областей 	Наиболее часто используемая разновидность диаграмм с областями для отслеживания разбивки числового значения по подгруппам	Активные пользователи с течением времени по сегментам Общий доход с течением времени по странам

Сплайн-диаграмма 	Сглаженная версия линейной диаграммы. Отличается тем, что точки данных соединены сглаженными кривыми для учета пропущенных значений, а не прямыми линиями	Потребление электроэнергии с течением времени Выбросы CO2 с течением времени
Способы визуализации одного значения		
Карточка 	Отлично подходят для отображения и отслеживания ключевых показателей эффективности на дашбордах или презентациях	Доход на текущий момент на панели продаж Общее количество регистраций после акции
Таблица диаграмм 	Лучше всего использовать для небольших наборов данных, отображает табличные данные в единой таблице	Таблица лидеров среди руководителей аккаунтов Регистрации на вебинар
Диаграмма оценки 	Часто используется на дашбордах для руководителей, чтобы показать соответствующие ключевые показатели эффективности (KPI).	Уровень лояльности потребителей к компании Текущий доход по сравнению с целевым
Способы отображения распределений		
Гистограмма 	Показывает распределение переменной, преобразуя числовые данные в ячейки в виде столбцов. Ось x показывает диапазон, а ось y представляет частоту	Распределение зарплат в организации Распределение роста в одной когорте
Диаграмма «ящик с усами» 	Показывает распределение переменной с использованием 5 ключевых сводных статистик — минимум, первый квартиль, медиана, третий квартиль и максимум	Эффективность топлива транспортного средства Время, потраченное на чтение разными читателями
Скрипичная диаграмма	Разновидность ящичной диаграммы. Она также	Время, потраченное в ресторанах по возраст-

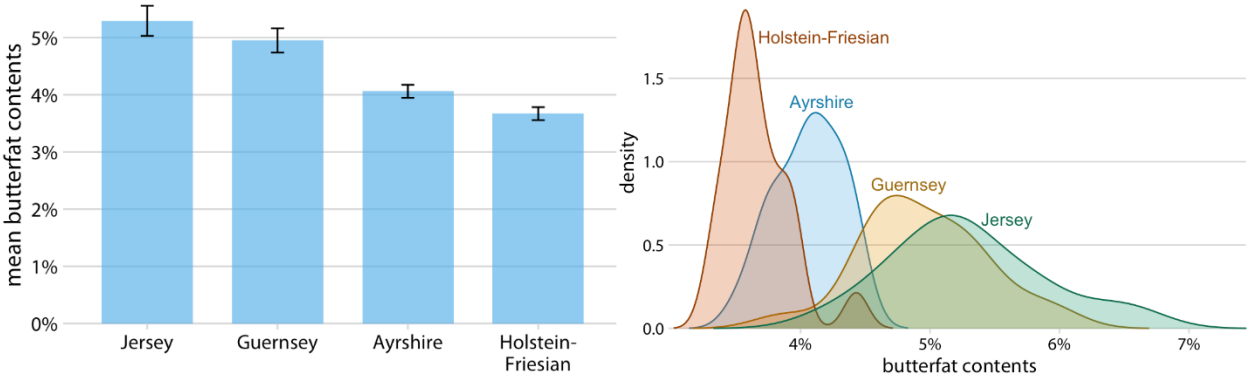
	показывает полное распределение данных вместе со сводной статистикой	ной группе Продолжительность действия таблеток по дозе
График плотности распределения 	Визуализирует распределение с помощью сглаживания, чтобы обеспечить более плавное распределение и лучше зафиксировать форму распределения данных	Распределение цены на номера в отелях Сравнение оценок лояльности компании по сегментам клиентов
Способы визуализации взаимосвязей		
Столбчатая диаграмма 	Одна из самых простых для чтения диаграмм, которая помогает быстро сравнивать категориальные данные. Одна ось содержит категории, а другая ось представляет значения	Объем поисковых запросов Google по регионам Доля рынка в доходах по продуктам
Вертикальная столбчатая диаграмма 	Категории размещены на оси x. Предпочтительнее столбчатых диаграмм для коротких меток, диапазонов дат или отрицательных значений	Доля рынка бренда Анализ прибыли по регионам
Диаграмма рассеяния 	Наиболее часто используемая диаграмма при наблюдении за взаимосвязью между двумя переменными. Она особенно полезна для быстрого выявления потенциальных корреляций между точками данных	Отображение взаимосвязи между временем, проведенным на платформе, и оттоком клиентов. Отображение взаимосвязи между зарплатой и годами, проведенными в компании
Связанная диаграмма рассеяния 	Гибрид между диаграммой рассеяния и линейной диаграммой, точки рассеяния соединены линией	Цена криптовалюты. Визуализация временных шкал и событий при анализе двух переменных
Пузырьковая диаграмма	Часто используется для визуализации точек	Взаимосвязь между CPC (cost per click), конвер-

	данных с 3 измерениями, а именно, визуализируется по оси x, оси y и с размером пузыря. Показывает взаимосвязи между точками данных с использованием местоположения и размера	сией и долей от общей конверсии Связь между ожидаемой продолжительностью жизни, ВВП на душу населения и численностью населения
Диаграмма облака слов 	Удобный инструмент для визуализации наиболее распространенных слов, которые встречаются в тексте	100 самых употребляемых клиентами слов в тикетах службы поддержки клиентов
Способы визуализации потока		
Диаграмма Сэнки 	Полезна для представления потоков в системах. Этот поток может быть любой измеримой величиной	Поток энергии между странами Объемы цепочки поставок между складами
Хордовая диаграмма 	Полезна для представления взвешенных отношений или потоков между узлами. Особенно полезна для выделения доминирующих или важных потоков	Экспорт между странами для демонстрации крупнейшего экспортного партнера Объемы цепочки поставок между крупнейшими складами
Сетевая диаграмма 	Иллюстрирует, как разные элементы связаны друг с другом	Как связаны разные аэропорты по всему миру Анализ групп друзей в социальных сетях

4.6.4. Визуализация неопределенности в оценке данных

Как отмечалось в разделе 4.3.2, в разведочном анализе часто используется подход описательной статистики, когда статистические показатели, построенные на выборочных оценках, необходимо распространить на всю генеральную совокупность. С этой целью широко используется визуализация, которая помогают помогает понять, насколько правомерно такое обобщение.

На рис. 4.29, а, в виде столбчатой диаграммы показано среднее содержание молочного жира в молоке четырех пород крупного рогатого скота. Планки погрешностей вверху каждого столбца указывают +/- одну стандартную ошибку среднего значения. Для сравнения на рис. 4.29, б, показано распределение содержания молочного жира внутри этих же пород скота.



а б

Рис. 4.29. Визуализация содержания молочного жира в молоке различных пород крупного рогатого скота: а – среднее содержание молочного жира и его стандартная ошибка; б – плотность распределения содержания молочного жира (источник [https://clauswilke.com/dataviz/histograms-density-plots.html])

Визуализации типа рис. 4.29, а, часто встречаются в научной литературе. Хотя они технически верны, их информативность невысока: они не представляют ни изменчивости внутри каждой категории, ни неопределенности выборочных средних значений.

Устранить эти недостатки призван переход от точечной к интервальной оценке параметров, т.е. использование доверительных интервалов (confidence intervals). Доверительным называется интервал, в который попадают измеренные в эксперименте значения, соответствующие доверительной вероятности [Гмурман]. В свою очередь, доверительной вероятностью называется вероятность, с которой данные, полученные в условиях конкретного эксперимента, можно считать надежными (достоверными). Обычно используют значения доверительной вероятности 99%, 95%, 80%.

Интуитивное толкование доверительного интервала состоит в следующем: если уровень доверия велик (скажем, 0,95 или 0,99), то доверительный интервал почти наверняка содержит истинное значение искомого параметра.

Алгоритмы оценки доверительного интервала для различных условий можно найти в литературе [Справочник, Кобзарь]. Принципиально важно, что на ширину доверительного интервала влияет не только выбранный уровень достоверности, но и размер выборки, и ее изменчивость. При прочих равных условиях, большая выборка дает более узкий доверительный интервал, большая изменчивость в выборке дает более широкий доверительный интервал, а более высокий уровень достоверности дает более широкий доверительный интервал.

Различные варианты визуализации мер изменчивости показаны на рисунке 4.30. Исходная выборка представляет собой экспертные оценки 125 шоколадных батончиков от производителей в Канаде по шкале от 1 (неприятный) до 5 (элитный). Планки погрешностей указывают сверху вниз удвоенное стандартное отклонение, удвоенную стандартную ошибку (стандартное отклонение среднего) и 80%, 95% и 99% доверительные интервалы среднего.

Напомним, что стандартное отклонение (standard deviation) отражает изменчивость в пределах выборки, тогда как стандартная ошибка (standard error)

оценивает изменчивость между выборками в популяции. Стандартная ошибка приблизительно определяется как стандартное отклонение выборки, деленное на квадратный корень из размера выборки.

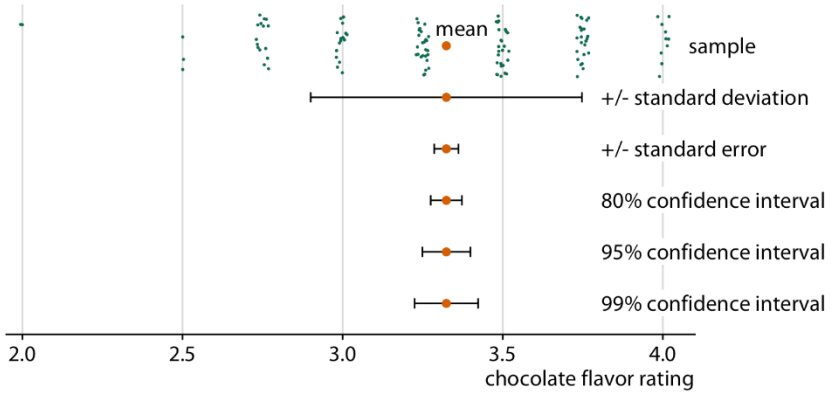


Рис. 4.30. Связь между выборкой, средним значением выборки, стандартным отклонением, стандартной ошибкой и доверительными интервалами на примере оценок шоколадных батончиков
(источник [https://clauswilke.com/dataviz/visualizing-uncertainty.html])

Более крупные выборки, как правило, имеют более узкие стандартные ошибки и доверительные интервалы, даже если их стандартное отклонение одинаково.

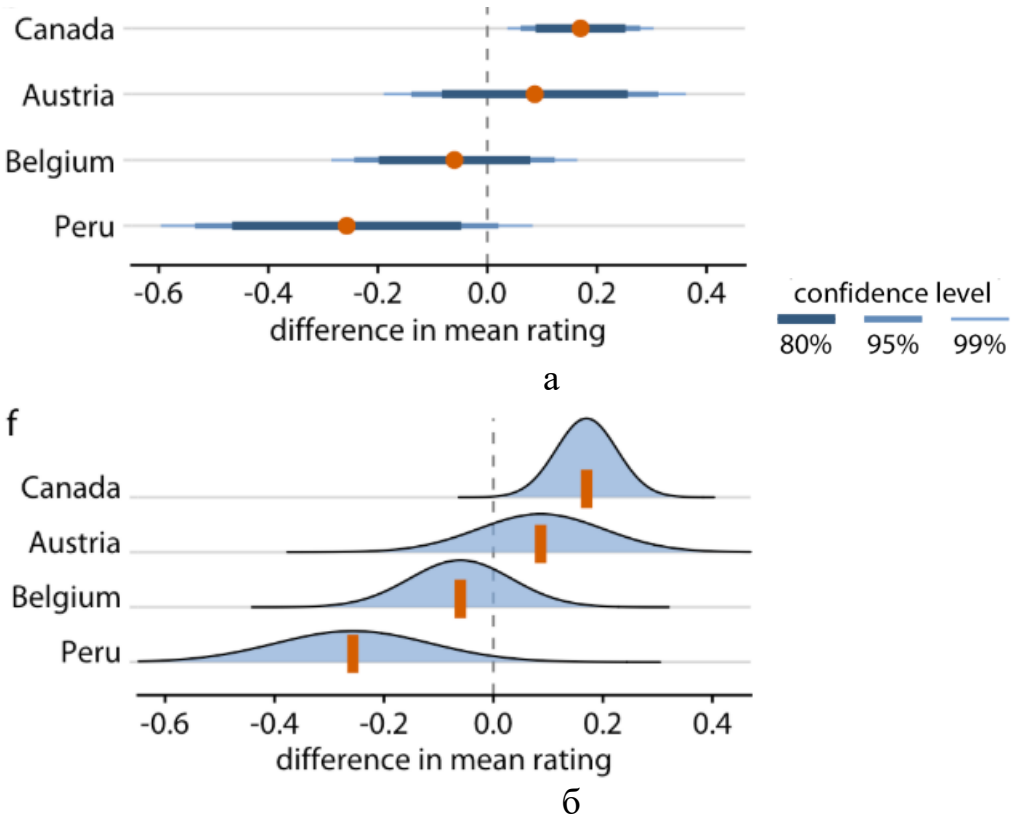


Рис. 4.31. Средние оценки вкуса шоколада для производителей из четырех разных стран относительно средней оценки плиток шоколада из США: а – доверительные интервалы с градуировкой по значениям; б – доверительные распределения [https://clauswilke.com/dataviz/visualizing-uncertainty.html]

Интегральное понимание поведения оцениваемого параметра внутри доверительного интервала обеспечивает доверительное распределение (confidence distribution), которое представляет доверительные интервалы всех уровней для интересующего параметра. Подчеркнем, что доверительное распределение НЕ является функцией распределения вероятностей интересующего параметра.

На рис. 4.31 представлены визуализации доверительного интервала для трех значений уровня доверия (рис. 4.31, а) и в виде доверительного распределения (рис. 4.31, б). Однако даже во втором случае трудно визуально интегрировать область под кривой и определить, где именно достигается заданный уровень уверенности, поэтому явное предпочтение тому или иному способу визуализации доверительных интервалов не отдается.

Визуализацию доверительного интервала можно также использовать в двумерном случае (рис. 4.32).

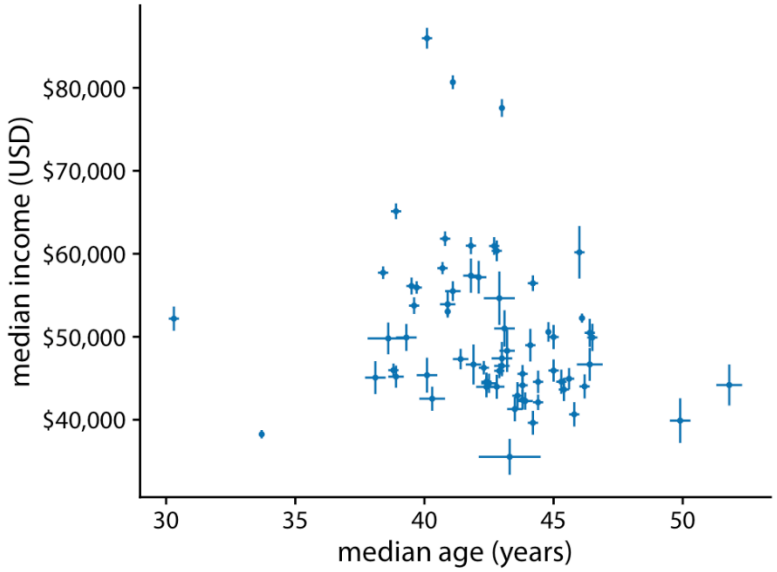


Рис. 4.32. Зависимость между медианным доходом и медианным возрастом. Планки погрешностей представляют 90% доверительные интервалы (источник [https://clauswilke.com/dataviz/visualizing-uncertainty.html])

4.6.5. Примеры визуализаций при оценке одномерного распределения

Проиллюстрируем особенности визуализации на примере распределения задержек прибытия самолетов [Koehrsen]. В качестве источника данных рассматривается набор данных NYCFlights13[1], содержащий информацию о более чем 300 000 самолетов, вылетевших из аэропортов Нью-Йорка в 2013 году. Расчеты выполнены на Python при помощи библиотек matplotlib и seaborn и доступны в Jupyter Notebook на GitHub[2] .

Фрагмент датасета приведен на рис. 4.33. Задержки рейсов указаны в минутах. Отрицательные значения означают, что самолёт совершил посадку с опережением графика.

[1] https://cran.r-project.org/web/packages/nycflights13/nycflights13.pdf
[2] https://github.com/WillKoehrsen/Data-nalysis/blob/master/univariate_dist/Histogram%20and%20Density%20Plot.ipynb

	arr_delay	name
0	11.0	United Air Lines Inc.
1	20.0	United Air Lines Inc.
2	33.0	American Airlines Inc.
3	-18.0	JetBlue Airways
4	-25.0	Delta Air Lines Inc.
5	12.0	United Air Lines Inc.
6	19.0	JetBlue Airways
7	-14.0	ExpressJet Airlines Inc.
8	-8.0	JetBlue Airways
9	8.0	American Airlines Inc.

Рис. 4.33. Фрагмент датасета

На рис. 4.34 приведены гистограммы распределения времени задержки, построенные для разных величин бинов. Видно, что ширина бина существенно влияет на вид графика. В библиотеке Matplotlib предусмотрен автоматический выбор оптимальной ширины бина, но целесообразно проверить этот выбор вручную.

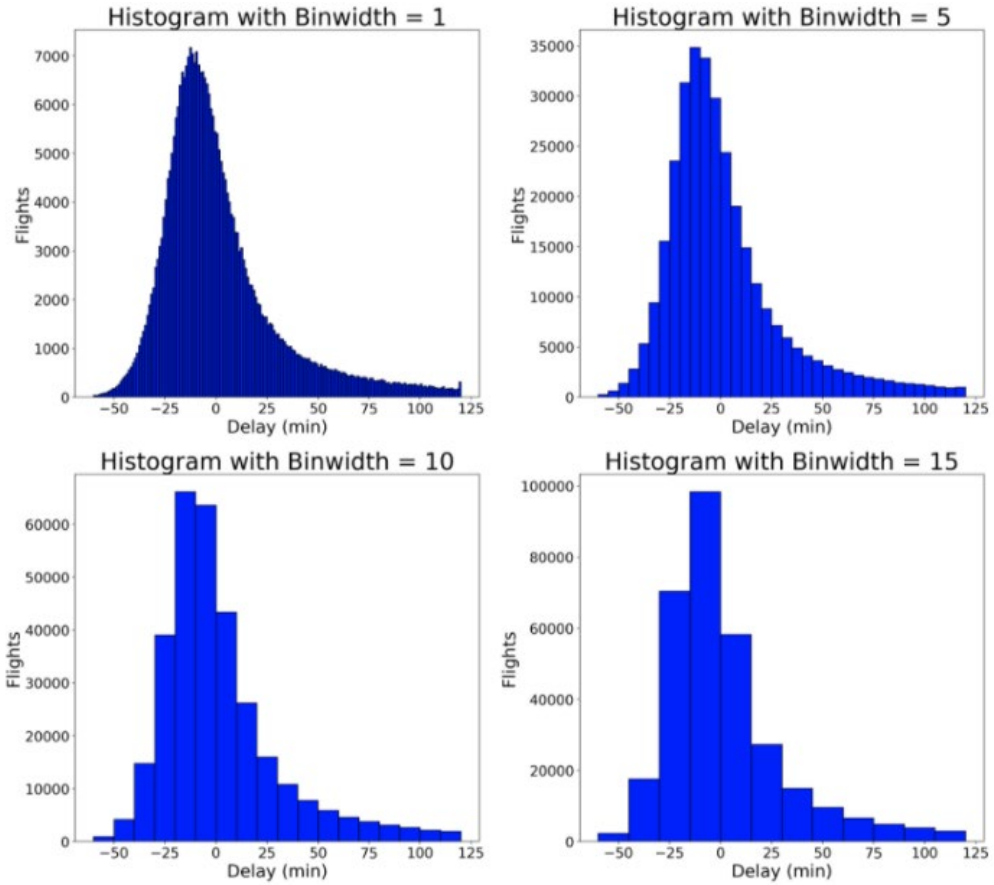


Рис. 4.34. Вид гистограммы при разной ширине бинов

Если необходимо сравнить распределения одной переменной по нескольким категориям (в данном случае – распределение задержек рейсов разных авиакомпаний), то можно использовать гистограммы другого вида, когда столбцы данных располагаются рядом (рис. 4.35, а), или столбчатые диаграммы, когда данные располагаются друг над другом (рис. 4.35, б). Однако, как легко видеть на рисунке, иллюстративная сила этих визуализаций невелика.

Рис. 4.35. Визуализация распределения переменной по нескольким категориям: а – сложная гистограмма; б – столбчатая диаграмма

Лучшие результаты дает построение графиков плотности распределения (Density Plots) с помощью ядерной оценки плотности (см. раздел 4.4.5). При построении графика (рис. 4.36) использована функция distplot в seaborn.

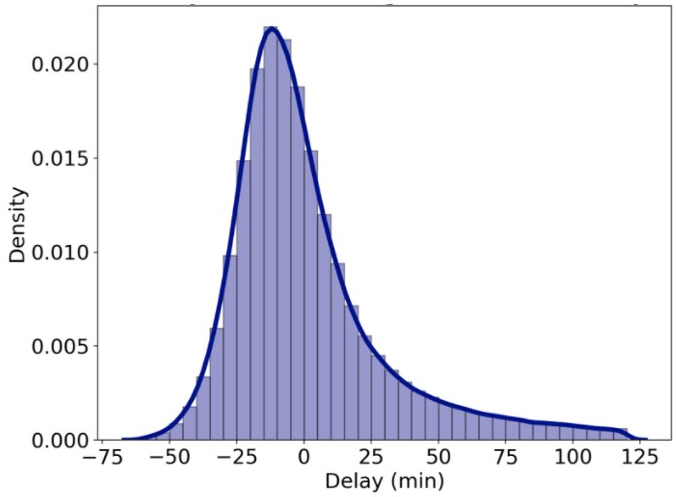


Рис. 4.36. График распределения и гистограмма задержки прибытия рейсов, построенные с помощью библиотеки seaborn

На вид графика плотности распределения существенно влияет параметр «ширина полосы пропускания», аналогичный ширине бина при построении гистограммы (рис. 4.37). Как видно из рисунка, наиболее адекватным является значение параметра по умолчанию (по Скотту[1]).

[1] https://stats.stackexchange.com/questions/90656/kernel-bandwidth-scotts-vs-silvermans-rules

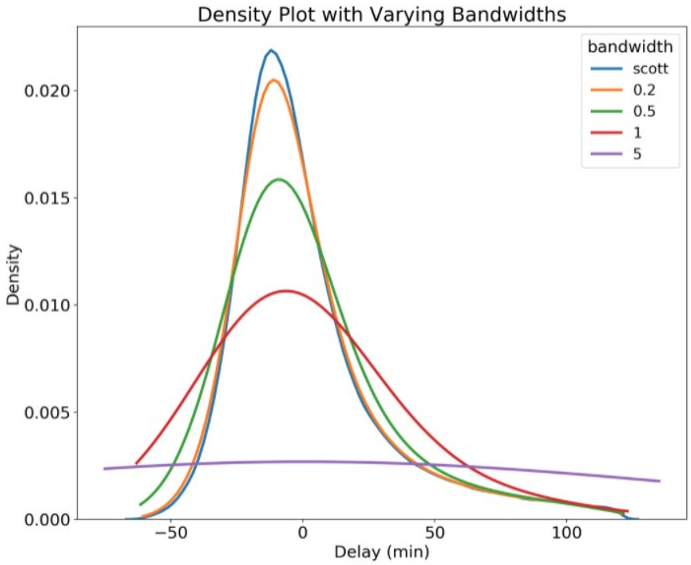


Рис. 4.37. Вид плотности распределения при разной ширине полосы пропускания

График плотности распределения можно дополнить штрих-диаграммой (Rug Plot) (рис. 4.38), которая позволяет увидеть каждое значение в распределении, а не только сглаженный график плотности. Это особенно важно в случае разреженной выборки, в которой есть интервалы, где данные вообще не представлены.

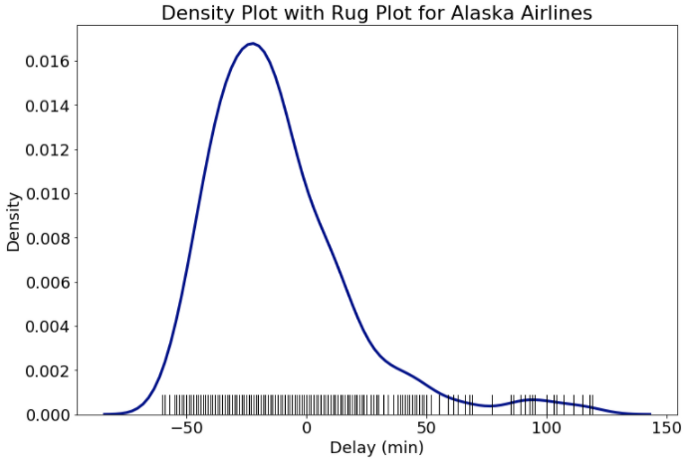


Рис. 4.38. График плотности распределения со штрих-диаграммой

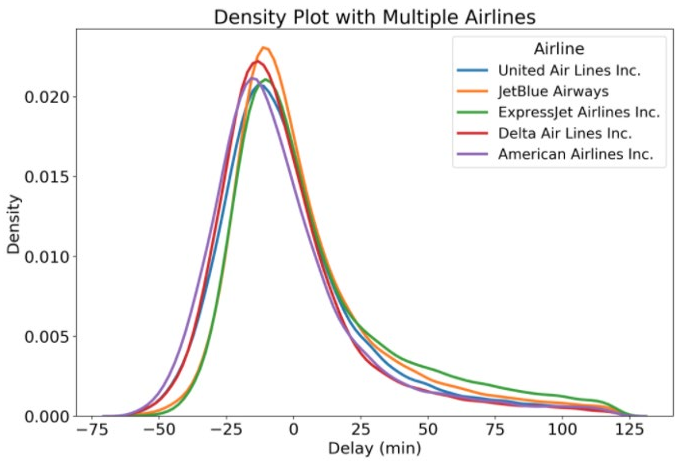


Рис. 4.39. Вид плотности распределения переменной по нескольким категориям

График плотности распределения одной переменной по нескольким категориям представлен на рис. 4.39. Он имеет гораздо большую выразительную силу по сравнению с рис. 4.37.

Вопросы для самопроверки

1. Перечислите методы сбора данных?
2. Какие две группы мер входят в дескриптивную статистику?
3. Для чего чаще всего используется мода?
4. От каких факторов зависит выбор метрики центральной тенденции?
5. Перечислите типы аномалий. Какие существуют проблемы при выявлении аномалий?
6. Достоинства и недостатки вероятностных и статистических методов обнаружения выбросов?
7. Перечислите основные недостатки методов выявления аномалий (основанных на расстоянии, кластеризации и т.д.)?
8. Какой существует инструментарий для выявления аномалий?
9. Какие подходы применяют дата-сайентисты при решении задач с малым объемом имеющейся выборки или потенциально возможными отклонениями от нормального закона?
10. Какие методы позволяют преобразовать исходную «ненормальную статистику» в «нормальную»?
11. Что главное для дата-сайентиста при выборе подхода к оценке распределения?
12. Перечислите основные методы визуализации данных и их особенности?
13. Что такое доверительный интервал?

5. ПОДГОТОВКА ДАННЫХ

5.1. Основные операции в подготовке данных

Подготовка данных (Data Preparation) – это традиционно наиболее затратная по времени фаза проекта. Разные источники по-разному оценивают долю подготовки данных в общем бюджете времени на проект, но средние цифры колеблются в диапазоне 50–70% времени проекта. Цель фазы – подготовить обучающую выборку для использования в моделировании.

К фазе подготовки данных можно переходить после того, как получено общее представление о данных (Data Understanding), т.е. выполнен разведочный анализ (см. раздел 4). Хотя методология Crisp-DM разделяет эти фазы, в реальной практике они часто выполняются итеративно, т.е. приходится уже на фазе подготовки данных возвращаться к отдельным разделам разведочного анализа, уточняя какие-то данные или проверяя вновь возникшие гипотезы.

Тем не менее, в целом при переходе к подготовке данных предполагается, что уже есть представление о статистике анализируемых данных, т.е. о типе распределений переменных и о взаимосвязях между ними, о наличии аномалий в данных (в частности, выбросов, пропущенных значений, дубликатов) и о стратегии работы с ними.

Напомним (см. раздел 1.3.3), что методология Crisp-DM предполагает в фазе «3. Подготовка данных» выполнение следующих этапов:

3.1. Выборка данных (Data Selection).
3.2. Очистка данных (Data Cleaning) – обработка ошибок, пропусков, выбросов, кодировок; анонимизация данных.
3.3. Генерация данных (Constructing new data) – агрегация и нормализация атрибутов; генерация кейсов (сэмплов) для балансировки обучающей выборки и для заполнения пропущенных значений; конвертация типов данных.
3.4. Интеграция данных (Integrating data) из разных источников в единую выборку.
3.5. Форматирование данных (Formatting Data) применительно к конкретному алгоритму моделирования.

При этом в практике работы дата-сайентиста этап 3.1 рассматривается как самостоятельный, а этапы 3.2–3.5 – как набор относительно независимых операций, выполняемых по мере необходимости для конкретной бизнес-задачи.

5.2. Выборка данных

5.2.1. Общие требования к выборке данных

На этапе выборки данных из всего многообразия информации, потенциально доступной в компании (см. раздел 4.1), отбираются те источники данных, которые необходимы и достаточны для решения поставленной бизнес-задачи.

Независимо от структуры, происхождения и способа формирования, отобранные данные должны удовлетворять критериям качества (см. раздел 1.2.6).

К сожалению, в реальной практике одновременное выполнение всех критериев качества данных практически не встречается, поэтому еще на этапе выборки данных нужно определить, с одной стороны, минимально необходимый набор критериев, а с другой стороны – возможность удовлетворить эти критерии в ходе последующих операций над данными (очистки, генерации и интеграции данных).

Помимо формальных критериев качества, при отборе данных следует активно использовать эвристики, опыт аналогичных проектов и просто соображения здравого смысла. Проиллюстрируем эти подходы на примере отбора релевантных данных.

Выявление релевантных данных является одним из самых важных шагов в дата-аналитике, так как позволяет сосредоточить свои усилия на самых важных фрагментах информации, но может быть значительной проблемой, особенно при работе с большими наборами данных. Существует множество факторов, которые следует учитывать при определении релевантных данных, включая исследовательский вопрос, цели анализа и доступные ресурсы.

Какова потенциальная релевантность атрибута решаемой задаче? Например, номер телефона или точный адрес проживания являются персональными данными клиента и, как правило, не могут быть использованы в качестве предикторов. Можно ли заменить эти признаки на наличие телефона или район проживания?

Один из подходов к определению релевантных данных – подход «сверху вниз», начиная с конкретного вопроса исследования и заканчивая выявлением конкретных данных, необходимых для ответа на этот вопрос. Например, если вопрос исследования – «Какие факторы способствуют удовлетворенности клиентов?», то соответствующие данные могут включать опросы отзывов клиентов, демографические данные клиентов и данные о продажах. Такой подход может быть полезен для сосредоточения ваших усилий на наиболее важной информации, но он также может быть ограничивающим, выводя из поля зрения другие ценные источники данных.

Другой вариант – подход «снизу вверх», начиная с имеющихся данных и заканчивая определением наиболее важных фрагментов информации. Например, если имеется большой набор данных о транзакциях клиентов, то можно начать с выявления наиболее часто покупаемых товаров, наиболее прибыльных продуктов или клиентов с самой высокой ценностью за весь срок службы. Этот подход может быть полезен для выявления закономерностей и тенденций, которые могут быть не очевидны сразу, но он также может быть непосильным в случае большого набора данных.

Независимо от выбранного подхода, при определении релевантных данных следует учитывать несколько ключевых факторов:

1. Вопрос исследования: На какой конкретный вопрос вы пытаетесь ответить? Какие данные необходимы для ответа на этот вопрос?

2. Цели анализа: Каковы цели вашего анализа? Какая информация вам нужна для достижения этих целей?

3. Релевантность данных: Действительно ли данные, которые вы рассматриваете, относятся к вопросу или целям вашего анализа?

Чтобы проиллюстрировать важность определения соответствующих данных, рассмотрим пример розничного магазина, пытающегося улучшить свои продажи. Магазин может иметь доступ к большому количеству данных, включая данные о продажах, отзывы клиентов, данные о запасах и маркетинговые данные. Однако не все эти данные имеют отношение к цели магазина по увеличению продаж. Сосредоточившись на самых важных данных, таких как данные о продажах, разбитые по категориям продуктов или демографическим данным клиентов, магазин может получить ценную информацию о том, какие продукты хорошо продаются и кто их покупает. Это, в свою очередь, может помочь в принятии решений об управлении запасами, маркетинговых стратегиях и усилиях по охвату клиентов.

5.2.2. Методы отбора признаков

Анализируемые наборы данных в исходном состоянии могут содержать большое количество признаков. Из них, как правило, только некоторые действительно важны (т.е. имеют связь с целевой переменной), а остальные являются избыточными (шумовыми), при использовании в модели лишь ухудшают ее качество и, следовательно, подлежат удалению.

Удаление избыточных признаков позволяет лучше понять данные, а также сократить время настройки модели, улучшить её точность и облегчить интерпретируемость. Часто задача удаления избыточных признаков имеет самостоятельное значение: например, нахождение оптимального набора признаков важно в банковском скоринге, а также в медицинской диагностике.

Отбор признаков – это процесс, в ходе которого из исходных признаков выбирается подмножество признаков таким образом, чтобы пространство признаков было оптимально сокращено в соответствии с определенным критерием.

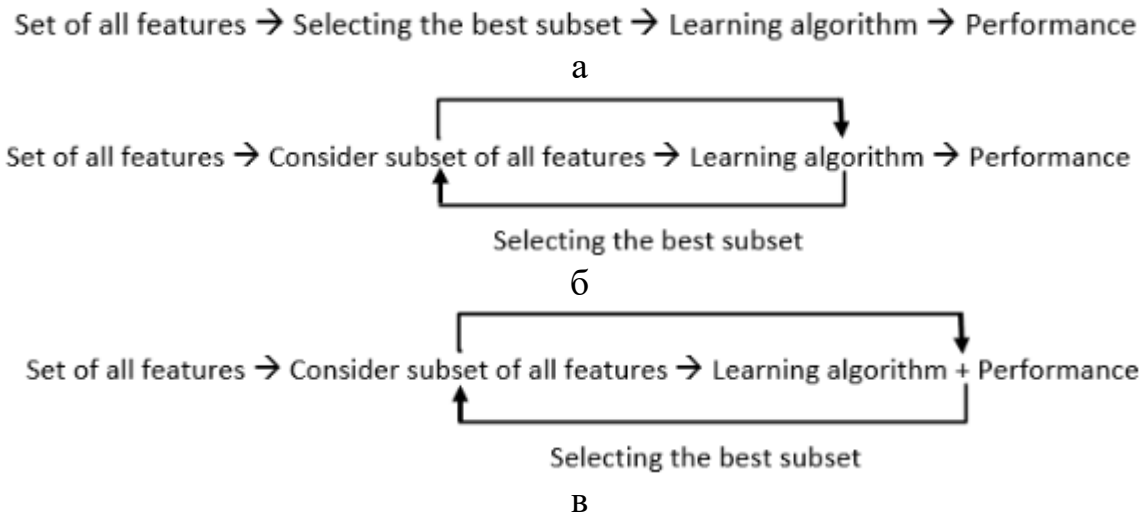


Рис. 5.1. Схемы методов отбора признаков: а – фильтрационные методы, б – методы-обертки, в – встроенные методы [https://www.geeksforgeeks.org/feature-selection-techniques-in-machine-learning/]

Методы отбора признаков обычно делят на три категории: методы фильтрации (filter methods), встроенные методы (embedded methods) и методы-обёртки (wrapper methods). Схемы работы каждой категории методов представлены на рис. 5.1. Выбор подходящего метода не всегда очевиден и зависит от задачи и имеющихся данных.

Методы фильтрации. Эти методы выбирают признаки из набора данных независимо от последующего алгоритма машинного обучения, поэтому они могут использоваться непосредственно в фазе предварительной обработки данных, т.е. до обучения модели.

К методам фильтрации можно отнести визуальный анализ (например, удаление признака, у которого только одно значение, или большинство значений пропущено) и экспертную оценку (удаление признаков, которые не подходят по смыслу, или признаков с некорректными значениями).

Однако наибольший интерес представляет формальная оценка фильтруемых признаков. В этом аспекте методы разделяются на универиантные, которые оценивают вклад в общую оценку каждого отдельного признака (независимо от других признаков), и мультивариантные, которые попарно оценивают значимость конкретного признака по сравнению с другим. Вклад признака в общую оценку может производиться с помощью различных критериев, в том числе:

– Прирост информации (Information Gain) определяется как объем информации, которую вносит использование конкретного признака в общей задаче классификации датасета:

$$IG(D, A) = H(D) – H(D|A),$$

где IG(D, A) – прирост информации от признака A в датасете D, H(D) – энтропия распределения классов в датасете D, H(D|A) – условная энтропия распределения классов в датасете D при условии признака A. В свою очередь,

$$H(p_1, p_2, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

где p_i – доля i-го класса в датасете. Для вычисления условной энтропии H(D|A) нужно разбить набор данных D на подмножества на основе значений признака A и вычислить энтропию каждого подмножества. Затем условная энтропия вычисляется как взвешенное среднее энтропий подмножеств, взвешенное по доле примеров в каждом подмножестве.

Прирост информации измеряет снижение энтропии, достигаемое при разделении набора данных на основе определенного признака, и используется для выбора наиболее информативных признаков при построении деревьев решений или аналогичных моделей машинного обучения.

– Критерий хи-квадрат (Chi-square test) обычно используется для проверки связи между категориальными переменными. Он сравнивает наблюдаемые значения из разных атрибутов набора данных с его ожидаемым значением:

$$\chi^2 = \sum \frac{(Observed\ value - Expected\ value)^2}{Expected\ value}$$

– Оценка Фишера (Fisher's Score) рассчитывается как отношение межклассовой и внутриклассовой дисперсии:

$$S_i = \frac{\sum n_j(\mu_{ij}-\mu_i)^2}{\sum n_j * \rho_{ij}^2},$$

где μ_{ij} и ρ_{ij} – среднее значение и дисперсия i-го признака в j-м классе соответственно, n_j – количество экземпляров в j-м классе, а μ_i – среднее значение i-го признака. Чем больше оценка Фишера, тем важнее выбранный признак. Однако каждый признак выбирается независимо в соответствии с оценкой Фишера, что приводит к неоптимальному набору признаков.

– Коэффициент корреляции Пирсона (Pearson's Correlation Coefficient) – это мера величины и направления связи между двумя непрерывными переменными, значения которой находятся в диапазоне от –1 до +1. Когда независимые переменные сильно коррелируют, результаты модели будут нестабильными и будут сильно различаться при небольшом изменении данных или модели. Поэтому, если два или более независимых признаков сильно коррелируют, то их можно считать дублирующимися признаками и отбросить.

– Порог дисперсии (Variance Threshold) основан на предположении, что признаки с более высокой дисперсией содержат больше информации. При этом подходе удаляются все признаки, дисперсия которых не соответствует определенному порогу. По умолчанию этот метод удаляет признаки с нулевой дисперсией.

– Средняя абсолютная разность (Mean Absolute Difference, MAD) средних значений:

$$\frac{1}{n}\sum_{i=1}^{n}|x_i - m(X)|$$

где $m(X)$ – среднее значение сэмплов x_i по датасету, n – число сэмплов в датасете. Чем больше MAD, тем более значимым релевантным признак.

– Отношение дисперсий (Dispersion Ratio) определяется как отношение среднего арифметического (AM) к среднему геометрическому (GM) для данного признака. Его значение варьируется от +1 до ∞, поскольку AM ≥ GM для данного признака. Более высокое отношение дисперсий подразумевает более релевантный признак.

– Взаимная зависимость (Mutual Dependence) определяет, являются ли две переменные взаимозависимыми, и таким образом предоставляет объем информации, полученной для одной переменной при наблюдении за другой переменной. Метод измеряет долю информации, которую признак вносит в создание целевого прогноза.

Унивариантные и простейшие мультивариантные методы фильтрации признаков реализованы в большинстве программных пакетов для машинного обучения, таких как sklearn и pandas. Кроме того, существуют специализированные мультивариантные алгоритмы фильтрации признаков, среди которых можно назвать алгоритмы семейств Relief и mRMR.

Алгоритмы семейства Relief [Urbanowicz] основаны на идентификации различий значений признаков между ближайшими парами экземпляров-соседей (рис. 5.2). Если разница значений признаков наблюдается в соседней паре экземпляров с тем же классом («попадание»), оценка признака уменьшается. Если же разница значений признаков наблюдается в соседней паре экземпляров с разными значениями класса («промах»), оценка признака увеличивается.

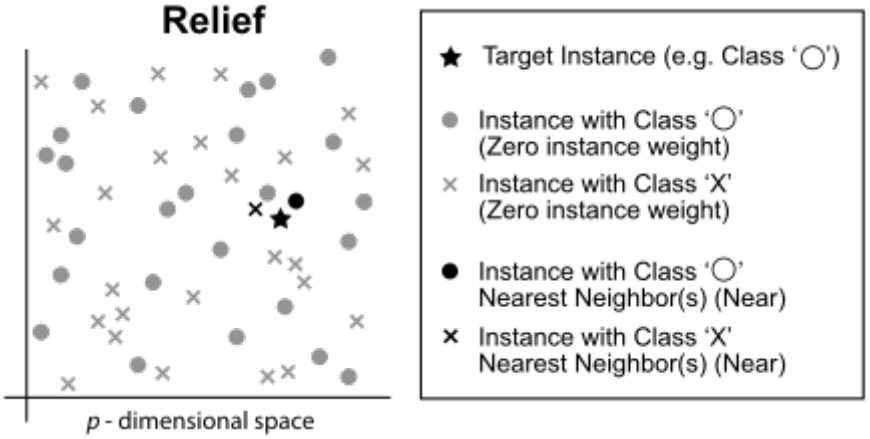


Рис. 5.2. Алгоритм Relief: выбор ближайших соседей по попаданию и промаху перед подсчетом очков

На каждой итерации берется вектор признаков X, принадлежащий одному случайному экземпляру, и векторы признаков экземпляра, ближайшего к X (по евклидову расстоянию) из каждого класса. Ближайший экземпляр того же класса называется «почти попадание» (nearHit) , а ближайший экземпляр другого класса называется «почти промах» (nearMiss). Обновляем вектор веса таким образом, чтобы

$$W_i = W_i - (x_i - \text{nearHit}_i)^2 + (x_i - \text{nearMiss}_i)^2,$$

где $1<i<p$ индексирует компоненты. Таким образом, вес любого данного признака уменьшается, если он отличается от этого признака в соседних экземплярах того же класса больше, чем в соседних экземплярах другого класса, и увеличивается в обратном случае. После m итераций каждый элемент вектора веса нормируется на m, тем самым формируется вектор релевантности. Признаки выбираются, если их релевантность больше порогового значения τ.

Алгоритмы семейства Relief не зависят от эвристик, работают за малое полиномиальное время, устойчивы к шумам и к взаимодействию признаков, а также применимы для двоичных или непрерывных данных. Однако они не выделяют избыточные признаки, а малое количество обучающих примеров вводит алгоритм в заблуждение.

Алгоритмы семейства mRMR (Minimum Redundancy Maximum Relevance) [Zhao] вычисляют меру важности признака, сравнивая его релевантность с целью и его избыточность с другими, ранее выбранными признаками. Высокая важность признака достигается, когда релевантность высока, а избыточность низкая. Значение MRMR можно получить как разность (Difference) или отношение (Ratio) между релевантностью и избыточностью. Релевантность и избыточность признака оцениваются по любому из вышеперечисленных критериев (табл. 5.1).

Таблица 5.1. Алгоритмы семейства mRMR

Метод	Критерий расчета релевантности	Критерий расчета избыточности	Схема вычисления mRMR
MID	Mutual information	Mutual information	Difference
MIQ	Mutual information	Mutual information	Ratio
FCD	F-Statistic	Correlation	Difference
FCQ	F-Statistic	Correlation	Ratio
RFCQ	Random Forests	Correlation	Ratio

В целом методы фильтрации признаков имеют низкую вычислительную сложность, но не ориентированы на работу с взаимозависимыми признаками.

Методы-обертки работают итеративно. Добавление и удаление признаков происходит на основе результатов обучения модели, причем критерии остановки для выбора лучшего подмножества обычно определяются заранее – например, производительность модели снижается до определенного уровня или достигается определенное количество признаков.

Некоторые примеры методов-оберток:

- Прямой отбор (Forward selection) представляет собой итеративный подход, при котором мы изначально начинаем с пустого набора признаков и продолжаем добавлять признак, который лучше всего улучшает нашу модель после каждой итерации. Критерий остановки – до тех пор, пока добавление новой переменной не улучшит производительность модели.
- Обратное исключение (Backward elimination) также является итеративным подходом, при котором мы изначально начинаем со всех признаков, а после каждой итерации удаляем наименее значимый признак. Критерий остановки – до тех пор, пока не будет наблюдаться улучшение производительности модели после удаления признака.
- Двунаправленное исключение (Bi-directional elimination) одновременно использует как метод прямого отбора, так и технику обратного исключения для достижения одного уникального решения.
- Исчерпывающий выбор (Exhaustive selection) рассматривается как подход «грубой силы» для оценки подмножеств признаков. Он создает все возможные подмножества, строит алгоритм обучения для каждого подмножества и выбирает подмножество, производительность модели которого наилучшая.
- Рекурсивное исключение (Recursive elimination). Этот жадный метод оптимизации выбирает признаки, рекурсивно рассматривая все меньший и меньший набор признаков. Оценщик обучается на начальном наборе признаков, и их важность получается с помощью feature_importance_attribute. Затем наименее важные признаки удаляются из текущего набора признаков, пока не останется требуемое количество признаков.

Главное преимущество методов-оберток перед методами фильтрации заключается в том, что они формируют оптимальный набор признаков для обуче-

ния модели, что приводит к лучшей точности, чем методы фильтрации. Однако они являются более затратными в вычислительном отношении.

Встроенные методы выполняют отбор признаков во время обучения модели, оптимизируя их набор для достижения лучшей точности. В этом случае алгоритм выбора признаков реализуется как параметризуемая часть алгоритма обучения. К этим методам можно отнести регуляризацию в линейных моделях и расчёт важности признаков в алгоритмах с деревьями.

Регуляризация [Регуляризация] добавляет штраф к различным параметрам модели машинного обучения, чтобы избежать переобучения модели. Два основных вида регуляризации – L1-регуляризация (англ. lasso regression), или регуляризация через манхэттенское расстояние, и L2-регуляризация, или регуляризация Тихонова (англ. ridge regression или Tikhonov regularization):

$$L_1 = \sum_i (y_i - y(t_i))^2 + \lambda \sum_i |a_i|.$$

$$L_2 = \sum_i (y_i - y(t_i))^2 + \lambda \sum_i {a_i}^2.$$

В регрессионных моделях штраф λ применяется к коэффициентам регрессии a_i, таким образом сводя некоторые коэффициенты к нулю. Признаки с нулевым коэффициентом могут быть удалены из набора данных. Такой подход к выбору признаков используется в методах отбора признаков Lasso (регуляризация L1) и Elastic Nets (регуляризация L1 и L2).

В моделях на основе деревьев (таких как Random Forest, Gradient Boosting) штрафуется выбор новых признаков по сравнению с уже выбранными признаками, дающими такой же IGain. Другими словами, если в каком-то дереве леса уже было выполнено разделение с признаком *i*, и прирост информации (Information Gain) признака *i* и признака *j* схожи, то регуляризация будет штрафовать признак *j* и отдавать предпочтение признаку *i*, который был выбран ранее. Таким образом формируются показатели важности признаков как уровень их воздействия на целевой признак.

Сравнительную оценку методов отбора признаков дает таблица 5.2 [Pudjihartono, Biswas].

Таблица 5.2. Сравнительная оценка методов отбора признаков

Метод	Достоинства	Недостатки
Фильтрационные - унивариантные	Быстрые, масштабируемые Не зависят от классификатора Снижен риск переобучения	Взаимозависимости признаков не моделируются Взаимодействие с классификатором не моделируется
Фильтрационные - мультивариантные	Могут моделировать зависимости признаков Не зависят от классификатора Снижен риск переобучения	Более медленные, хуже масштабируемые Взаимодействие с классификатором не моделируется

Методы-обертки	Моделируют зависимости признаков Лучшая производительность, чем у фильтрационных методов Моделируют взаимодействие с классификатором Лучше обобщаются по сравнению с фильтрационными	Медленнее, чем фильтрационные и встроенные методы Больше склонны к переобучению по сравнению с фильтрационными и встроенными. Вычислительные сложности при увеличении числа признаков. Отобранные признаки зависят от классификатора
Встроенные методы	Моделируют зависимости признаков Быстрее методов-оберток Моделируют взаимодействие с классификатором Снижен риск переобучения по сравнению с методами-обертками	Медленнее фильтрационных методов Отобранные признаки зависят от классификатора Плохо работают при малом числе признаков

5.3. Очистка и генерация данных

5.3.1. Общие сведения

Методология CRISP-DM выделяет очистку данных (Data Cleaning) и генерацию данных (Constructing new data) в качестве самостоятельных этапов обработки данных. Однако в реальной практике DS эти этапы рассматриваются как набор относительно независимых операций, которые дата-сайентист отбирает по мере необходимости для конкретной бизнес-задачи. Подробный обзор задач и решений в области очистки данных можно найти в [Rahm].

Обработка ошибок в данных. Здесь имеются в виду ошибки в отдельных коллекциях данных, таких как файлы и базы данных. Это, в первую очередь, опечатки при вводе данных, несоответствующая кодировка, дублирующие данные и пропуски данных. Опечатки при вводе данных можно исправить вручную (в том числе с использованием регулярных выражений) или удалить из рассмотрения. Дублирующие данные удаляются из рассмотрения. Способы кодирования данных рассмотрены в разделе 5.3.2, способы работы с пропущенными значениями – в разделе 5.3.3.

Обработка выбросов представляет собой большую и самостоятельную проблему, которая подробно рассмотрена в разделе 4.4.

Нормализация (стандартизация) данных (см. раздел 5.3.4) – приведение различных данных с исходно разными единицами измерения и диапазонами значений к единому виду, который позволит сравнивать их между собой или использовать для расчёта схожести объектов. Чаще всего выполняется стандартизация данных, при которой исходный набор данных преобразуется в новый

со средним значением, равным 0, и стандартным отклонением, равным 1, поэтому в реальной практике термины «нормализация» и «стандартизация» используются как синонимы.

Генерация данных. Чаще всего под этим понимается агрегация атрибутов с целью получения описательной статистики (например, расчет sum, avg, min, max, var и т.д.), который подробно рассмотрен в разделе 4.3.

Генерация кейсов (сэмплов) необходима для балансировки обучающей выборки и для заполнения пропущенных значений (см. раздел 5.3.5).

Анонимизация данных, т.е. удаление персональных данных, выполняется в соответствии с требованиями законодательства (см. раздел 5.3.6).

5.3.2. Кодирование данных

Числовое кодирование. Простой метод числового кодирования заключается в назначении каждому значению признака неотрицательного целого числа. Для порядковых признаков выполнение числового кодирования от малых к большим значениям признака гарантирует, что закодированные данные сохранят исходное отношение порядка. Например, признак «уровень дохода» = {бедность, низкий доход, обеспеченный, средний доход, богатый} можно преобразовать в «уровень дохода» = {0, 1, 2, 3, 4}.

One-Hot кодирование. Для номинальных признаков отношение порядка, как правило, неправомерно. Например, преобразование набора данных «марка автомобиля» = {Land Rover, Geely, Audi, Volkswagen, Mercedes-Benz} в набор данных «марка автомобиля» = {0, 1, 2, 3, 4} может привести к последующим неверным результатам моделирования и анализа. В этом случае можно использовать One-Hot кодирование, которое преобразует дискретный признак со значением K в двоичный признак K (значения 0 или 1). Для признака «марка автомобиля» = Land Rover, Geely, Audi, Volkswagen, Mercedes-Benz значения преобразованных признаков показаны в таблице 5.3.

Таблица 5.3. One-Hot кодирование

Исходное значение признака	f_1	f_2	f_3	f_4	f_5
Land Rover	1	0	0	0	0
Geely	0	1	0	0	0
Audi	0	0	1	0	0
Volkswagen	0	0	0	1	0
Mercedes-Benz	0	0	0	0	1

Фиктивное кодирование (dummy encoding). Кодирование One-Hot значительно увеличит размерность признаков, а также корреляцию между преобразованными признаками. Например, из таблицы 5.3 видно, что пять закодированных признаков имеют следующую линейную связь:

$$f1 + f2 + f3 + f4 + f5 = 1.$$

От этого недостатка свободен метод фиктивного кодирования: дискретный признак, содержащий K значений, преобразуется в K −1 двоичных признаков. Например, признак «марка автомобиля» = {Land Rover, Geely, Audi, Volkswagen, Mercedes-Benz} можно закодировать как 4 двоичных признака (таблица 5.4).

Таблица 5.4. Фиктивное кодирование

Исходное значение признака	f_1	f_2	f_3	f_4
Land Rover	1	0	0	0
Geely	0	1	0	0
Audi	0	0	1	0
Volkswagen	0	0	0	1
Mercedes-Benz	0	0	0	0

Для дискретного признака отсутствующие значения можно рассматривать как дополнительное значение. Например, в наборе данных отсутствующим значениям бинарного признака присваивается специальное значение «неизвестно», т.е. признак содержит три различных значения − «да», «нет» и «неизвестно».

5.3.3. Обработка пропущенных значений

Удаление значений. Удаление сэмплов подходит, когда некоторые сэмплы имеют несколько признаков с пропущенными значениями, при этом доля таких сэмплов невелика. Когда доля пропущенных сэмплов велика, их удаление может привести к значительной потере информации.

Удаление признаков правомерно, когда на какой-то признак приходится много пропущенных значений, при этом сам этот признак слабо влияет на цели анализа данных.

Заполнение средним значением. В этом случае для признака с пропущенными значениями сначала вычисляется среднее значение или мода непропущенных значений, а затем используется для замены пропущенных значений. Для непрерывных признаков для подстановки обычно используется среднее значение, а для дискретных признаков − мода.

Такая замена приведет к недооценке дисперсии признака. Кроме того, метод среднего значения значительно снижает корреляцию между признаками. На практике набор данных можно разделить на несколько подмножеств в соответствии с определенными вспомогательными признаками, а затем применить метод замены средним значением для каждого подмножества.

Стохастическое заполнение преодолевает проблему чрезмерной концентрации пропущенных значений за счет увеличения случайности заполненных значений. Предположим, что набор данных содержит n образцов, и есть k непропущенных значений и $(n-k)$ пропущенных значений для признака f.

Метод байесовского бутстрэппинга заполняет пропущенные значения в два этапа: на первом этапе он случайным образом извлекает $k - 1$ случайных чисел из равномерного распределения $U(0, 1)$, сортируя числа в порядке возрастания и обозначая их как $a_1, a_2, \ldots, a_{k-1}$; на втором этапе он заполняет $(n - k)$ пропущенных значений, извлекая числа из значений $\{f_1, f_2, ..., f_k\}$ с вероятностями $\{a_1, a_2-a_1, ..., 1 -a_{k-1}\}$.

Приближенный байесовский метод бутстрэппинга сначала создает новый набор F размером k из k значений, взятых с заменой среди непропущенных значений $\{f_1, f_2, ..., f_k\}$. Затем для замены каждого из $n-k$ пропущенных значений из F случайным образом выбирается одно значение.

Замена на основе модели. Признак *f* с пропущенными значениями используется в качестве цели прогнозирования, а оставшиеся признаки или их подмножества − в качестве входных признаков. Создается обучающий набор с непропущенными значениями признака *f*, и обучается классификационная или регрессионная модель, которая затем используется для прогнозирования пропущенных значений для признака *f*.

Для использования этого метода необходима оценка модели: если эффективность прогнозирования невысока, этот метод не подходит. Кроме того, замена на основе модели увеличит корреляцию между признаками.

5.3.4. Стандартизация данных

Многие алгоритмы машинного обучения, используемые в DS, требуют стандартизации входных признаков. Например, при использовании алгоритма SVM, а также линейной модели с L1 и L2 регуляризацией целевая функция предполагает, что все признаки центрированы в нуле и имеют дисперсии одного и того же порядка. Если признак имеет дисперсию большего порядка, он будет доминировать в целевой функции. Это может привести к тому, что модель будет работать плохо и не сможет учиться на других признаках.

Не требуют нормализации входных данных такие алгоритмы, как деревья принятия решений (и случайные леса), градиентный бустинг, наивный Байес. Напротив, требуют нормализации такие алгоритмы, как логистическая регрессия, SVM, нейронные сети, метод главных компонент.

К наиболее распространенным методам стандартизации данных относятся Z-оценка, минимаксная стандартизация, десятичное масштабирование.

Стандартизированная оценка (z-оценка, Standard score, z-score) − это мера относительного разброса наблюдаемого или измеренного значения, которая показывает, сколько стандартных отклонений составляет его разброс относительного среднего значения.

Пусть набор значений для признака *f* − это $\{f_1, f_2, \ldots, f_n\}$, тогда значение признака f_i, стандартизированное методом Z-оценки, равно

$$f_i' = \frac{f_i - \mu}{\sigma},$$

где $\mu = \frac{1}{n}\sum_{i=1}^{n} f_i$ − это среднее значение признака *f*, $\sigma = \sqrt{\frac{1}{2}\sum_{i=1}^{n}(f_i - \mu)^2}$ − его стандартное отклонение.

Когда в данных есть выбросы, стандартное отклонение в методе Z-оценки можно заменить средним абсолютным отклонением. В этом случае среднее абсолютное отклонение признака *f* равно

$$s = \frac{1}{n}\sum_{i=1}^{n} |f_i - \mu|,$$

и новое значение признака f_i, стандартизированное методом Z-оценки, равно

$$f_i' = \frac{f_i - \mu}{s}.$$

Минимаксная стандартизация линейно преобразует признак таким образом, что преобразованные значения находятся в интервале [0, 1]. Если исходный набор значений для признака f – это $\{f_1, f_2, \ldots, f_n\}$, то значение признака f_i, стандартизированное минимаксным методом, равно

$$f_i' = \frac{f_i - f_{min}}{f_{max} - f_{min}},$$

где f_{min} и f_{max} – минимальное и максимальное значения признака f соответственно. Поэтому, если мы хотим, чтобы стандартизированный признак принимал значения в интервале [−1, 1], то формула принимает вид

$$f_i' = \frac{2(f_i - f_{min})}{f_{max} - f_{min}} - 1,$$

а для интервала $[a, b]$ – вид

$$f_i' = \frac{(b-a)(f_i - f_{min})}{f_{max} - f_{min}} + a.$$

Десятичное масштабирование стандартизирует данные, перемещая десятичную часть значений признаков так, чтобы абсолютные стандартизированные значения признаков всегда были меньше 1. Количество перемещенных десятичных знаков зависит от максимального абсолютного значения значений признаков.

Если исходный набор значений для признака f – это $\{f_1, f_2, \ldots, f_n\}$, то значение признака f_i, стандартизированное методом десятичного масштабирования, равно

$$f_i' = \frac{fi}{10^j},$$

где j – наименьшее целое число, удовлетворяющее условию

$$max\{|f_1'|, |f_2'|, \ldots, |f_n'| < 1\}.$$

Выбор метода нормализации данных зависит от специфики нормализуемой переменной, в первую очередь, от вида распределения. В целом пайплайн нормализации данных можно представить тремя шагами:

1. Центрирование данных.
2. Масштабирование данных до заданного интервала.
3. Смещение масштабированных данных так, чтобы границы скорректированного интервала приходились на [0..1].

В качестве центра (нулевого значения) для нормальных распределений берется среднее арифметическое, для распределений, отличных от нормального – медиана, так как она практически не чувствительна к выбросам и асимметрии распределения.

Масштабирование приводит диапазоны значений признаков к одному интервалу и тем самым уравнивает их возможное влияние на выходную переменную. Однако для распределений, отличных от нормального, при выборе этого интервала могут возникнуть проблемы, которые иллюстрируются на рис. 5.3.

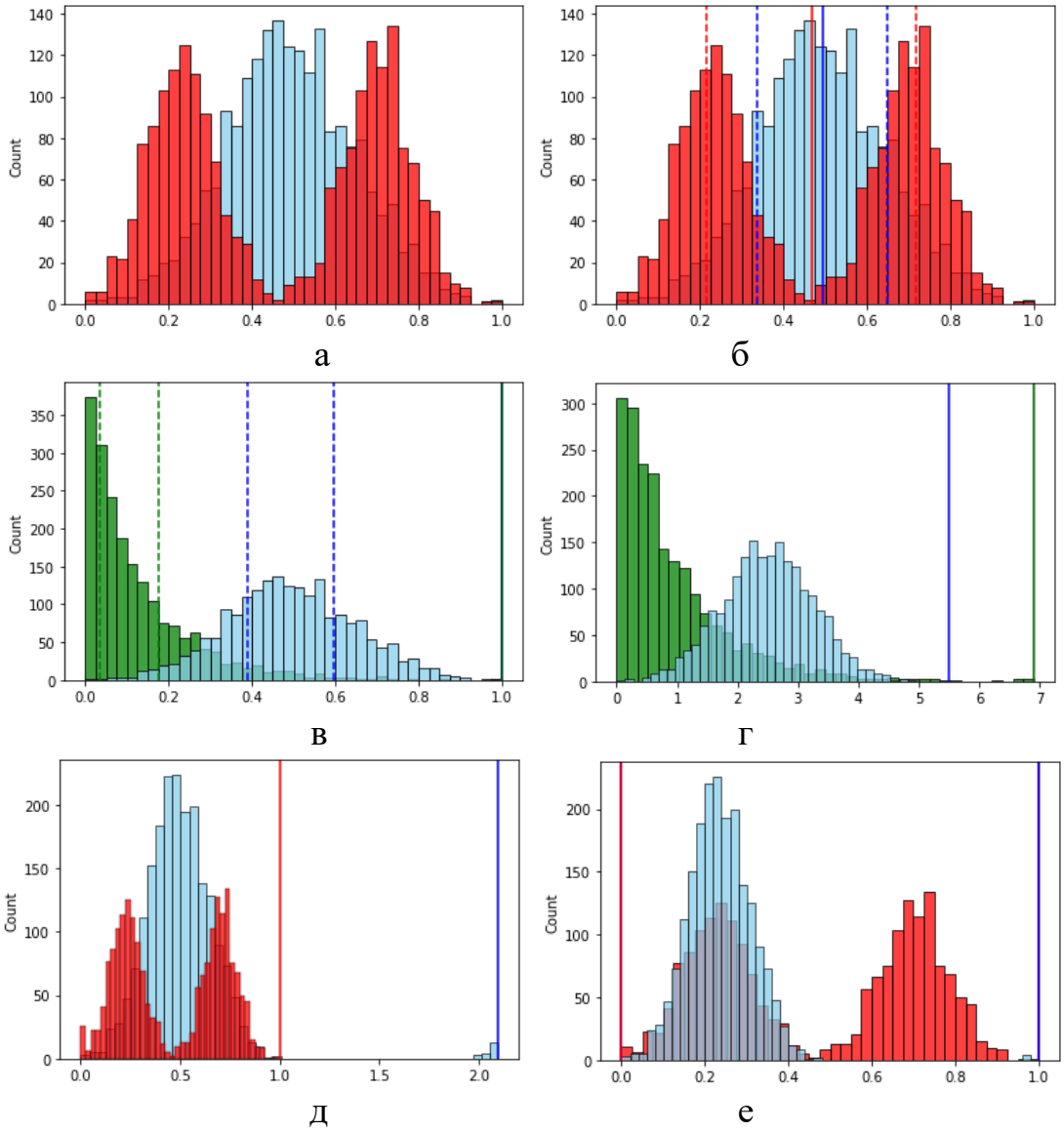


Рис. 5.3. К выбору интервала для масштабирования: а, в, д – исходные распределения; б, г, е – распределения после масштабирования (источник [https://habr.com/ru/articles/527334/])

На рис. 5.3, а, б для двух распределений – нормального и бимодального – с одинаковыми начальными диапазонами (рис. 5.3, а) в качестве интервала для масштабирования выбрано стандартное отклонение. После масштабирования новый диапазон первого признака сужается (рис. 5.3, б), и его влияние на целевую переменную снижается относительно второго признака.

На рис. 5.3, в, г для двух распределений – нормального и экспоненциального – с одинаковыми интервалами значений (рис. 5.3, а) в качестве интервала для масштабирования выбран межквартильный интервал, т.е. разница между 75-м и 25-м процентилями данных. После масштабирования интервал у признака с экспоненциальным распределением из-за большого “хвоста” стал больше (рис. 5.3, б), и его влияние на целевую переменную увеличивается относительно первого признака.

На рис. 5.3, д, использованы те же распределения, что и на рис. 5.3, а, но к нормальному распределению добавлены выбросы. Для масштабирования ис-

пользован минимаксный метод. В результате (рис. 5.3, е) значимый интервал нормального распределения уменьшился почти вдвое, что приведет к существенному снижению значимости признака.

Решением в подобных ситуациях может служить переход к скорректированному интервалу [Hubert]: границы “интервала доверия” перевычисляются с учетом асимметрии распределения, но чтобы для симметричного случая он был равен всё тому же 1,5 * IQR (рис. 5.4). В результате величина интервала, за пределами которого находятся выбросы, одинакова у каждого признака, и они уравниваются по возможному влиянию на целевую переменную.

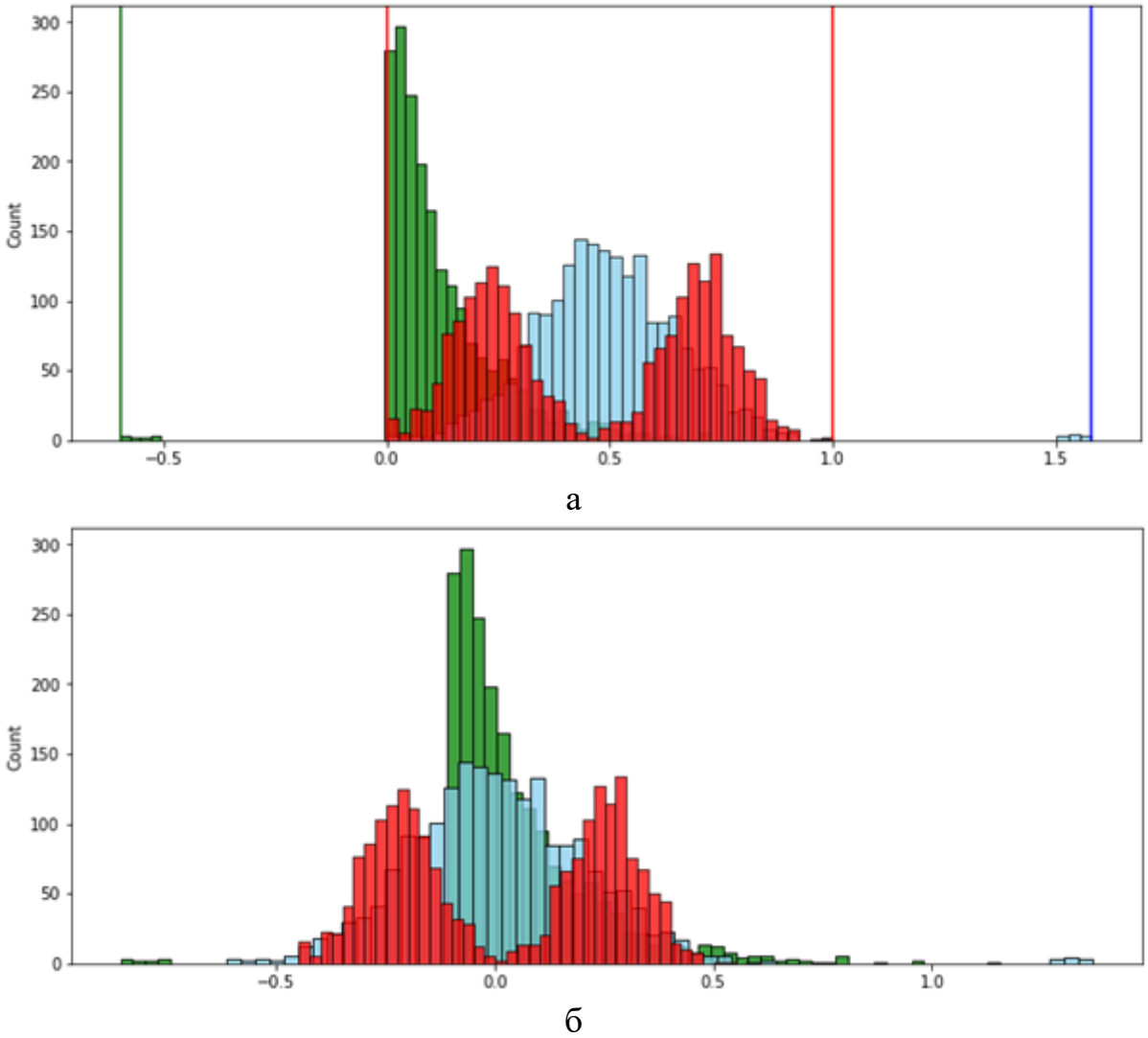


Рис. 5.4. Нормализация данных методом скорректированного интервала:
а – исходные распределения, б – скорректированные распределения
(источник [https://habr.com/ru/articles/527334/])

5.3.5. Генерация кейсов (сэмплов)

Генерация кейсов, т.е. новых экземпляров (сэмплов) набора данных, используется для заполнения пропущенных значений, но основное применение этой операции – балансировка обучающей выборки для последующего алгоритма машинного обучения.

Несбалансированными считаются наборы данных, в которых количество сэмплов, связанных с каждым классом, сильно варьируется. Например, в общем объеме банковских транзакций количество мошеннических транзакций, как правило, составляет менее 0,1%. Как правило, проблемы несбалансированных данных возникают тогда, когда соотношение сэмплов в классах меньшинства и большинства падает ниже 25%. Эти проблемы можно решать тремя способами:

- на уровне модели – модицифировать модель, чтобы придать больший вес менее репрезентативным классам;
- на уровне оценки – использовать альтернативные метрики оценки качества моделей (например, взвешенную F-меру);
- на уровне данных – ввести новые экземпляры в класс меньшинства.

Способы последнего типа объединяются в группу методов выборки данных (data samplers).

Случайный неполный выбор (Random Under-Sampler) уменьшает выборки более крупных классов, случайным образом выбирая доступные экземпляры из каждого класса. Количество выбранных экземпляров определяется как часть приемлемого порога баланса класса и, следовательно, является переменным. При этом никакие данные не генерируются искусственно, но происходит большая потеря доступных обучающих данных и, в конечном итоге, снижение производительности модели.

Случайный избыточный выбор (Random Over-Sampler) увеличивает выборку класса меньшинства за счет случайного повторения некоторых его экземпляров. В результате в наборе данных появляется смещение, и могут сдвинуться разделяющие границы классов.

Синтетическая передискретизация меньшинства (Synthetic Minority Oversampling Technique, SMOTE) генерирует недостающие экземпляры в классе меньшинства в соответствии со следующим алгоритмом. Пограничные области аппроксимируются опорными векторами после обучения классификатора SVM на исходном наборе обучающих данных. После вычисления образцы синтезируются рядом с аппроксимированной границей. SMOTE остается одним из самых распространенных механизмов передискретизации и привел к большому семейству вариантов.

Пограничный SMOTE (BorderlineSMOTE) – это вариант алгоритма SMOTE, основанный на предположении, что наиболее релевантными являются экземпляры, наиболее близкие к границе принятия решений. При этом граница принятия решений определяется путем рассмотрения ошибок классификации в пределах K-соседей экземпляра.

SVM-SMOTE аппроксимирует пограничные области опорными векторами после обучения классификатора SVM на исходном наборе обучающих данных. После вычисления образцы синтезируются рядом с аппроксимированной границей.

Адаптивная синтетическая выборка (ADASYN) генерирует больше синтетических данных там, где плотность сэмплов из класса меньшинства низкая, и наоборот.

На рис. 5.5 показано сравнение различных методов группы SMOTE.

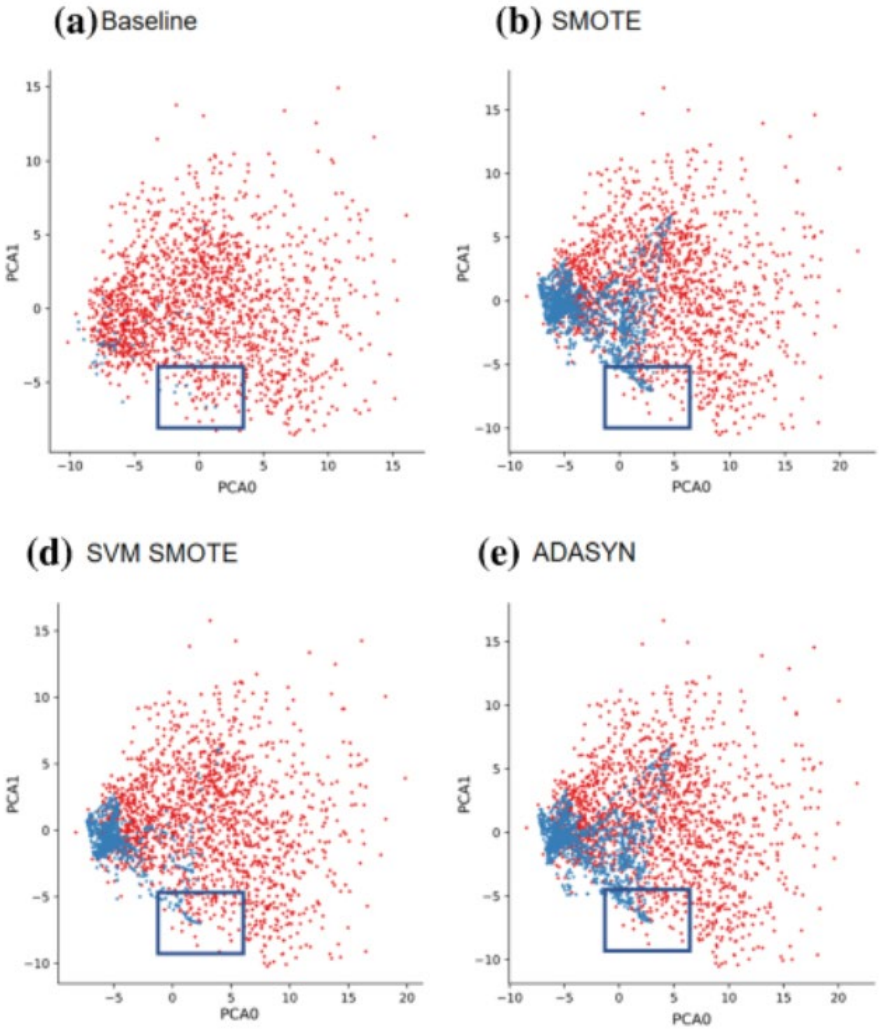


Рис. 5.5. Сравнение различных методов группы SMOTE

5.3.6. Анонимизация данных

Определение и виды анонимизация данных. Анонимизация (обезличивание) данных включает в себя процесс преобразования персонально идентифицируемой информации (personally identifiable information, PII) в неидентифицируемые данные, что делает невозможным или крайне сложным связывание этих данных с конкретным лицом. Анонимизация данных объединяет процессы и инструменты, позволяющие сделать конфиденциальные данные неузнаваемыми для авторизованных пользователей, но функциональными для них.

Анонимизация данных регулируется законодательно. Так, существует Общий регламент Европейского союза по защите данных (General Data Protection Regulation, GDPR). Международный стандарт ISO 29100:2011 фокусируется на управлении PII в организациях, подчеркивая необходимость защиты личной конфиденциальности путем деидентификации PII, чтобы предотвратить повторную идентификацию анонимных данных. Сравнительно новый международный стандарт SO/IEC 27559:2022 описывает конкретные методы и процессы анонимизации данных.

В Российской Федерации существует Федеральный закон от 27.07.2006 № 152-ФЗ “О персональных данных”. Согласно пункту 2 статьи 5 обработка

персональных данных должна ограничиваться достижением конкретных, заранее определенных и законных целей, а в статье 6 установлено, что обработка персональных данных осуществляется с согласия субъекта персональных данных. Несмотря на то, что Россия не входит в Европейский союз, Федеральный закон № 152 “О персональных данных” содержит в себе ключевые принципы GDPR.

Наиболее распространенные типы данных, требующие маскировки:

- Персональные данные (PII), такие как имена, паспорта, социальное страхование и номера телефонов;
- Защищенная информация о состоянии здоровья человека, уходе за ним или платежных данных;
- Защищенные финансовые данные;
- Тестовые данные, связанные с жизненным циклом разработки программного обеспечения.

Маскированные данные обычно используются в непроизводственных средах, таких как разработка и тестирование программного обеспечения, аналитика, машинное обучение и обмен данными B2B, которые не требуют исходных производственных данных. В связи с этим возникают требования к методикам и инструментам анонимизации данных:

- Реляционная согласованность. Когда одни и те же данные разбросаны по множеству различных исходных систем (устаревших, локальных, облачных), они могут отображаться по-разному в разных местах. ПО для маскирования данных должно иметь возможность маскировать все идентичные сущности (имена клиентов, счета клиентов, покупки и пр.) согласованно во всех системах. Аналогично, если счета клиентов и покупки клиентов хранятся в отдельных системах, одни и те же замаскированные идентификаторы должны использоваться во всех системах, чтобы гарантировать, что данные остаются реляционно согласованными;
- Контекстная целостность набора данных. Например, если интернет-провайдер маскирует поле «почтовый индекс» в почтовом адресе клиента, то домашний IP-адрес этого человека должен быть замаскирован, чтобы соответствовать новому почтовому адресу. Несовпадающие данные являются подозрительными и могут насторожить потенциальных хакеров;
- Обеспечение динамической маскировки данных. Например, когда агенты колл-центра могут видеть только последние 4 цифры номера кредитной карты (на основе их прав доступа и привилегий), первые 12 цифр необходимо маскировать «на лету»;
- Обеспечение безопасности централизованного хранилища токенизации важно, если выбрана политика обратимой анонимизации (токенизации). В этом случае исходные конфиденциальные данные хранятся вместе с соответствующими токенами (бессмысленными данными) в централизованном хранилище токенов. Такое хранилище, содержащее PII каждого клиента, является естественной целью для хакеров.

Комплексное выполнение этих требований представляет собой сложную проблему. Согласно стандарту ISO/IEC 29100, есть две методики обезличивания – необратимое (ISO/IEC 29100 anonymization) и обратимое (ISO/IEC 29100 pseudo-anonymization), у каждой из которых есть свои преимущества и недостатки [Петровский] (Таблица 5.5). поэтому к выбору методики и инструментария анонимизации следует подходить внимательно, учитывая конкретную бизнес-задачу и политику компании. Например, политика компании может запрещать использование атрибутов с персональной информацией в качестве предикторов даже относительно данных не выходящих за пределы компании.

Таблица 5.5. Сравнение методик анонимизации данных

Необратимая анонимизация	Обратимая анонимизация
+ полная анонимизация	– возможна деанонимизация
+ подходит для dev-сред	+ подходит для dev- и prod-сред
– нарушается распределение данных	+ сохраняется распределение данных
– нарушается корреляция данных	+ сохраняется корреляция данных

Методы анонимизации структурированных данных.

1. Обобщение – замена конкретных значений более обобщенными, тем самым уменьшая гранулярность данных. Например, вместо хранения точного возраста человека его можно обобщить в возрастной диапазон (например, 20-30 лет).
2. Подавление – удаление или маскировка определенных атрибутов, которые потенциально могут идентифицировать отдельных лиц. Например, если набор данных включает имена и адреса, эти поля можно подавить, заменив их уникальными идентификаторами или просто удалив их полностью.
3. Возмущение – добавление контролируемого шума или рандомизации к данным, что затрудняет связывание анонимизированных записей с исходными лицами. Например, числовые значения, такие как доход или возраст, могут быть возмущены путем добавления небольшого случайного значения к каждой записи.
4. Маскировка – замена конфиденциальных данных вымышленными, но реалистичными значениями. Например, вместо хранения реальных номеров кредитных карт их можно заменить случайно сгенерированными числами в том же формате. Маскировка данных обычно используется в ситуациях, когда данные необходимо предоставить третьим лицам или использовать в целях тестирования. Крайне важно гарантировать, что замаскированные данные непреднамеренно не раскроют какую-либо конфиденциальную информацию посредством шаблонов или корреляций.
5. Токенизация – замена конфиденциальных данных уникальными токенами или ссылками. Например, номера кредитных карт можно токенизировать, заменив их случайно сгенерированным токеном, который не имеет прямой связи с исходным номером. Токенизация обычно используется в сценариях, где данные необходимо хранить или передавать безопасно. Важно реализовать надежные меры безопасности для защиты сопоставления токенов и ис-

ходных данных, поскольку это сопоставление может потенциально привести к повторной идентификации в случае компрометации.

6. Дифференциальная конфиденциальность – введение тщательно откалиброванного шума в ответы на запросы, что затрудняет определение наличия или отсутствия конкретных лиц в данных. Дифференциальная конфиденциальность гарантирует, что даже при доступе к анонимным данным злоумышленник не сможет различить, присутствует ли информация конкретного лица или нет. Эта техника обеспечивает строгую математическую основу для защиты конфиденциальности, но она требует тщательного проектирования и реализации, чтобы сбалансировать гарантии конфиденциальности с полезностью данных.

Лучшие практики анонимизации неструктурированных данных. Неструктурированные данные (см. раздел 1.2.3) относятся к любым данным, которые не организованы в предопределенном формате, например, текст, изображения, аудио, видео и т.д. Анонимизация неструктурированных данных может быть сложной задачей, поскольку данные могут содержать различные типы информации, которые трудно обнаружить формальными способами, поэтому здесь принято говорить не столько о методах, сколько о лучших практиках.

Ключевым понятием в этой задаче является конфиденциальный (чувствительный) атрибут (sensitive attribute, SA), описывающий данные, которые могут быть предоставлены для целей исследования/статистического анализа, но не должны быть связаны с отдельным пользователем. Квази-идентификатор (QID) состоит из неконфиденциального атрибута (или набора атрибутов), который может быть объединен или связан с внешней/фоновой информацией для повторной идентификации лица, к которому относятся данные. Наконец, ключевой атрибут состоит из явного/уникального идентификатора (ID) лица или, другими словами, персонально идентифицируемой информации (PII).

Анонимизация неструктурированных данных реализуется как последовательность шагов:

1. Выявление конфиденциальной информации в данных. Первым шагом является определение того, какой тип информации необходимо сделать анонимным и где она находится в данных. Это можно сделать, вручную просматривая данные или используя автоматизированные инструменты, которые могут сканировать и классифицировать данные на основе предопределенных критериев. Например, инструмент может идентифицировать и маркировать имена, даты, местоположения или другие сущности в текстовом документе или лица, номерные знаки или логотипы на изображении.

2. Выбор подходящего метода анонимизации в зависимости от типа и формата данных, требуемого уровня анонимности и влияния на качество и полезность данных. Здесь могут быть использованы вышеприведенные методы анонимизации в том числе удаление, маскирование, замена.

3. Оценка эффективности и влияния анонимизации как с точки зрения защиты данных, так и с точки зрения полезности данных. С этой целью могут использоваться следующие критерии:

- Анонимность – степень защиты или безопасности, которую обеспечивает анонимизация, в сочетании со сложностью повторной идентификации субъектов данных из анонимизированных данных. Это можно количественно оценить с помощью таких метрик, как *k*-анонимность, *l*-разнообразие или *t*-близость [Cunha], которые фиксируют уровень неразличимости или разнообразия среди записей данных или уровень сходства или различия между чувствительными и нечувствительными атрибутами данных.
- Полезность – степень ценности или полезности, которую сохраняет анонимизация, применительно к предполагаемой цели анализа. Это можно количественно оценить с помощью таких метрик, как потеря информации, искажение или ошибка, которые фиксируют объем изменений или отклонений от исходных данных или влияние на производительность или функциональность данных.
- Компромисс – баланс между анонимностью и полезностью анонимизированных данных или оптимальному или приемлемому уровню анонимизации, который удовлетворяет как требованиям защиты данных, так и требованиям полезности данных. Это можно определить с помощью таких методов, как анализ затрат и выгод, оценка рисков или анализ чувствительности.

Важно отметить, что задача анонимизации данных не имеет универсального решения, и различные типы данных, форматы и сценарии могут требовать разных подходов и соображений. С конкретными практиками по анонимизации данных можно познакомиться по литературным источникам [Cunha, Петровский]. Опыт авторов пособия по деидентификации информации из электронных медицинских карт подробно описан в статье [Suzdaltseva].

5.4. Интеграция данных

Хорошо, когда данные берутся из корпоративного хранилища или заранее подготовленной витрины данных. Однако часто данные необходимо загружать из нескольких источников, и для подготовки обучающей выборки требуется их интеграция.

Интеграция данных (Integrating data) – процесс объединения данных из нескольких источников в организации для предоставления полного, точного и актуального набора данных для бизнес-анализа, анализа данных и других приложений и бизнес-процессов. Она включает репликацию данных, прием и преобразование для объединения различных типов данных в стандартизированные форматы для хранения в целевом репозитории, таком как хранилище данных (data warehouse) или озеро данных (data lake).

Существуют различные **подходы для выполнения интеграции данных**: ETL, ELT, потоковая передача, хранилище данных, интеграция приложений (API), виртуализация данных, интеграция промежуточного ПО, пакетная обработка. Для реализации этих процессов инженеры по данным, архитекторы и разработчики могут либо вручную кодировать архитектуру с помощью SQL, либо, что чаще, они настраивают и управляют инструментом интеграции данных, который оптимизирует разработку и автоматизирует систему.

Конвейер ETL (Extract, transform, load – Извлечение, преобразование, загрузка) (рис. 5.6) – это традиционный тип конвейера данных. Данные преобразуются перед загрузкой в систему хранения данных, т.е. вне системы хранения данных, как правило, в отдельной промежуточной области. Конвейеры данных ETL могут быть лучшим выбором в сценариях, где качество и согласованность данных имеют первостепенное значение, поскольку процесс преобразования может включать строгие этапы очистки и проверки данных.

Конвейер ELT (Extract, load, transform – Извлечение, преобразование, загрузка) включает извлечение данных из источника, загрузку их в базу данных или хранилище данных в необработанном виде, а затем их последующее преобразование внутри БД в формат, который соответствует потребностям бизнеса. Конвейеры данных ELT обычно используются в проектах больших данных и обработке в реальном времени, где скорость и масштабируемость имеют решающее значение.

Процесс ELT в значительной степени опирается на мощность и масштабируемость современных систем хранения данных. Загружая данные перед их преобразованием, ELT в полной мере использует вычислительную мощность этих систем. Такой подход обеспечивает более быструю обработку данных и более гибкое управление данными по сравнению с традиционными методами.

Потоковая передача (Real-time data integration), т.е. интеграция данных в реальном времени, включает сбор и обработку данных по мере их поступления в исходные системы, а затем немедленную интеграцию их в целевую систему. Этот метод потоковой передачи данных обычно используется в сценариях, где требуются самые последние сведения, например, для аналитики в реальном времени, обнаружения и мониторинга мошенничества.

Одна из форм интеграции данных в реальном времени – сбор данных об изменениях (change data capture, CDC) – применяет обновления, внесенные в данные в исходных системах, к хранилищам данных и другим репозиториям. Затем эти изменения можно применить к другому репозиторию данных или сделать доступными в формате, пригодном для использования, например, ETL или другими типами инструментов интеграции данных.

Интеграция приложений (Application integration, API) позволяет отдельным приложениям работать вместе, перемещая и синхронизируя данные между ними.. Этот метод обычно используется в сценариях, где разным приложениям необходимо обмениваться данными и работать вместе, например, чтобы гарантировать, что ваша HR-система имеет те же данные, что и ваша финансовая система.

Виртуализация данных (Data virtualization) подразумевает создание виртуального слоя, который обеспечивает унифицированное представление данных из разных источников, независимо от того, где физически находятся данные. Он позволяет пользователям получать доступ и запрашивать интегрированные данные по требованию без необходимости физического перемещения данных. Он полезен для сценариев, где гибкость и доступ к интегрированным данным в реальном времени имеют решающее значение.

Интеграция федеративных данных (Federated data integration). Данные остаются в исходных системах, а запросы выполняются в этих разрозненных системах в реальном времени для получения необходимой информации. Он лучше всего подходит для сценариев, где данные не нужно физически перемещать, и их можно виртуально интегрировать для анализа. Хотя федеративная интеграция уменьшает дублирование данных, она может страдать от проблем с производительностью.

Интеграция промежуточного программного обеспечения (Middleware Integration). В этом случае промежуточное программное обеспечение может форматировать и проверять данные перед отправкой в целевую систему. Такой тип интеграции обеспечивает бесперебойное соединение между разнородными системами и может обрабатывать сложные сценарии интеграции, поддерживая различные форматы данных и протоколы.

Пакетная обработка (Batch Processing) подразумевает сбор и обработку данных большими группами или пакетами через запланированные интервалы времени. Данные обрабатываются пакетами в часы наименьшей нагрузки, чтобы снизить нагрузку на систему. Этот тип интеграции подходит для обработки больших объемов данных, не требующих анализа в реальном времени и часто используется в финансовой отчетности, обработке заработной платы и других периодических задачах по обработке данных.

Варианты использования интеграции данных. Интеграция данных используется в широком спектре отраслей и сценариев для решения различных бизнес-задач и задач. Наиболее распространенные варианты использования интеграции данных включают:

- Хранилище данных: интеграция данных используется при построении хранилища данных для создания централизованного хранилища данных для аналитики и базовой отчетности.
- Разработка озер данных: Среды больших данных часто включают комбинацию структурированных, неструктурированных и полуструктурированных данных. Перемещение этих данных из изолированных локальных платформ в озера данных упрощает извлечение ценности путем выполнения расширенной аналитики данных, включая искусственный интеллект (ИИ) и машинное обучение (МО).
- Обзор клиента на 360°: консолидация данных о клиентах из разных источников, таких как системы управления взаимоотношениями с клиентами (CRM) , маркетинговые базы данных и платформы поддержки, позволяет организациям создавать единое представление о каждом клиенте. Хорошо интегрированные данные о клиентах могут помочь компаниям лучше нацеливать свои маркетинговые усилия, выявлять возможности перекрестных/дополнительных продаж и предоставлять лучшее обслуживание клиентов.
- Бизнес-аналитика и отчетность: интеграция данных имеет важное значение для создания комплексных отчетов и панелей бизнес-аналитики, которые предоставляют информацию о различных аспектах эффективности бизнеса, таких как продажи, маркетинг, финансы и операции.

– Обработка данных Интернета вещей: интеграция данных с устройств Интернета вещей (IoT) позволяет организациям контролировать и управлять подключенными устройствами, анализировать данные датчиков и автоматизировать процессы на основе информации в режиме реального времени.

Инструменты интеграции данных. На протяжении многих лет наиболее распространенный подход к интеграции данных требовал от разработчиков ручного написания сценариев на языке структурированных запросов (SQL) — стандартном языке программирования, используемом в реляционных базах данных. Сегодня поставщики ИТ предлагают множество различных инструментов интеграции данных, которые автоматизируют, оптимизируют и документируют процесс интеграции данных, начиная от решений с открытым исходным кодом и заканчивая комплексными платформами интеграции данных. Эти системы интеграции данных обычно включают в себя многие из следующих инструментов:

- Инструменты ETL: Инструменты ETL используются для извлечения данных из различных источников, преобразования их в соответствии с желаемым форматом или структурой, а затем загрузки их в целевую систему, включая хранилища данных и базы данных. Помимо хранилищ данных, эти инструменты используются для интеграции данных и миграции данных.
- Enterprise service bus (ESB) и middleware: эти инструменты облегчают интеграцию различных программных приложений и служб, предоставляя инфраструктуру обмена сообщениями и связи. Они обеспечивают обмен данными в реальном времени, оркестровку рабочих процессов и управление API.
- Инструменты репликации данных: Инструменты репликации данных используются для непрерывной репликации данных из исходных систем в целевые системы, поддерживая их синхронизацию. Интеграция данных в реальном времени, аварийное восстановление и сценарии высокой доступности являются обычными вариантами использования этих инструментов.
- Инструменты виртуализации данных: используются для создания виртуального слоя, который обеспечивает унифицированное представление данных из разных источников – независимо от того, где физически находятся данные. Эти инструменты позволяют пользователям получать доступ и запрашивать интегрированные данные без необходимости физического перемещения данных.
- Платформы интеграции данных как услуга (iPaaS) : решения iPaaS предлагают облачные службы интеграции данных, включая преобразование данных, маршрутизацию данных, управление API и подключение к различным облачным и локальным приложениям. Обычно используются для гибридной облачной интеграции и подключения приложений SaaS.
- Инструменты интеграции потоковых данных: Эти инструменты фокусируются на интеграции потоковых данных в реальном времени из таких

источников, как устройства IoT, датчики, социальные сети и потоки событий. Они позволяют организациям обрабатывать и анализировать данные по мере их генерации.

- Инструменты качества данных и управления данными: Инструменты, помогающие гарантировать, что данные, интегрированные из нескольких источников, соответствуют стандартам качества, нормативным актам и политикам управления данными. Эти инструменты часто включают возможности профилирования данных, очистки и управления метаданными.
- Инструменты CDC: Инструменты CDC фиксируют и реплицируют изменения в данных из исходных систем в режиме реального времени. Эти инструменты часто используются для поддержания актуальности хранилищ данных, а также для аналитики в режиме реального времени.
- Инструменты управления основными данными (MDM) : инструменты MDM фокусируются на управлении основными данными клиентов, продуктов, сотрудников и других типов и обеспечивают их согласованность и точность в масштабах всей организации. Эти инструменты часто включают возможности интеграции данных для консолидации и синхронизации основных данных из различных систем.
- Платформы управления API: эти платформы предлагают инструменты для проектирования, публикации и управления API. Хотя их основная цель – обеспечить интеграцию API, они играют важную роль в подключении систем и приложений.

5.5. Конвертация и форматирование данных

Хотя методология CRISP-DM относит конвертацию данных (Data Conversion) и форматирование данных (Data Formatting) к разным этапам фазы подготовки данных, в реальной практике эти понятия часто рассматриваются как взаимозаменяемые.

Содержательно задача этих этапов заключается в следующем: привести подготовленные данные к виду и формату, который может быть непосредственно использован в выбранной модели машинного обучения. Это особенно актуально для тех алгоритмов, которые работают с определенным форматом данных, например:

- если речь идет об анализе временного ряда – к примеру, прогнозируем ежемесячные продажи торговой сети – возможно, здесь можно будет использовать данные в формате .csv, но предварительно отсортированные;
- если речь идет об анализе изображений, то на вход модели нужно подать файл, содержащий многомерную матрицу яркостей, но с удаленной служебной информацией.

В зависимости от политики компании, принятого пайплайна решения бизнес-задачи и используемых средств автоматизированной поддержки к этапу конвертации данных могут быть отнесены следующие операции:

- Изменение форматов файлов. Этот процесс включает преобразование данных, хранящихся в одном формате файла, в другой. Примером может быть преобразование файла «data.xlsx» (электронная таблица Excel) в файл «data.docx» (документ Word);
- Адаптация кодировки символов. Кодировка символов – это набор правил, которые компьютеры используют для понимания и отображения текста; Адаптировать его означает убедиться, что текст отображается правильно, особенно при работе с языками и символами из разных уголков мира. Например, преобразование данных из UTF-8 в UTF-16 позволяет использовать более широкий диапазон символов;
- Настройка форматов даты и времени. Корректировка формата даты и времени включает в себя переформатирование представления даты и времени, чтобы они были единообразными повсюду – например, замена «2023-09-21» на «21.09.2023»;
- Изменение единиц измерения: В некоторых случаях данные могут быть первоначально выражены в одной системе измерения, например в милях и фунтах, что приводит к необходимости преобразования в альтернативную систему, например в километрах и килограммах. Примером этого является преобразование 10 миль в 16.09 километра;
- Преобразование чисел: Это предполагает изменение способа представления чисел. Например, преобразование целого числа в десятичную, например превращение 5 в 5.0, может быть важным для точных вычислений;
- Преобразование типов данных: Типы данных относятся к изменению того, как компьютер интерпретирует определенные типы информации. Например, убедиться, что компьютер понимает, что «да» и «правда» означают одно и то же. Примером является преобразование поля базы данных со значением «Да» в логическое поле со значением «истина»;
- Очистка данных. Очистка данных подразумевает исправление несоответствий и неточностей в наборах данных, обеспечение их целостности и надежности. Примером может служить устранение повторяющихся записей имен клиентов в списке контактов;
- Пользовательские преобразования данных: Настройка преобразований данных включает в себя адаптацию конкретных изменений данных, чтобы привести их в соответствие с уникальными требованиями и целями. Например, в маркетинге это может включать указание местоположения клиента (полный адрес), чтобы более эффективно ориентироваться на определенную демографическую группу;
- Обработка ошибок. Если во время преобразования данных возникают ошибки, крайне важно использовать методы обработки ошибок. Ведение журнала включает в себя документирование каждого шага преобразования, что позволяет отслеживать, анализировать и исправлять ошибки. Например, обычной практикой является систематическая регистрация ошибок преобразования данных в специальном файле для последующего анализа и исправления.

Для разрешения терминологической неоднозначности может быть полезна Таблица 5.6.

Таблица 5.6. Содержание различных операций подготовки данных

Аспект	Конвертация данных (Data conversion)	Миграция данных (Data migration)	Преобразование данных (Data transformation)	Очистка данных (Data cleansing)
Задача	Изменить формат или структуру данных	Переместить данные в новое место	Преобразовать данные в соответствии с требованиями	Очистить данные для улучшения их качества
Содержание изменений	Переформатировать или перевести данные	Сохранить данные как есть	Преобразовать контент или структуру	Исправить ошибки и несоответствия
Цель изменений	Узкий фокус на формат\структуру	Только релокация данных	Широкие изменения содержания данных	Фокус на качестве данных
Примеры	Конвертация csv в xml	Миграция данных в новую БД	Агрегирование данных о продажах	Удаление дубликатов в записях
Результат	Измененные данные в новом формате	Данные остаются неизменными	Данные, адаптированные для новой задачи	Очищенные данные без ошибок

Вопросы для самопроверки

1. Какие этапы входят в фазу «3. Подготовка данных» в рамках методологии CRISP-DM?
2. Назовите ключевые критерии в выборке данных?
3. К методам фильтрации можно отнести…?
4. Главное преимущество методов-оберток перед методами фильтрации?
5. В чем заключается простой метод числового кодирования данных?
6. В каких случаях правомерно удаление признаков?
7. Какие алгоритмы машинного обучения не требуют нормализации входных данных?
8. Для чего выполняется генерация новых экземпляров (семплов) набора данных?
9. Какие типы данных требуют маскировки? Какие существуют требования к методикам и инструментам анонимизации данных?
10. Перечислите подходы для выполнения интеграции данных и наиболее распространенные варианты и инструменты их использования?
11. Приведите примеры операций: конвертации, миграции, преобразования и очистки данных?

6. МОДЕЛИРОВАНИЕ В DATA SCIENCE

6.1. Выбор алгоритмов

Как подчеркивалось в разделе 1.1, основной целью DS является исследование разрозненных источников данных и поиск лучших способов анализа информации с целью прогнозирования потенциальных тенденций в конкретной предметной области – например, оценить будущий сезонный спрос на товары. Для этого можно, опираясь на статистику прошлых продаж и используя готовые инструменты (такие как Excel, Tableau, SAS, SPSS и т. д.), провести ретроанализ, который позволяет обнаружить основные закономерности и предсказать их дальнейшее развитие – например, построить сезонную кривую. Как показывает практика, для многих проблем, имеющих место в реальном бизнесе, такой чисто статистический подход оказывается вполне достаточным.

Однако для все большего числа проблем этот подход оказывается малоэффективным (например, излишне грубым) или даже неправомерным, и требуется переход к поиску закономерностей средствами машинного обучения (МО). Правильно подобранный и реализованный алгоритм прогностического моделирования чаще всего находит закономерности лучше, чем человек вручную, и быстрее собирает данные.

На выбор алгоритма МО решающее влияние оказывают три группы факторов:

- структура и содержание решаемой бизнес-задачи и ее соответствие в терминах задачи машинного обучения. Задачи в МО обычно делятся на категории, такие как классификация, регрессия, кластеризация и т.д. (см. раздел 1.1). Например, предсказание того, является ли электронное письмо спамом или нет, является задачей классификации, в то время как предсказание цен на жилье является задачей регрессии;
- тип и структура имеющихся данных. Данные могут быть числовыми, категориальными, текстовыми или основанными на изображениях, и для каждого из них требуются различные методы предварительной обработки и моделирования. Например, числовые данные могут хорошо подходить для регрессионных моделей, в то время как категориальные данные могут потребовать кодирования перед использованием в алгоритме МО. Поэтому важно знать, какие модели имеют хотя бы шанс хорошо работать с имеющимися данными;
- уровень сложности модели МО. Процесс выбора подходящей модели МО должен проходить постепенно. Лучше всего начать с простейших моделей в своей категории, таких как линейная или логистическая регрессия. Эти модели легко построить и интерпретировать, они быстро обучаются и обеспечивают надежный бейзлайн для дальнейших усовершенствований. Постепенное увеличение сложности модели позволяет определить, обеспечивают ли более сложные модели значительные улучшения.

В таблице 6.1 представлены основные алгоритмы МО, с которых рекомендуется начинать выбор для использования в DS.

Таблица 6.1. Основные модели МО для использования в DS

Алгоритм	Описание и варианты использования	Преимущества	Недостатки
Линейная регрессия (Linear Regression	Моделирует отношения между непрерывными переменными. Большие датасеты с высокой размерностью	Хорошо работает с пропусками и выбросами, работает с категориальными и числовыми переменными	Предполагается линейная зависимость между признаками (плохо работает при нелинейных зависимостях), чувствительность к выбросам
Логистическая регрессия (Logistic Regression)	Прогнозирует вероятности бинарных результатов. Большие датасеты с малым числом признаков	Легко интерпретируется, подходит для проблем классификации	Предполагается линейная зависимость между признаками (плохо работает при нелинейных зависимостях), чувствительность к выбросам
Дерево решений (Decision Tree)	Разделяет данные на ветви для принятия решения. Выдача кредита, кредитный скоринг, медицинская диагностика	Легко визуализировать, работает с категориальными и числовыми переменными	Легко переобучается, требуется большой объем данных
Случайный лес (Random Forest)	Создает несколько деревьев решений для надежных прогнозов. Прогнозирование фондового рынка, погоды, медицинская диагностика	Работает с пропусками и выбросами, с категориальными и числовыми переменными	Требуется большой объем данных
Градиентный бустинг (GBA)	Усиливает слабые классификаторы, объединяя их в ансамбль. Прогностическая аналитика, рекомендательные системы, обнаружение мошенничества	Высокая точность, эффективно находит нелинейные зависимости в данных различной природы	Легко переобучается, требуется большой объем данных
Адаптивный бустинг (Ada-Boost)	Усиливает слабые классификаторы, объединяя их в ансамбль	Легко интерпретируется, подходит для проблем классификации и регрессии; высокая	Чувствителен к шуму в данных и выбросам, плохо работает с несбалансированными данными;

		точность при правильной подготовке данных; работает со сложными, нелинейными зависимостями; высокая обобщающая способность.	склонен к переобучению при большом числе базовых моделей
Машина опорных векторов (SVM)	Находит оптимальную гиперплоскость для классификации. Может быть адаптирован к обнаружению аномалий в данных; обработка многомерных данных без их предварительного преобразования или понижения размерности.	Подходит для проблем классификации и регрессии; хорошо работает с нелинейными зависимостями, в условиях малых датасетов и высокоразмерных данных	Вычислительно затратный, плохо работает на больших датасетах, неустойчив к шуму
К ближайших соседей (K Nearest Neighbors)	Классифицирует на основе наиболее близких обучающих примеров. Небольшие датасеты, мало признаков. Рекомендации по продуктам, обнаружение аномалий, распознавание образов	Простой, интуитивно понятный, эффективный для классификации и регрессии	Плохо работает на больших датасетах, чувствителен к нерелевантным признакам, вычислительно затратный
Наивный Байес (Naïve Bayes)	Большие датасеты, мало признаков. Классификация текста, анализ настроений, обнаружение спама	Простой, интуитивно понятный, эффективный для классификации и регрессии	Предполагает отсутствие зависимости между признаками
К средних (K means)	Кластеризует данные в К групп на основе сходства. Большие датасеты, мало признаков. Сегментация клиентов, анализ рынка, сжатие изображений	Простой, легко интерпретируется, обрабатывает большие датасеты	Чувствителен к выбору начальных центров кластеров, вычислительно затратный, не всегда сходится к глобальному оптимуму
Алгоритмы снижения размерно-	Уменьшает пространство признаков, сохраняя дисперсию. Большие датасеты высокой размерности.	Упрощает модели, удаляет шум, улучшает производительность, может работать с пропус-	Вычислительно затратные, трудно интерпретировать

сти (SVD, PCA)	Визуализация данных, извлечение признаков, ускорение обучения основной модели	ками	
Нейронные сети (Neural Networks)	Большие датасеты высокой размерности. Распознавание скрытых закономерностей и корреляций	Хорошо работают с нелинейными зависимостями и высокоразмерными данными	Вычислительно затратные, трудно интерпретировать

6.2. Тестирование модели

6.2.1. Дилемма «смещение–дисперсия»

Известная народная мудрость «бесплатный сыр бывает только в мышеловке» применительно к моделям МО формулируется в виде теоремы об отсутствии бесплатных обедов (No free lunch theorem) [Wolpert]. Содержательно эта теорема говорит о том, что не существует идеальной модели МО, т.е. такой модели, которая бы хорошо работала в любой ситуации, а не только в той, на которую она была обучена.

На практике эта теорема задает связь между сложностью модели, точностью ее предсказаний и тем, насколько хорошо она может делать предсказания на ранее неизвестных данных, которые не использовались для обучения модели.

Очевидно, что модель с малым количеством настраиваемых параметров может уловить только некоторые (и не обязательно базовые) связи, существующие в наборе данных, что приводит к недообучению модели. По мере увеличения количества настраиваемых параметров в модели она становится более гибкой и может лучше соответствовать набору обучающих данных. Однако излишне сложные модели будут просто запоминать всю конфигурацию тренировочного набора данных, а не выделять закономерности в нем, и будет плохо работать на ранее не виденных (тестовых) данных, т.е. произойдет переобучение модели.

Смещение – это ошибка модели, возникающая в результате ошибочного предположения в алгоритме обучения. В результате большого смещения алгоритм может пропустить связь между признаками и выводом (недообучение). Типичные алгоритмы, приводящие к высоким смещениям модели МО – это линейная и логистическая регрессия; напротив, алгоритмы, приводящие к низкому смещению – это деревья решений, k-NN и SVM.

Дисперсия – это ошибка модели, возникающая вследствие излишней чувствительности к малым отклонениям в тренировочном наборе. При высокой дисперсии алгоритм пытается трактовать все колебания тренировочного набора, в том числе случайные и шумовые, а не выделять желаемые закономерности (переобучение). Соотношение между ошибками смещения и дисперсии показано на рис. 6.1.

Таким образом, возникает неустранимая дилемма «смещение–дисперсия», которую дата-сайентисту приходится каждый раз разрешать в виде компромисса между допустимыми ошибками недообучения и переобучения модели МО, адекватными для конкретной задачи (рис. 6.2).

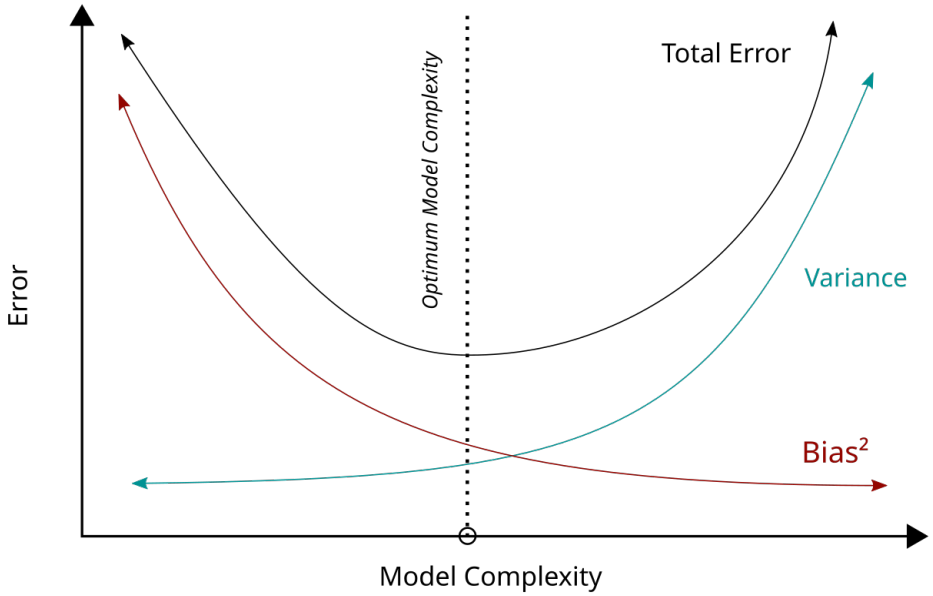


Рис. 6.1. Соотношение между ошибкой и сложностью модели

Рис. 6.2. Дилемма «смещение–дисперсия» (источник [https://www.techtarget.com/searchenterpriseai/feature/How-to-build-a-machine-learning-model-in-7-steps])

6.2.2. Стратегии оценки модели

Методология CRISP-DM ставит этап планирования тестирования (Generating a test design) перед собственно обучением модели, и это вполне оправдано. Чтобы правильно обучить модели и выбрать лучшую из них, необходимо заранее выбрать стратегию оценки и метрики оценки, адекватные для конкретной решаемой задачи.

Стратегия оценки определяет то, каким образом будут разделены имеющиеся данные для обучения модели и проверки ее эффективности.

В МО используются две группы стратегий оценки модели – повторная выборка (resampling) и вероятностные методы.

Методы повторной выборки, как следует из названия, являются простыми приемами перестановки выборок данных для проверки того, хорошо ли работает модель на выборках данных, на которых она не была обучена. Другими словами, методы повторной выборки помогают понять, будет ли модель хорошо обобщать. К ним относятся случайное разделение (random split), разделение по времени (time-based split), кросс-валидация (k-fold cross-validation), бутстрап (bootstrap).

При *случайном разделении* имеющиеся данные делятся случайным образом на три различных набора, обычно в соотношении 60–20–20%:

- Обучающая выборка (Training set) – используется для обучения (настройки параметров) модели;
- Тестовая выборка (Test set) – используется для оценки прогнозируемой ошибки модели;
- Валидационная выборка (Validation set) – используется для оценки ошибки генерализации.

Преимущество этого метода в том, что есть большая вероятность того, что исходная популяция хорошо представлена во всех трех наборах. Говоря более формально, случайное разделение предотвратит смещенную выборку данных.

Разделение по времени используется для данных типа временных рядов или других последовательностей, для которых случайные разделения невозможны. Например, если нужно обучить модель для прогнозирования погоды, то нельзя случайным образом разделить данные на обучающие и тестовые наборы. В таких случаях используется разделение по времени. Например, обучающий набор может содержать данные за последние три года и 10 месяцев текущего года, а последние два месяца резервируются для тестового или валидационного набора.

Существует также подход «оконных» наборов данных, когда модель обучается до определенной даты и тестируется в будущих датах итеративно, так что окно обучения продолжает увеличиваться, сдвигаясь на один день (следовательно, тестовый набор также уменьшается на день). Такой подход стабилизирует модель и предотвращает переобучение, когда тестовый набор очень мал (скажем, от 3 до 7 дней).

При обучении модели МО на последовательностях нужно постоянно следить за тем, чтобы все данные обучающего набора были статистически однородными. Например, пандемия коронавируса разделила экономические данные на «до» и «после», и использовать их для обучения в едином датасете нужно крайне осторожно.

Перекрестная проверка (кросс-валидация) организуется путем случайного перемешивания набора данных и последующего его разделения на групп. После этого на каждой k-й итерации одна из групп необходимо рассматривается как тестовый набор, в то время как все остальные группы объединяются в обучающий набор. Модель тестируется на тестовой группе, и процесс повторяется k раз для каждой из k групп. Таким образом, к концу процесса имеется k различ-

ных результатов на k различных тестовых группах. Лучшей объявляется модель с наивысшим баллом.

Стратифицированная кросс-валидация аналогична предыдущему варианту, но специально разработана для обработки несбалансированных наборов данных. В этом случае на каждой итерации соотношение между классами сохраняется (рис. 6.2). Это делает оценку модели более точной, а обучение модели – менее предвзятым.

Бутстрап – один из самых мощных способов получения стабилизированной модели. Он близок к технике случайного расщепления, поскольку следует концепции случайной выборки. Бутстрап-выборка формируется из оригинала с помощью выборки с заменой. Например, при N=5 можно «перевыбрать» 5 раз из [1,2,3,4,5] и получить [2,5,4,4,1]). При больших N вероятность получить бутстрап-выборки, идентичные оригиналу, весьма мала. Модель обучается на выборке bootstrap, а затем оценивается на всех тех точках данных, которые не попали в выборку bootstrap. Они называются выборками out-of-bag.

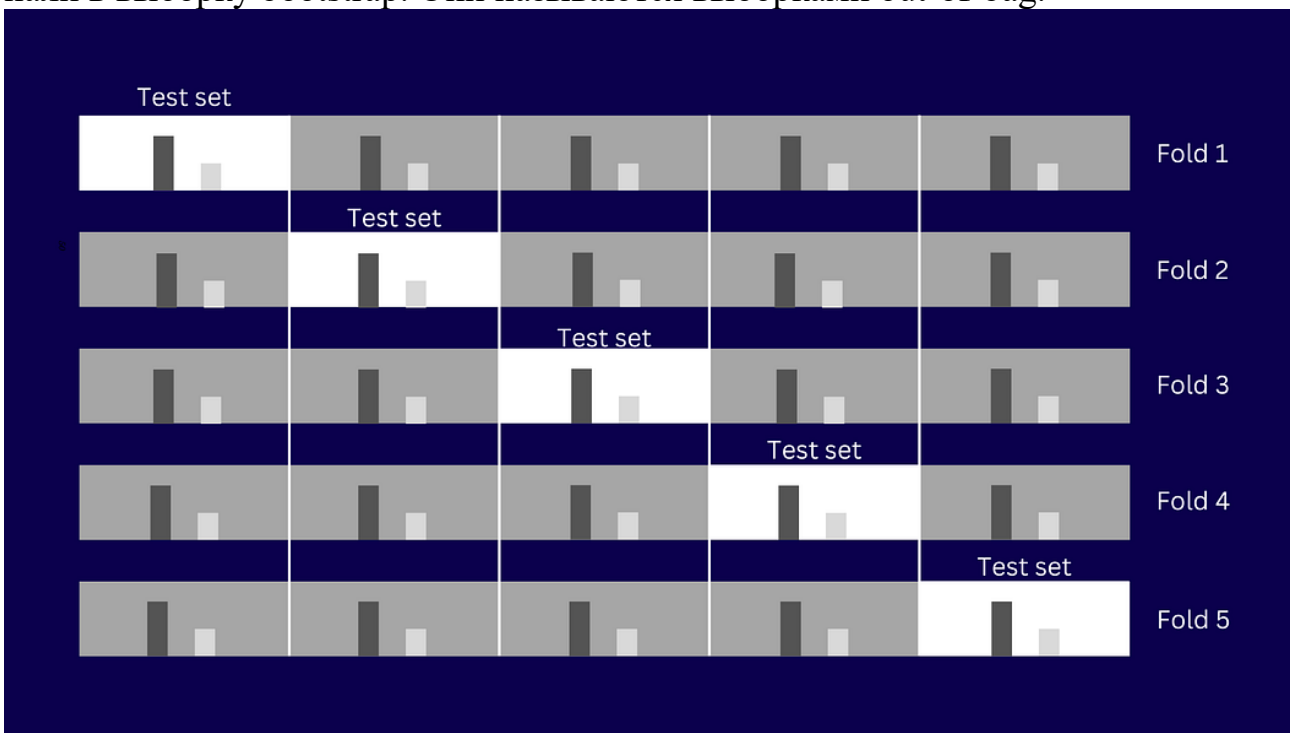


Рис. 6.2. Стратифицированная кросс-валидация (источник[https://medium.com])

Вероятностные методы направлены на то, чтобы в определенной степени ограничить сложность формируемой модели и тем самым потенциально улучшить ее обобщаемость.

Критерий информации Акаике (AIC) является мерой потери информации, которая характерна для конкретной модели:

$$AIC=(2K-2\log L)/N,$$

где K – количество независимых переменных или предикторов, L – максимальное правдоподобие модели, N – количество точек данных в обучающем наборе. Напомним, что максимальное правдоподобие означает максимизацию условной вероятности наблюдения точки данных X, учитывая параметры и заданное распределение вероятностей. Критерий полезно использовать в случае небольших

наборов данных. Однако, по сравнению с другими информационными критериями, он имеет тенденцию выбирать более сложные модели, которые теряют меньше обучающей информации.

Байесовский информационный критерий (BIC) был разработан на основе концепции байесовской вероятности и подходит для моделей, которые обучаются с использованием оценки максимального правдоподобия:

$$BIC = K \times \log N - 2 \log L,$$

где K – количество независимых переменных или предикторов, L – максимальное правдоподобие модели, N – количество точек данных в обучающем наборе. *BIC* штрафует модель за ее сложность и предпочтительно используется, когда размер набора данных не очень мал (в противном случае он имеет тенденцию останавливаться на очень простых моделях).

Существуют также другие стратегии ограничения сложности модели, такие как *минимальная длина описания* (MDL) и *минимизация структурного риска* (SRM). Но в целом все вероятностные стратегии используются не на этапе предварительного отбора моделей, а, скорее, на этапе валидации гиперпараметров отобранного пула моделей.

6.2.3. Выбор метрик эффективности модели

Различные варианты функционалов качества для задач DS подробно описаны в разделе 3.2.3. Ниже перечисляются наиболее типичные метрики, используемые для оценки эффективности моделей МО при решении различных задач DS.

Задачи классификации

Матрица ошибок (confusion matrix) фиксирует количество правильно и неправильно классифицированных случаев:

	актуальные 0	**актуальные 1**
предсказанные 0	Истинно отрицательные, TN	Ложноотрицательные, FN
предсказанные 1	Ложноположительные, FP	Истинно положительные, TP

Точность (Accuracy) – это самая простая метрика, которую можно определить как количество правильно классифицированных тестовых случаев, деленное на общее количество тестовых случаев.

Прецизионность (Precision) показывает долю объектов, названных классификатором положительными и при этом действительно являющимися положительными:

$$\text{Precision} = TP/(TP+FP).$$

Полнота (Recall) показывает, какую долю объектов положительного класса из всех объектов положительного класса нашел классификатор:

$$\text{Recall} = TP/(TP+FN).$$

F1-мера (F1-score) представляет собой гармоническое среднее значение полноты и точности:

$$\text{F1-score} = 2 \times \text{Precision} \times \text{Recall} / (\text{Precision}+\text{Recall})$$

ROC-кривая (AUC-ROC) представляет собой график истинно положительного уровня (Recall) против ложноположительного уровня (TN/(TN+FP)). AUC-ROC означает площадь под рабочей характеристикой приемника, и чем больше

площадь, тем лучше производительность модели. Если кривая находится где-то вблизи диагональной линии 50%, это говорит о том, что модель случайным образом предсказывает выходную переменную.

Логарифм потерь (Log Loss) является очень эффективной метрикой классификации. Он представляет собой взятый с обратным знаком логарифм от функции правдоподобия, где функция правдоподобия показывает, насколько вероятным, по мнению модели, был наблюдаемый набор результатов. Поскольку функция правдоподобия дает очень малые значения, лучший способ их интерпретации – преобразовать значения в логарифм и добавить отрицательное значение, чтобы изменить порядок метрики на обратный, так что более низкая оценка потерь предполагает лучшую модель.

Графики прироста и подъема (Gain and lift charts) (рис. 6.3) по принципу действия аналогичны матрице ошибок, но с существенным отличием. Матрица ошибок определяет производительность модели на всей популяции или на всем тестовом наборе, тогда как диаграммы прироста и подъема оценивают модель на частях всей популяции. Таким образом, у нас есть оценка (ось Y) для каждого процента популяции (ось X). Графики подъема измеряют улучшение, которое модель приносит по сравнению со случайными прогнозами. Улучшение называется «подъемом».

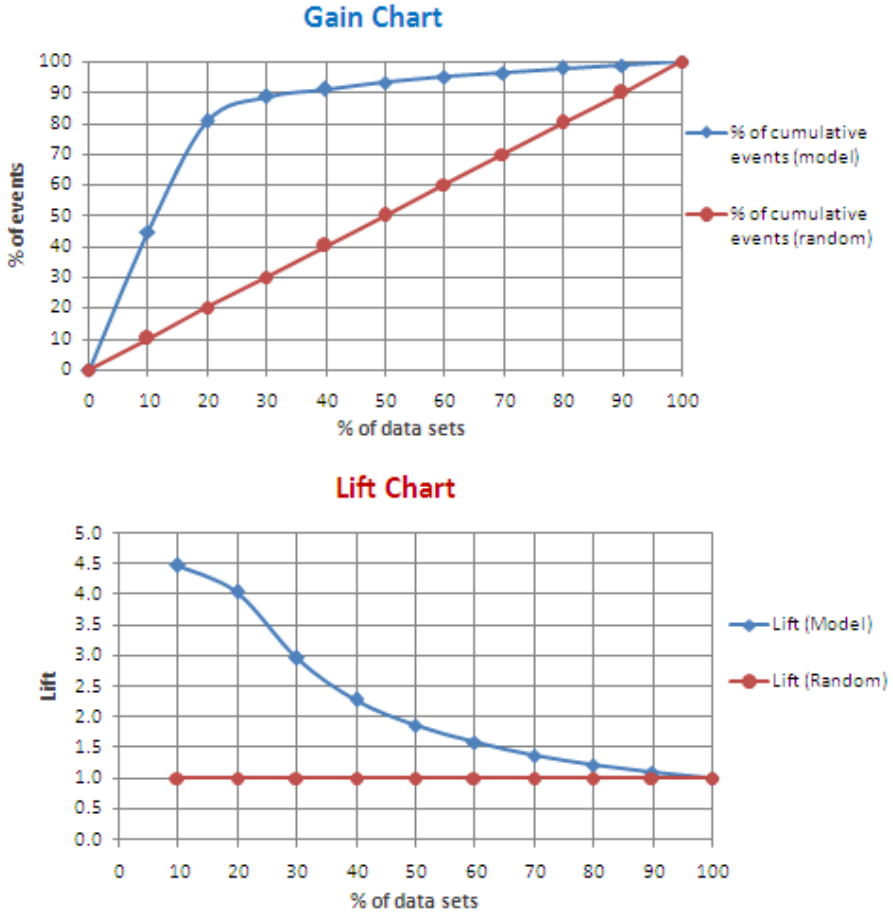


Рис. 6.3. Графики прироста и подъема

Диаграмма Колмогорова-Смирнова (K-S chart, Kolmogorov-Smirnov chart) (рис. 6.4) интерпретирует эффективность моделей классификации как степень разделения между положительными и отрицательными распределениями.

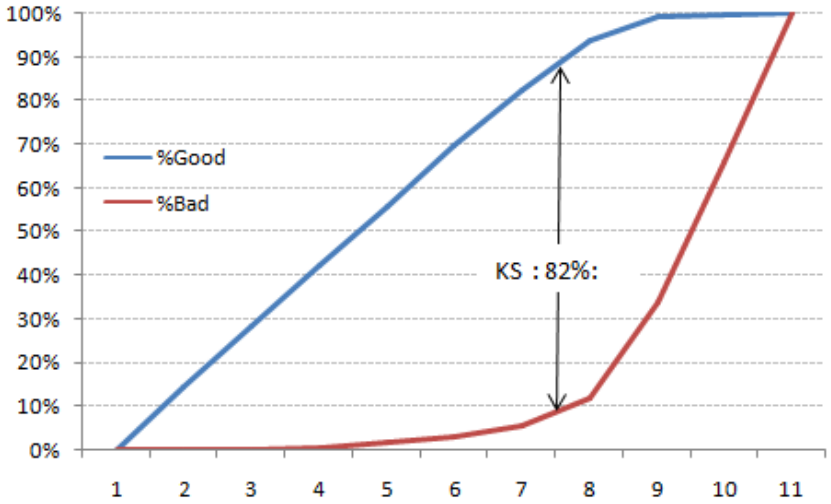


Рис. 6.4. Диаграмма Колмогорова-Смирнова
[https://discuss.boardinfinity.com/t/what-is-kolomogorov-smirnov-chart/4805/1]

KS=100, если оценки разделяют совокупность на две отдельные группы, в которых одна группа содержит все положительные, а другая – все отрицательные примеры. С другой стороны, если модель не может различать положительные и отрицательные случаи, то KS=0. В большинстве моделей классификации KS будет находиться в диапазоне от 0 до 100, и чем выше значение, тем лучше модель разделяет положительные и отрицательные случаи.

Задачи регрессии

Среднеквадратическая ошибка. MSE – простая метрика, которая вычисляет разницу между фактическим и прогнозируемым значением (ошибку), возводит ее в квадрат, а затем выдает среднее значение всех ошибок:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widetilde{y}_i)^2$$

RMSE является корнем MSE и полезна, поскольку помогает приблизить масштаб ошибок к фактическим значениям, что делает его более интерпретируемым.

Средняя абсолютная ошибка (MAE) – среднее значение абсолютных значений погрешности, т.е. разности между фактическим и прогнозируемым значением:

$$MAE = \frac{1}{n}\sum_{i=1}|x_i - x|$$

Среднеквадратическая ошибка логарифма (RMSLE) использует то же уравнение, что и RMSE, за исключением добавленной логарифмической функции вместе с фактическими и прогнозируемыми значениями:

$$RMSLE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(log(x_i+1)) - (log(y_i+1))^2}$$

Здесь x – фактическое значение, а y – прогнозируемое значение. Это помогает уменьшить влияние выбросов, преуменьшая более высокие показатели ошибок с помощью функции логарифма. Кроме того, RMSLE помогает фикси-

ровать относительную ошибку (путем сравнения всех значений ошибок) с помощью логарифмов.

Задачи кластеризации

Индекс Данна (Dunn index) фокусируется на выявлении кластеров, которые имеют низкую дисперсию (среди всех членов в кластере) и являются компактными. Средние значения различных кластеров также должны быть далеко друг от друга:

$$Dunn\ index(U) = \min_{1\le i\le c}\left\{\min_{1\le i\le c, j\ne i}\left\{\frac{\delta(x_i, y_j)}{\max_{1\le k\le c}\{\Delta(Xk)\}}\right\}\right\}$$

Здесь δ(Xi, Yj) – межкластерное расстояние между кластерами Xi и Xj, Δ(Xk) – внутрикластерное расстояние кластера Xk.

Коэффициент силуэта (silhouette score) отслеживает, насколько каждая точка в одном кластере близка к каждой точке в других кластерах в диапазоне от -1 до +1. Вычисление коэффициента силуэта включает две метрики:

– сцепление (cohesion) a_i показывает сходство точек внутри кластера,

$$a_i = \frac{1}{|C_i| - 1}\sum_{j\in C_i, i\ne j} d(i, j),$$

где C_i – кластер, ассоциированный с точкой i, $|C_i|$ – число точек в кластере, d(i, j) – расстояние между точками;

– разделение (separation) b_i показывает, насколько кластеры не перекрываются,

$$b_i = \min_{k\ne i}\frac{1}{|C_k|}\sum_{j\in C_k} d(i, j)$$

Метрика силуэта для одной точки (рис. 6. 5) рассчитывается как

$$s_i = \frac{b_i - a_i}{\max(b_i, a_i)}$$

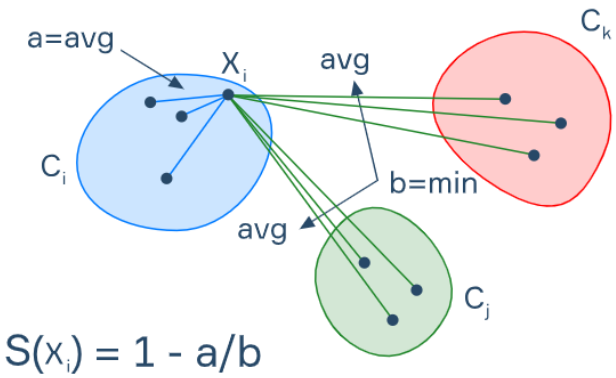


Рис. 6.5. К расчету метрики силуэта
(источник [https://hyperskill.org/learn/step/28303])

Более высокие значения силуэта (ближе к +1) указывают на то, что точки выборки из двух разных кластеров находятся далеко друг от друга, значения вблизи нуля указывают на то, что точки близки к границе решения, а значения, близкие к –1, говорят о том, что точки были неправильно отнесены к кластеру.

Метод локтя (elbow method) используется для определения количества кластеров в наборе данных путем построения графика зависимости количества кластеров на оси x от процента дисперсии, объясненной на оси y. Точка на оси x, где кривая внезапно изгибается (локоть), считается соответствующей оптимальному количеству кластеров.

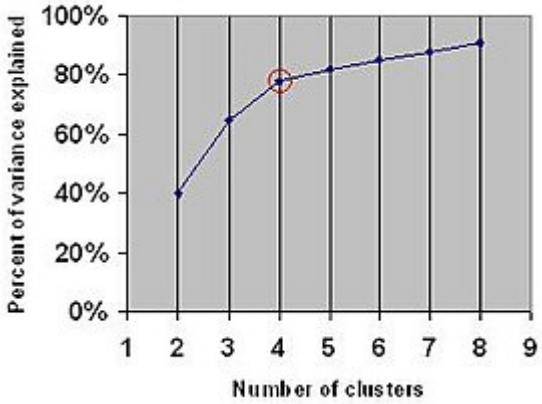


Рис. 6.6. Метод локтя для расчета числа кластеров
(источник [https://neptune.ai/blog/ml-model-evaluation-and-selection])

6.3. Обучение и оценка моделей

Хотя методология CRISP-DM рассматривает обучение моделей (Building the models) и оценку результатов (Assessing the model) как самостоятельные этапы фазы Моделирование, в реальной практике они выполняются итеративно, в тесной связке. Здесь отобранные методы и алгоритмы применяются к подготовленному набору данных. В комплекс задач этого этапа входят: установка и настройка гиперпараметров; проверка текущей версии обученной модели; разработка и тестирование ансамблевых моделей, если это необходимо; оптимизация модели.

На что следует обращать внимание в ходе обучения модели? Пайплайн процесса обучения модели детально рассмотрен в сопроводительных документах к соответствующим программным пакетам, в частности, scikit-learn [https://scikit-learn.org/stable/]. Тем не менее, обучение модели, как правило, не может быть формализовано и требует креативного подхода. Ниже приводятся некоторые практические советы.

(1) Целесообразно просматривать содержание ложноположительных (FP) и ложноотрицательных (FN) результатов, чтобы понять, правильно ли модель интерпретирует данные:

– Возможно, несовпадение происходит из-за неудачной разметки датасета – например, модель должна выделять опухоли на МРТ-изображении, а в датасете выделены полные зоны внимания врачей при постановке диагноза, которые гораздо больше самой опухоли. В этом случае нужно заменить датасет. Если данные размечаются людьми, с которыми можно поговорить, целесообразно обсудить с ними любые ошибки в разметке, а также выяснить, заметили ли они что-нибудь интересное. Возможно, такое обсуждение приведет к корректировкам в постановке задачи МО.

– Возможно, это связано с недостатком данных в конкретной части признакового пространства. Тогда необходимо достроить датасет, т.е. получить больше обучающих данных, похожих на FP или FN. Отдельный вопрос – как это сделать: достаточно ли будет применить методы аугментации или нужны новые реальные сэмплы.
– Возможно, поможет построение какой-то новой функции из уже существующих признаков – здесь, как правило, нужны дополнительные эксперименты.
– Полезно бывает добавить в модель какую-то функцию XAI, которая наглядно покажет зону внимания модели и позволит лучше понять, соответствует ли она бизнес-задаче.

(2) Целесообразно просматривать содержание пограничных точек данных, которые имели классификационный балл около 0,5. Если данные явно относятся к той или иной классификации, то можно применить к ней методы из п. (1). Если точка данных действительно неоднозначна, ее целесообразно убрать из обучающего набора.

(3) Целесообразно отслеживать важность переменных, как их формирует модель, и сравнивать ее с реальностью. Например, если модель обучается классифицировать опухоли, то, видимо, самыми важными сэмплами должны быть те, на которых эта опухоль видна лучше всего. Полезно также намеренно искать точки данных, которые противоречат признаку. Это заставляет алгоритм более тщательно оценивать то, как он использует признак.

(4) Целесообразно отслеживать ход построения ROC-кривой, чтобы увидеть динамику обучения. Если скорость обучения модели является проблемой, возможно, потребуется оптимизировать гиперпараметры, поискать более эффективный алгоритм или уменьшить обучающую выборку.

(5) Целесообразно регулярно документировать, просматривать и тестировать код извлечения признаков.

(6) Необходимо постоянно отслеживать переобучение!

Оценка и оптимизация модели. На этом этапе результаты моделирования оцениваются с технической точки зрения. В первую очередь выявляется соответствие модели ранее установленным метрикам.

Большинство моделей машинного обучения имеют некоторые гиперпараметры, которые можно настраивать или корректировать. Например, гребневая регрессия имеет гиперпараметры, такие как уровень регуляризации, модель дерева решений имеет гиперпараметры, такие как желаемая глубина или количество листьев в дереве. Процесс настройки этих гиперпараметров для повышения производительности модели известен как оптимизация настройка гиперпараметров. Оптимизация повышает производительность моделей машинного обучения, что, в свою очередь, повышает точность моделей и дает наилучшие прогнозы.

После расчета показателей оценки выбираются модели с наилучшими результатами, а затем настраиваются гиперпараметры для улучшения результатов. Для этого используются библиотеки, такие как Scikit-learn и т.д., или фреймворки, такие как Optuna, и такие подходы, как:

- поиск по сетке,
- случайный поиск,
- байесовская оптимизация,
- генетические алгоритмы.

На что следует обращать внимание в ходе оценки и оптимизации модели?

(1) Целесообразно содержательно оценить признаки, которые построила модель (особенно если это приходится пользоваться чужим кодом, извлекающий признаки).

– Например, если точность предсказания на 99% объясняется всего одним атрибутом, то стоит подумать об изменении модели.
– Можно построить для признаков диаграммы «ящик–усы», чтобы увидеть, насколько они разделяют данные. Не стоит использовать признаки, которые при этом сильно перекрываются.

(2) Нужно сравнить производительность различных версий модели.

– Если ни одна из них не достигает необходимой производительности, можно подумать об ансамблевых моделях для повышения производительности.
– Можно попытаться найти другие функции. Для алгоритмов дерева можно подумать об умных способах комбинирования функций.

(3) Были ли проблемы с качеством данных? Например, в тестовую выборку попали кейсы с пропущенными значениями, и из-за этого не для всей выборки были получены прогнозы («не вся выборка проскорилась»).

В заключение этапа целесообразно сравнить результаты обученной модели машинного обучения с базовой моделью или эвристикой. Также нужно оценить требования к работе и развертыванию построенной модели. Будет ли она соответствовать бизнес-требованиям и эксплуатационным требованиям?

Рассматривайте оценку модели как гарантию качества машинного обучения. Адекватная оценка производительности модели по метрикам и требованиям поможет понять, как модель будет работать в реальном мире.

Вопросы для самопроверки

1. Перечислите основные модели МО для использования в DS?
2. Что такое смещение модели МО?
3. Какие группы стратегий используются в МО для оценки модели?
4. Как формируется бутстрап-выборка? Для обработки каких наборов данных она применяется?
5. На каких этапах можно использовать вероятностные методы оценки модели?
6. Приведите примеры типичных метрики, используемых для оценки эффективности моделей МО при решении различных задач классификации, регрессии, кластеризации в DS?
7. На что следует обращать внимание в ходе обучения модели?

7. ЗАПУСК И ОКОНЧАНИЕ ПРОЕКТА DS

7.1. Содержание фаз запуска и окончания проекта DS

Любой проект, выполняемый дата-сайентистами, возникает из потребностей бизнеса и должен реализоваться в бизнесе. Это подчеркивают практически все методологии DS, среди которых методология CRISP-DM предлагает максимальную детализацию. А именно, в рамках CRISP-DM предусмотрена специальная фаза для поддержки запуска проекта (Бизнес-анализ) и две отдельных фазы для окончания проекта (Оценка результата и Внедрение), каждая из которых дополнительно разделяется на этапы.

Бизнес-анализ (Business Understanding) – первая фаза любого проекта DS, независимо от его объема. В этой фазе выявляются и согласуются с заказчиком основные цели проекта, а также оценивается его выполнимость с точки зрения имеющихся ресурсов. Этапы фазы:

- Определение бизнес-целей проекта (Business objectives),
- Оценка текущей ситуации (Assessing situation),
- Определение целей аналитики (Data Mining goals),
- Подготовка плана проекта (Project Plan).

Оценка результата (Evaluation). В этой фазе при активном взаимодействии с заказчиком производится оценка результатов проекта с точки зрения достижения бизнес-целей. Этапы фазы:

- Оценка результатов моделирования (Evaluating the results),
- Разбор полетов (Review the process),
- Принятие решения (Determining the next steps).

Внедрение (Deployment). В этой фазе проводится развертывание модели и мониторинг ее производительности в реальной бизнес-практике. Этапы фазы:

- Планирование развертывания (Planning Deployment),
- Настройка мониторинга модели (Planning Monitoring),
- Отчет по результатам моделирования (Final Report).

Как показывает практика, строго соблюдать предусмотренную последовательность этапов часто избыточно или просто невозможно. Скорее, их нужно рассматривать как задачи, без решения которых выполнение проекта DS невозможно.

Является ли решение этих задач непосредственной обязанностью дата-сайентиста? Ответ на этот вопрос может зависеть от целого ряда обстоятельств – например, от масштаба организации, ведущей DS-проект, от размеров самого проекта, уровня срочности, глубины требований и т.д. Как отмечалось в разделе 1.1.4, в то время как крупные организации могут позволить себе роскошь нанимать экспертов для каждой роли, предусмотренной штатом DS проекта, небольшим командам часто приходится сбалансировать различные обязанности, требуя от участников проекта выходить за рамки своих основных компетенций. Основная нагрузка в таких случаях ложится на дата-сайентиста.

В свою очередь, успешное решение задач, характерных для запуска и окончания проекта DS, определяется не столько техническим регламентом, сколько «мягкими навыками» (soft-skills) участников команды проекта – их умением взаимодействовать с людьми, вовлеченными в проект, выявлять их реальные потребности и переводить их в технически реализуемые процедуры. Полномасштабная подготовка дата-сайентиста в этом направлении выходит за рамки учебного курса DS; ниже приводятся лишь некоторые рекомендации, сформулированные на основе опыта выполнения DS-проектов.

7.2. Моделирование участия заинтересованных лиц в проекте DS

Так как проект DS направлен на решение определенной проблемы бизнеса, в нем необходимо учитывать интересы лиц, так или иначе вовлеченных в этот бизнес. Этих лиц можно разделить на три группы:

- Заказчики – руководители определенного уровня, имеющие полномочия принимать решения по вопросам, касающимся проекта (в первую очередь – стратегическим и финансовым);
- Клиенты – лица, на удовлетворение потребностей которых направлен бизнес (покупатели, потребители услуг и т.д.);
- Персонал – сотрудники бизнеса, работу которых необходимо организовать или модернизировать в результате выполнения проекта DS. Например, заказчик хочет создать программу удержания персонала в бизнесе, которая сократит текучесть кадров; в этом случае проект DS направлен на то, чтобы понять, кто остается в организации на самое короткое время и почему они уходят.

Каждая из групп имеет свои интересы, которые могут быть даже противоположными. С точки зрения DS-проектов эффективным способом единообразного их моделирования является RFM-анализ [RFM-анализ].

RFM-анализ (Recency, Frequency, Monetary – новизна, частота, денежная стоимость) представляет собой тип поведенческого таргетинга, используемый для ранжирования и сегментирования клиентов по их лояльности к компании на основе новизны, частоты и денежной стоимости транзакции. RFM-маркетинг может помочь маркетологам и владельцам бизнеса определить свою целевую аудиторию для наиболее эффективного использования своего бюджета.

Этот метод дает клиентам оценку на основе трех факторов (recency, frequency and monetary):

- Недавность (новизна). Насколько недавней была последняя покупка клиента? Клиенты, которые недавно совершили покупку, все еще будут иметь продукт в виду и с большей вероятностью купят или используют продукт снова. Компании часто измеряют недавность в днях. Но, в зависимости от продукта, они могут измерять ее в годах, неделях или даже часах.
- Частота. Как часто этот клиент совершал покупку в течение определенного периода? Клиенты, которые совершили покупку один раз, часто более склонны покупать снова. Кроме того, клиенты, которые делают покупку

впервые, могут быть хорошими целями для последующей рекламы, чтобы превратить их в более постоянных клиентов.

– Денежные траты. Сколько денег потратил клиент за определенный период? Клиенты, которые тратят много денег, с большей вероятностью потратят деньги в будущем и представляют большую ценность для бизнеса.

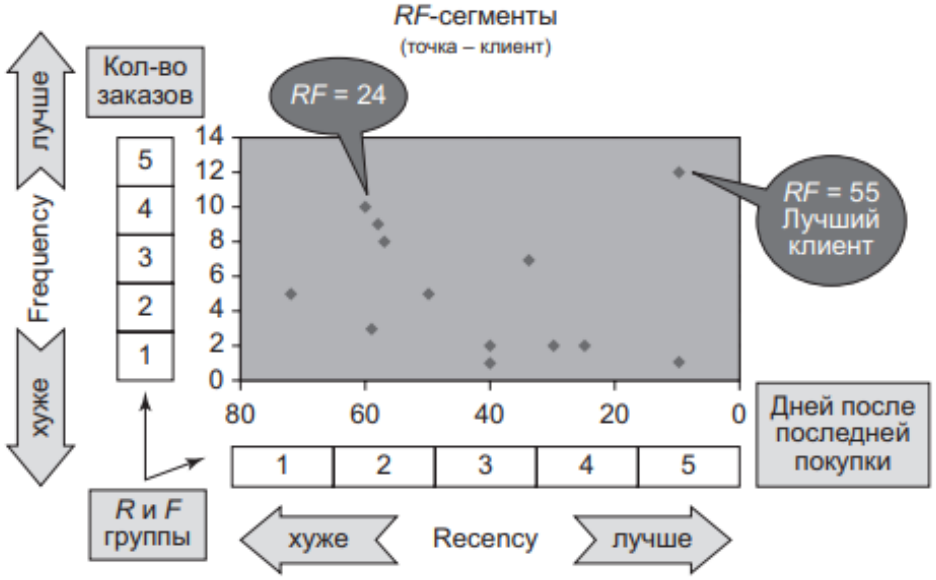


Рис. 7.1. Схема сегментации в RFM-анализе [Зыков]

Рис. 7.2. Пример детальной оценки клиентов с помощью RFM-анализа (источник [https://medium.com/@hhuseyincosgun/customer-segmentation-rfm-analysis-recency-frequency-monetary-5b29d5d45e35])

RFM-анализ оценивает клиентов по каждому из трех основных факторов по рейтинговой шкале (обычно от 1 до 5, где 5 – наивысшая). Набор из трех значений для каждого клиента называется ячейкой RFM. Основным инструментов RFM-анализа является RF-сетка (рис. 7.1). В простой системе организации

усредняют эти значения, а затем разбивают их на группы, чтобы найти самых ценных клиентов (рис. 7.2).

Некоторые компании, вместо простого усреднения придают значениям весовые коэффициенты. Например, среднестатистический клиент вряд ли купит несколько новых автомобилей за несколько лет. Но клиент, который покупает несколько автомобилей – высокочастотный клиент – должен быть очень востребован. Поэтому автосалон может соответствующим образом взвешивать значение показателя частоты.

Наконец, компании, которые не используют прямые платежи от клиентов, могут применять другие факторы в своем анализе. Например, веб-сайты и приложения, которые ценят читательскую аудиторию, количество просмотров или взаимодействие, могут использовать вместо денежных трат показатель вовлеченности, тем самым реализуя по тому же принципу анализ RFE (недавность, частота, вовлеченность).

RFM-анализ также полезен для организаций, которые не продают продукцию напрямую клиентам. Некоммерческие и благотворительные организации могут использовать RFM-анализ для поиска лучших доноров, например, поскольку те, кто жертвовал в прошлом, с большей вероятностью будут жертвовать снова в будущем.

Определение наиболее ценных сегментов RFM, по существу, позволяет выявить случайные связи в данных, т.е. является вариантом статистического анализа данных, но обладает достоинством простоты: не требуется специализированного статистического программного обеспечения , а результаты легко понятны деловым людям.

Важно подчеркнуть, что получения статистически значимых количественных оценок параметров RFM необходимы определенные условия, в том числе достаточные величины клиентской базы (по оценке авторов [RFM-анализ] – не менее 1000 человек), товарного ассортимента и времени нахождения на рынке. В то же время на качественном уровне принцип RFM сохраняется.

Более того, как показывает практика [Зыков], принцип RFM можно распространить на другие действия людей, а не только на покупки, – вероятность заболеть, вероятность вернуться на сайт, вероятность совершить противоправное действие, вероятность кликнуть на баннер, уровень лояльности сотрудников и т.д. Например, методология BADIR предлагает использовать принцип RFM для оценки эффективности шоферов такси в рамках бизнес-агрегатора по заказу такси.

Подход RFM является популярным, но далеко не единственным вариантом моделирования участия заинтересованных лиц в DS-проекте. В теории маркетинга можно найти полезные модели, например, основанные на метриках DAU (Day Active User) и MAU (Month Active User) [Active], когортном анализе [Jiang], A/B тестировании [Young] и других подходах. Кроме того, для описания участия людей в DS проектах активно развиваются data-driven подходы, основанные на различных моделях машинного обучения [Quynh, Hu, Rajak]. Регулярно проводятся соревнования Kaggle, посвященные этой проблеме [Kaggle].

7.3. Артефакты фазы запуска проекта DS

7.3.1 Бизнес-кейс

Составление бизнес-кейса является одним из первых шагов по разработке проекта DS. Бизнес-кейс – это целевой, краткосрочный или среднесрочный анализ конкретного проекта, это документ, который обосновывает реализацию проекта или инициативы. Его главная цель – оценить потенциальные выгоды, затраты и риски, предоставив лицам, принимающим решения, доказательства того, почему инвестиции оправданы.

Бизнес-кейс и бизнес-план служат разным целям. Бизнес-кейс направлен на обоснование конкретного проекта, описывая его выгоды, затраты и риски. Он фокусируется на одной инициативе, чтобы ее обеспечить одобрение и финансирование. В то же время бизнес-план представляет собой всеобъемлющий обзор всего бизнеса, содержащий подробное описание бизнес-стратегии, анализ рынка, финансовые прогнозы и операционные планы. Он служит долгосрочной дорожной картой для компании.

Хорошо структурированное бизнес-кейс (бизнес-обоснование) обычно включает в себя следующие ключевые элементы:

- Краткое изложение: Это краткий обзор бизнес-кейса, в котором освещаются бизнес-проблемы, предлагаемое решение и ожидаемые выгоды.
- Постановка проблемы: четко описывает бизнес-проблему или возможность, на решение которой направлен проект.
- Анализ вариантов: включает оценку потенциальных решений, включая их плюсы, минусы и предполагаемую стоимость.
- Рекомендуемое решение: на основе анализа предлагается следующий курс действий вместе с обоснованием его выбора.
- План реализации: включает в себя создание общей дорожной карты, описывающей сроки проекта , основные этапы, необходимые ресурсы и основные результаты.
- Финансовый анализ: включает в себя анализ затрат и выгод, который детализирует ожидаемые затраты проекта, выгоды и окупаемость инвестиций (ROI).
- Оценка рисков: помогает выявить потенциальные риски, связанные с проектом, и стратегии их снижения, включая анализ чувствительности.
- Анализ заинтересованных сторон: это обзор лиц или групп, на которых влияет проект, а также уровень их влияния и интересов.
- Заключение: Здесь суммируются основные моменты и предлагается убедительный призыв к действию для лиц, принимающих решения, по одобрению проекта.

7.3.2. Бизнес-цели

Термин «бизнес-цели» в русском языке является зонтичным, а в английском языке термины business-goals и business-objectives различаются:. business-goals задают направление, в котором компания намерена двигаться, и опреде-

ляют, чего организация хочет достичь, в то время как business-objectives определяют методы и пути, которые могут помочь бизнесу достичь этой цели.

Бизнес-цели, с которыми заказчики приходят в проект DS, часто являются слишком размытыми, аморфными и, следовательно, неизмеримыми. Для формирования конкретных и измеримых целей DS проекта необходимо тесное взаимодействие с заказчиком, для поддержки которого показала свою эффективность концепция SMART-целей [SMART].

Аббревиатура SMART (Specific, Measurable, Achievable, Relevant, Time-Bound) характеризует требования, которые предъявляются к правильно сформулированным бизнес-целям (business-goals), в виде набора вопросов:

- *Specific* (конкретная) – Что именно будет достигнуто? Какие действия вы предпримете для этого?
  Нельзя использовать неконкретные формулировки, например, «улучшить обслуживание клиентов» или «постоянно уменьшать стоимость».
- *Measurable* (измеримая) – Какие данные будут измерять цель? Какие показатели характеризуют достижение цели (насколько хорошо она достигнута)?
  В качестве показателей можно использовать метрики качества/точности, полученные суммы (доход, экономия), показатели производительности, удовлетворенность клиентов.

Важно согласовывать цели бизнеса и цели аналитики, чтобы понимание успешности проекта было одинаковое у заказчика и у команды DS проекта. Например, DS-проект состоит в повышении отклика на рекламную компанию. До проекта по ссылке переходили примерно 10% клиентов, а разработанная модель отбирает 1000 наиболее склонных к отклику клиентов, и из них переходит по ссылке каждый четвертый. Таким образом, модель демонстрирует точность (precision) в 25%, т.е. улучшение в 2.5 раза лучше «случайной» модели, но заказчик недоволен, так как он ожидал от ML-модели показателей точности 80-90%.

С этой целью полезно заранее определить базовый уровень (baseline) – показатель, достигнутый альтернативными решениями и\или конкурентами – от которого вы будете отталкиваться при сдаче DS проекта заказчику. Кроме того, полезно получить предварительную оценку ожидаемого результата – например, на уровне эвристики или простейших экспериментов, не требующих машинного обучения:

- *Achievable* (достижимая) – Достижима ли цель? Есть ли у вас необходимые навыки и ресурсы?
  Эта группа требований фокусируется на том, насколько важна для вас цель и что вы можете сделать, чтобы сделать ее достижимой, и может потребовать развития новых навыков и изменения отношения. Цель призвана вдохновлять мотивацию, а не обескураживать. Подумайте о том, как достичь цели, есть ли у команды необходимые инструменты/навыки, и если нет, то что потребуется для их достижения.
- *Relevant* (соответствующая) – Как данная цель соотносится с более широкими целями бизнеса в целом? Почему важен результат, достигаемый именно этой целью?

Например, DS проект успешно выполнен, но для его внедрения не хватает ресурсов, так как они заняты другим, более приоритетным проектом.

– *Time-Bound* (ограниченная по времени) – Каковы временные рамки для достижения цели?

Необходимо фиксировать совместно с заказчиком целевую дату для результатов. Задайте конкретные вопросы о сроке достижения цели и о том, что можно сделать за этот период времени. Если достижение цели займет три месяца, полезно определить, что должно быть достигнуто на полпути к процессу. Указание временных ограничений также создает ощущение срочности.

Ниже приводится пример удачной формулировки целей проекта в концепции SMART-goals [SMART].

Цель. Я хочу запустить и успешно завершить проект по созданию мобильного приложения:

– *Конкретная цель.* Многие люди заходят на наш текущий сайт со своих мобильных устройств. Поскольку это не адаптивный сайт, он неудобен для клиентов. Я хочу запустить мобильное приложение для сайта моей компании к концу июня, что требует участия разработчиков программного обеспечения, дизайна и маркетинга.
– *Измеримая цель.* Создание мобильного приложения для сайта нашей компании потребует много ресурсов. Чтобы это окупилось, я хотел бы иметь 50 000 установок сайта в течение шести месяцев с момента запуска. Я также хотел бы показать коэффициент конверсии 5% от клиентов, использующих мобильный сайт.
– *Достижимая цель.* Отделы, которые будут участвовать, одобрили создание мобильного приложения. Мне нужно будет управлять проектом и устанавливать контрольные точки, чтобы все были мотивированы и придерживались цели.
– *Соответствующая цель.* Улучшение клиентского опыта на мобильных устройствах является основной инициативой для моей компании в этом году.
– *Ограниченные сроки.* чтобы достичь 50 000 установок мобильного приложения и коэффициента конверсии 5% к концу финансового года, приложение необходимо будет запустить ко второму кварталу с активной маркетинговой кампанией, которая должна продолжаться до конца года.

7.3.3. Бизнес-гипотеза

Бизнес-гипотеза – это утверждение или предположение, сделанное бизнесом или предпринимателем относительно определенного аспекта их бизнеса. Обычно оно формулируется как утверждение «если–то» и служит отправной точкой для проверки и подтверждения идей.

Бизнес-гипотеза служит для перевода требований и целей в измеримую плоскость. Для этого она формулируется в терминах связи между двумя или более переменными.

Типы гипотез:

– *Простая гипотеза* оценивает базовую связь между независимой и зависимой переменной. Простые гипотезы конкретны и лаконичны, и хотя они не обязательно предсказывают причинно-следственную связь, они могут помочь вам определить, есть ли связь между двумя переменными. Примеры:
  - сотрудники, которые приносят с собой обед, тратят меньше денег в течение дня;
  - сотрудники, которые приходят на работу раньше, работают более производительно;
  - предоставление сотрудникам более полезных вариантов перекусов способствует более здоровому питанию.
– *Комплексная гипотеза* исследует связь между несколькими переменными. Она гипотеза обычно полезна в ситуациях, включающих множество противоречивых факторов, которые могут потенциально влиять друг на друга. Примеры:
  - сотрудники, работающие в компаниях, которые предоставляют праздничные бонусы и оплачиваемые отпуска, работают усерднее и обладают более высоким моральным духом,
  - регулярная оценка эффективности работы и увеличенные перерывы на обед повышают удовлетворенность работой и производительность,
  - программы физических упражнений на рабочем месте и более здоровый выбор обедов приводят к сокращению числа больничных и улучшению психического здоровья сотрудников.
– *Нулевая гипотеза* предполагает, что нет никакой связи между независимой и зависимой переменной, и является базой в задачах оценки статистических гипотез частотным методом (см. раздел 3.2.3). Примеры:
  - количество рабочих часов в день не влияет на моральный дух сотрудников,
  - планирование большего количества совещаний команды не влияет на качество результатов,
  - изменения в структуре рекламных акций не являются одной из причин более низкого показателя удержания клиентов компании в этом квартале.
– *Альтернативная гипотез*а направлена на опровержение нулевой гипотезы после того, как вы проверили свой первоначальный прогноз и узнали, что он неверен. Она может быть направленной (предсказывает конкретный результат) и ненаправленной (предсказывает более общий результат). Примеры:
  - Сотрудники работают более продуктивно, если компания предоставляет им один перерыв каждые два часа, а не один перерыв каждые четыре часа,
  - Сбалансированное трехразовое питание оказывает положительное влияние на производительность труда в отличие от небольших перекусов в течение дня,

- Удовлетворенность работой возрастает, когда сотрудники имеют доступ к различным возможностям профессионального развития в течение года, а не к одной возможности обучения раз в год.

– *Логическая гипотеза* – это рациональное заключение, которое предполагает истинность определенных события на основе предшествующих знаний или базовых наблюдений и опыта. В отличие от других гипотез, логическую гипотезу обычно невозможно проверить, но этот тип прогнозирования может быть полезен, когда вы хотите оценить проблему и быстро разработать эффективное решение. Примеры:
  - Если сотрудник опаздывает на работу, значит, на дорогах образовалась большая пробка, из-за которой возникло несколько задержек,
  - Для написания эффективного отчета о моделях покупок клиентов необходим доступ к ресурсам в папке маркетинговых исследований,
  - Если число клиентов, ответивших, что они планируют совершить еще одну покупку в компании, увеличилось, то лояльность к бренду также возросла.
– *Статистическая гипотеза* – это измеримое предсказание данных, основанное на наборе переменных. Большинство статистических гипотез используют статистический анализ для оценки репрезентативной выборки населения (см. раздел 3.1.2) и применения своих результатов к более крупной популяции. Эта гипотеза может быть полезна, если вы планируете разработать исследование или провести опрос. Примеры:
  - 70% сотрудников офисов предпочитают использовать планшет вместо компьютера при выполнении маркетинговых задач.
  - По меньшей мере 30% руководителей, скорее всего, будут использовать новую систему управления эффективностью при проведении оценок.
  - 60% руководителей, которые запрашивают обратную связь от сотрудников и разрабатывают процессы для мониторинга собственной эффективности, сообщают о том, что чувствуют себя более подготовленными к руководству другими и принятию новых задач.
– *Эмпирическая гипотеза* – это теория, основанная на предыдущих экспериментах или прошлых наблюдениях. Этот тип гипотезы может быть эффективным, если вы хотите проверить несколько других гипотез и определить корреляцию между двумя или более элементами. Примеры:
  - Сотрудники, которые приходят на работу как минимум на 15 минут раньше, также заканчивают свою работу на 15 минут раньше,
  - Новые сотрудники быстрее продвигаются по службе, если у них есть возможность встретиться с членами своей команды до первой рабочей недели.

Обобщая, можно сказать, что бизнес-гипотеза – это обоснованное предположение относительно конкретного аспекта бизнеса, сформулированное для руководства экспериментами и анализом данных. Понятно, что таких гипотез может быть много, и они образуют список (бэклог, backlog), требующий прио-

ретизации, т.е. расстановки приоритетов их исполнения. Команда DS проекта занимается этим совместно с бизнес-аналитиками. Предварительная оценка различных гипотез может помочь определить, как лучше распределить ресурсы для поддержки исследовательских усилий.

7.4. Фазы окончания проекта DS

Оценка результата (Evaluation)

В этой фазе достигнутые результаты оцениваются не с точки зрения разработчиков проекта, а уже с точки рения заказчика Здесь большой проблемой могут стать нечетко или неудачно сформулированные на первом этапе цели проекта (см. раздел 7.3). Например, если в качестве целей проекта записана методика оценки Lift для конкретных условий DS-проекта, а в качестве результата предъявляется формула для оценки, то заказчик может не принять проект или потребовать доработки.

С другой стороны, приятным сюрпризом для заказчика может быть достижение незапланированных целей, например, найденная новая информация или зависимость. Например, компания-ритейлер ориентировалась на сегмент «активная молодежь», но при прогнозировании вероятности отклика выявился еще один сегмент – «молодые родители».

Еще один этап этой фазы, важный для самой команды DS проекта, можно назвать разбором полетов. Здесь производится ретроанализ хода выполнения процесса, выявляются его сильные и слабые стороны, допущенные ошибки возможность их избежать в будущем. Отдельного внимания заслуживают выдвинутые, но не использованные гипотезы – возможно, они пригодятся для будущих проектов.

Фаза заканчивается официальным принятием решения о дальнейшей судьбе разработанной модели. Если она устраивает заказчика, то нужно переходить к ее внедрению; если требуются доработки, то их нужно сделать до перехода к внедрению.

Внедрение (Deployment)

Перед началом проекта с заказчиком всегда оговаривается способ поставки модели – от уровня модифицированной базы клиентов до аналитического решения, предполагающего интеграцию в информационную систему бизнеса.

Под внедрением модели может пониматься как физическое добавление функционала, так и инициирование изменений в бизнес-процессах компании.

Развертывание (Deployment) модели – важный процесс, требующий отдельного планирования и мониторинга. Здесь определяется и фиксируется, что именно и в каком виде будет внедряться, а также подготавливается техническая документация по внедрению (описание процесса с паролями и пр.) и по сопровождению внедренной модели. Уточняется и апробируется интерфейс между внедряемой моделью и существующий информационной системой, а также ее интерфейс для пользователей – например, на экране сотрудника колл-центра будет показываться склонность клиента к подключению дополнительных услуг.

Процесс внедрения модели носит название «Операционализация модели» (Model Operations, или Model Ops). Процесс Model Ops (также известный как ML Ops) гарантирует, что модели продолжают приносить пользу организации. Они также предоставляют критически важную информацию для управления потенциальными рисками принятия решений на основе моделей, даже если основные деловые и технические условия меняются.

Model Ops – это кросс-функциональный, совместный, непрерывный процесс, который фокусируется на управлении моделями машинного обучения, чтобы сделать их повторно используемыми и высокодоступными посредством повторяющегося процесса развертывания. Четыре основных шага в процессе Model Ops – создание, управление, развертывание/интеграция и мониторинг – образуют повторяемый цикл, который можно использовать для повторного использования моделей в качестве программных артефактов. Более того, Model Ops охватывает различные аспекты управления, такие как управление версиями моделей, аудит, мониторинг и обновление, чтобы гарантировать, что они по-прежнему обеспечивают положительную бизнес-ценность при изменении условий. Сейчас существует несколько средств, реализующих основную функциональность MLOps: MLflow, Neptune AI, Kubeflow (для развертывания в Kubernetes), Weights & Biases (W&B), ClearML.

Существует четыре варианта развертывания ML моделей (рис. 7.3), которые определяются тем, на каких данных модель обучается – на заранее собранных (Offline Training) или на потоковых (Online Training), а также тем, какие данные используются в ходе эксплуатации модели – заранее заготовленные (Offline или Batch Prediction) или поступающие из текущего запроса (Real-time или On-demand Prediction).

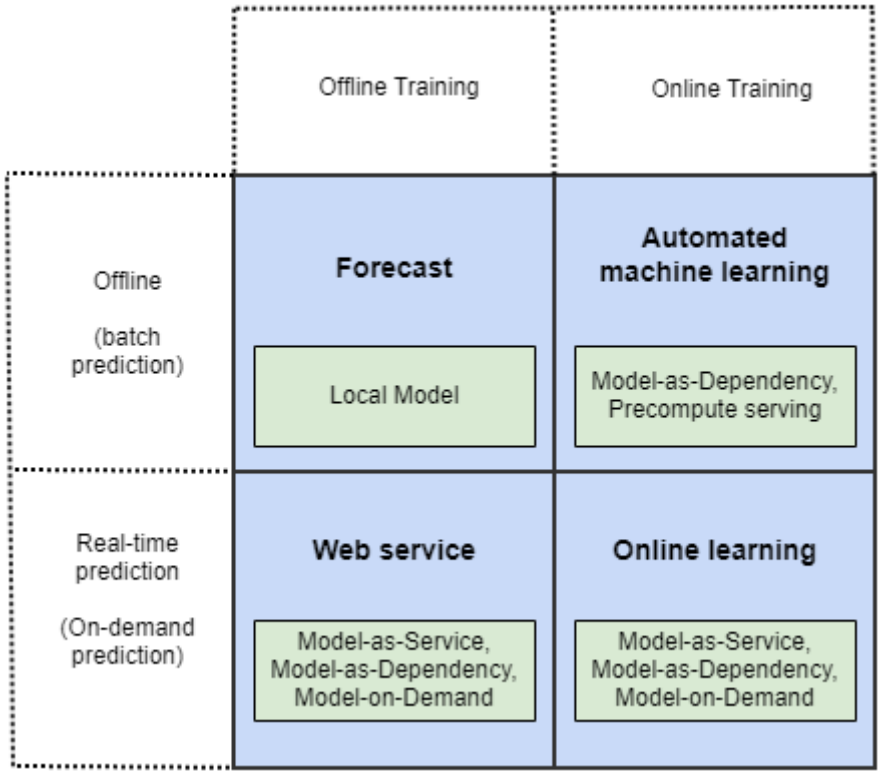


Рис. 7.3. Варианты развертывания модели
(источник [https://habr.com/ru/articles/795985/])

Для реализации этих условий существуют специальные схемы:

Forecast – используется в научных исследованиях, но не в промышленных системах;

Web service – ML-модель обучается на исторических данных и реализует предсказание в реальном времени на вновь поступающих данных. Данные могут поступать посредством запросов через REST API (Model-as-Service), модель может быть встроена в приложение непосредственно (Model-as-Dependency) или через message broker, регулирующий очередь запросов (Model-on-Demand) (рис. 7.4).

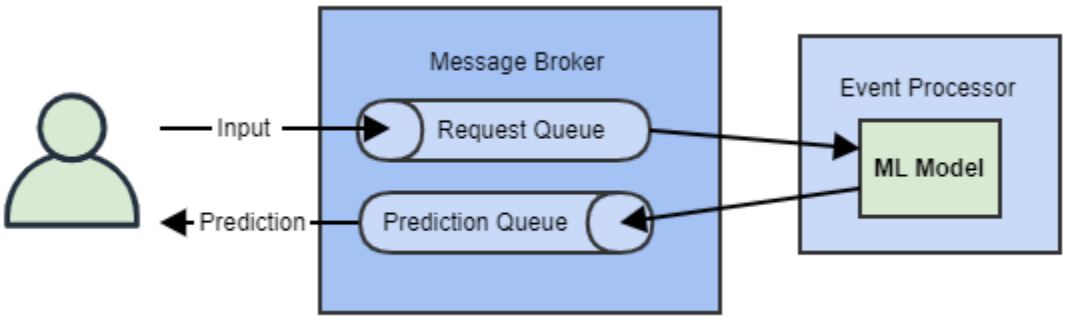


Рис. 7.4. Архитектура шаблона Model-on-Demand

Online learning – используется на потоковых данных, когда модель должна постоянно перенастраиваться (переучиваться). Это может происходить либо непосредственно на промышленной системе, либо на параллельно подключенной модели.

Automated machine learning – происходит не только автоматическое обучение и оптимизация результатов, но и автоматический подбор ML-модели. Такая схема на сегодня реализуется только крупными провайдерами ИИ, такими как Google или Microsoft.

В большинстве случаев фаза внедрения не является зоной ответственности дата-сайентиста, хотя, как уже отмечалось в разделе 7.1, его участие всегда полезно, а иногда (особенно в небольших проектах), является необходимостью. Но в любом случае завершающим этапом DS проекта является отчет о результатах моделирования, который представляется заказчику и веем заинтересованным лицам и является, наряду с техническим заданием, основным документом проекта. В написании отчета дата-сайентист принимает самое непосредственное участие.

7.5. Практика работы дата-сайентиста в крупной компании

В каждой компании, использующей анализ данных, есть своя специфика организации процесса DS, причем специфические детали зачастую важнее общих требований и теоретических положений. Поэтому в заключение приводим описание типового процесса выполнения DS-проекта в крупной ритейл-компании, которая занимается продажей продуктов питания в своей торговой

сети. Описание составлено дата-сайентистом, непосредственно работающим в этой компании, и приводится почти в авторской редакции (чтобы сохранить те самые специфические детали, которые и составляют его главную ценность!).

«Вообще, если и выделять общую схему во взаимодействиях с заказчиком, то она примерно такая.

(1) Запуск проекта.

Заказчик приходит с запросом, который обычно звучит одним из следующих образов:

- Мы провели испытания нового подхода/методики/чего-то ещё, нужна аналитика результатов – статзначимость, сколько мы приобрели/потеряли и т.д.
- Нам нужно знать текущее состояние дел по такой-то метрике.
- У нас есть данные, нужен прогноз того, что будет.
- Нужна модель для автоматизации определенного подпроцесса (в этом случае данные могут как предоставить, так и их нужно будет найти).

Если данные уже предоставлены, то они просматриваются, и с заказчиком обсуждаются детали и контекст работы модели: в каком формате требуется отчёт (если он предполагается), какой объём данных должна обрабатывать модель, какая точность/надёжность требуется от модели, какого качества будут данные на практике и т.д.. При необходимости с заказчиком обсуждается расширение объёма данных. Если данные не предоставлены, то сначала уточняются, насколько можно вопросы, перечисленные выше, а потом происходит поиск данных под заданные условия.

(2) Выполнение проекта.

Здесь коммуникация с заказчиком достаточно фрагментарная – происходит отбор/обработка/аугментация данных, подбор и обучение моделей, оформление результатов. При возникновении трудностей возвращаемся к предыдущему пункту, чтобы уточнить детали для большей специализации моделей или донабора/улучшения качества данных.

Обработка данных обычно происходит в рамках тех самых штук с форматами, которые мы обсуждали (имеются в виду различные варианты обработки данных, описанные в разделе 5) – только это оформляется в виде алгоритма, удовлетворяющего тому, как заказчик описывал свои данные. Когда заказчик хорошо подготавливает данные и они стандартизованы, эта часть простая, в ином случае – сложная из-за необходимости обработки тонн краевых случаев и т.д..

Подбор модели происходит уже на обработанных данных и представляет из себя смесь из известных методов (описанных в разделе 6) и интуиции из практики. Модели нередко берутся сначала из открытых источников и всяческих соревнований (типа Kaggle), а потом «допиливаются напильником».При необходимости добавляются специфические вариации – например, я заменил Isolation Forest в одном из заказов на его версию, работающую с кластерами, когда заметил, что больно уж мало аномалий выделялось. Если модель в итоге работает хорошо, то результаты оформляются и презентуются заказчику. Если нет и совсем никак долгое время не удается вытянуть нужный уровень каче-

ства, обычно смотрят на качество данных более внимательно или сужают задачу вместе с заказчиком.

(3) Сдача заказа.

Происходит сдача заказа и демонстрация результатов заказчику. Обычно на этом этапе он уже «плюс-минус в курсе» хода работы, поэтому демонстрация больше формальная. Чаще всего потом заказчик берёт модель в продакшн и, по необходимости, возвращается за доработками. Если доработки возникли больше из-за недоработок программиста, то они делаются бесплатно, если из-за неуточнённого ТЗ, то за доплату».

Авторы пособия предполагают, что по мере использования пособия число подобных кейсов с описаниями реальных практик работы дата-сайентистов будет возрастать, и планируют вносить их в электронную версию пособия которая размещена на сайте Университета ИТМО.

Вопросы для самопроверки

1. Что представляет собой RFM-анализ? Для каких организаций он может быть полезен?
2. В чем разница между бизнес-кейсом и бизнес-планом?
3. Какие ключевые моменты включает в себя бизнес-кейс?
4. Приводите пример формулировки целей проекта в концепции SMART-целей?
5 Для чего формируется бизнес-гипотеза? Какие типы гипотез бывают?
6. На каком этапе подготавливается техническая документация по внедрению (описание процесса с паролями и пр.) и по сопровождению внедренной модели?
7. Опишите четыре варианта развертывания ML моделей?
8. Приведите пример типового процесса выполнения DS-проекта в компании?

ЗАКЛЮЧЕНИЕ

Предметная область «Data Science» является одной из самых востребованных приложений ИТ в бизнесе. Она уже имеет набор апробированных и документированных методологий, поддерживающих процесс выполнения DS-проекта с разной степенью детализации. Однако, помимо знакомства с методологиями DS проекта и поддерживающими их технологическими решениями, успешному дата-сайентисту необходимо понимание научных основ всех этих решений, а также высокий уровень креативности и любопытство – другими словами, весь комплекс качеств, которые отличают настоящего исследователя.

В настоящем пособии сделана попытка отразить пересечение всех этих тенденций. Насколько известно авторам, материал такого охвата предлагается русскоязычным читателям впервые.

С другой стороны, внимательный читатель мог заметить, что в пособии не представлено ни одной строчки кода, а все базовые концепции – которых в пособии очень много – разъясняются «на пальцах». По собственному опыту авторов пособия, именно такой подход формирует достаточно широкий научный кругозор, который, в сочетании с логикой «здравого смысла», позволяет успешно работать в такой разнообразной и междисциплинарной предметной области, как DS.

Если посмотреть на список использованных источников, то можно заметить, что большинство технологических решений, используемых в DS, представляют собою симбиоз предметных знаний конкретной области и возможностей машинного обучения и искусственного интеллекта. Ключевым здесь является слово «симбиоз», т.е. от дата-сайентиста требуется умение быстро погружаться в новые для себя предметные области, осознавать их задачи, переводить их на язык ИТ и быстро находить решения, эффективные не только с точки зрения ИТ-показателей, но и с точки зрения бизнеса, работающего в этих предметных областях.

Учитывая репутацию, которую имеет Университет ИТМО в области информационных технологий и их приложений, можно надеяться, что нынешние студенты – адресаты настоящего пособия – будут успешно отвечать на эти вызовы.

ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ

[Agrawal] Agrawal R., Imielinski T., Swami A.. 1993. Mining Associations between Sets of Items in Massive Databases. In Proc. of the 1993 ACM-SIGMOD Int'l Conf. on Management of Data, Pp. 207-216.

[Akinshin] Akinshin A. Finite-sample bias-correction factors for the median absolute deviation based on the Harrell-Davis quantile estimator and its trimmed modification. arXiv:2207.12005 (stat) 25 Jul 2022.

[Amazon] Amazon Web Services (AWS). What is data science? (Электронный ресурс). https://aws.amazon.com/what-is/data-science/?nc1=h_ls. [Дата обращения – 11.08.2024].

[Arbelaitz] Arbelaitz O., et al. An extensive comparative study of cluster validity indices. Pattern Recognition. Volume 46, Issue 1, January 2013, Pp. 243-256.

[Averianova] Averianova I. Бета-распределение: интуиция, примеры, вывод. Jan 29, 2020. (Электронный ресурс) https://medium.com/nuances-of-programming/бета-распределение-интуиция-примеры-вывод-4662b929305e# [Дата обращения – 11.08.2024].

[Awan] Awan A.A. Top 10 Data Science Tools To Use in 2024. DataCamp blog. Nov 2023. (Электронный ресурс) https://www.datacamp.com/blog/top-data-science-tools [Дата обращения – 11.08.2024].

[Azevedo] Azevedo A., Santos M.F. KDD, SEMMA and CRISP-DM: a parallel overview. In Proceedings of the IADIS European Conference on Data Mining 2008, Pp. 182-185.

[Becht] Becht E., McInnes L., Healy J. et al. Dimensionality reduction for visualizing single-cell data using UMAP. Nat Biotechnol 37, 38–44 (2019). https://doi.org/10.1038/nbt.4314.

[Bell] Bell G., Hey T., Szalay A. (2009). Computer Science: Beyond the Data Deluge. Science. 323 (5919): 1297–1298. doi:10.1126/science.1170411. ISSN 0036-8075. PMID 19265007. S2CID 9743327.

[Вех] Вех Т. Потрясающе красиво: как отобразить десятки признаков в данных. Блог компании Scillfactory. 28 сен 2021. (Электронный ресурс) https://habr.com/ru/companies/skillfactory/articles/580154/ [Дата обращения – 11.08.2024].

[Biswas] Biswas S., Bordoloi M., Purkayastha B. Review on Feature Selection and Classification using NeuroFuzzy Approaches. Int. J. Applied Evolutionary Computation. V.7. Is.4 2016.

[Breunig] Breunig M., Kriegel H., Ng R., Sander J. LOF: identifying densitybased local outliers, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, vol. 29, ACM, 2000, Pp. 93–104.

[Cannavò] Cannavò, F., Nunnari, G. On a Possible Unified Scaling Law for Volcanic Eruption Durations. Sci Rep 6, 22289 (2016). https://doi.org/10.1038/srep22289.

[Cao] Cao L. (2017). Data Science: A Comprehensive Overview. ACM Computing Surveys. 50 (3): 43:1–43:42. arXiv:2007.03606. doi:10.1145/3076253. ISSN 0360-0300. S2CID 207595944.

[Chalapathy] Chalapathy R., Chawla S. (2019). Deep Learning for Anomaly Detection: A Survey. CoRR, abs/1901.03407.
[Chen] Liang-Chieh Chen, et al. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. Proceedings of the European Conference on Computer Vision (ECCV), 2018, Pp. 801-818.
[Clauset] Clauset A., Shalizi C.R., Newman M.E.J. Power-law distributions in empirical data. arXiv:0706.1062v2 [physics.data-an] 2 Feb 2009.
[Cotton] Cotton R. Data Visualization Cheat Sheet. Блог компании Datacamp. Apr 22, 2022. (Электронный ресурс) https://www.datacamp.com/cheat-sheet/data-viz-cheat-sheet [Дата обращения – 11.08.2024].
[Cui] Cui Y., et al. DETC2020-22591 A weighted network modeling approach for analyzing product competition. IDETC-CIE Int. Design Engineering Tech. Conf. & Comp. and Inf. in Engineering Conf., August 2020.
[Cunha] Cunha M., Mendes R., Vilela J.P. A survey of privacy-preserving mechanisms for heterogeneous data types. Computer Science Review, V. 41, 2021, 100403, https://doi.org/10.1016/j.cosrev.2021.100403.П.
[Danyluk] Danyluk A., Leidig P. (2021). Computing Competencies for Undergraduate Data Science Curricula (PDF). ACM Data Science Task Force Final Report (Report).
[Das] Das S., et al. Active Anomaly Detection via Ensembles: Insights, Algorithms, and Interpretability. arXiv:1901.08930v1 [cs.LG] 23 Jan 2019.
[Delignette-Muller] Delignette-Muller M.L. (2015) fitdistrplus: an R package for fitting distributions. Journal of Statistical Software 64(4).
[Dhar, 2013] Dhar, V. (2013). "Data science and prediction". Communications of the ACM. 56 (12): 64–73. doi:10.1145/2500499.
[Donoho] Donoho D. (2017). 50 Years of Data Science. Journal of Computational and Graphical Statistics. 26 (4): 745–766. doi:10.1080/10618600.2017.1384734. S2CID 114558008.
[Dutang] CRAN Task View: Extreme Value Analysis. 2023-11-04. https://cran.r-project.org/web/views/ExtremeValue.html [Дата обращения – 11.08.2024].
[Emmert-Streib] Emmert-Streib F., Dehmer M. (2018). Defining data science by a data-driven quantification of the community. Machine Learning and Knowledge Extraction. 1: 235–251. doi:10.3390/make1010015.
[Farmer] Farmer D.J., Geanakoplos J. Power laws in economics and elsewhere. May14, 2008. https://oms-inet.files.svdcdn.com/staging/files/powerlaw3.pdf [Дата обращения – 11.08.2024].
[Fayyad] Fayyad U., Piatetsky-Shapiro G., Smyth P. (1996). From Data Mining to Knowledge Discovery in Databases. AI Magazine, 17(3), 37. https://doi.org/10.1609/aimag.v17i3.1230.
[Gamma] Gamma distribution: Understanding Probability Distributions. Блог компании FasterCapital. 25 Jun 2024. (Электронный ресурс) https://fastercapital.com/content/Gamma-distribution--Understanding-Probability-Distributions.html [Дата обращения – 11.08.2024].

[Goldstein] Goldstein M., Dengel A. Histogram-based Outlier Score (HBOS) : A fast Unsupervised Anomaly Detection Algorithm. DFKI. 2012. https://www.dfki.de › import › 6431_HBOS-poster.

[Guo] Guo N., et al. An Efficient Query Algorithm for Trajectory Similarity Based on Fréchet Distance Threshold. October 2017ISPRS International Journal of Geo-Information 6(11):326. DOI:10.3390/ijgi6110326.

[Hayashi] Hayashi C. (1998). What is Data Science? Fundamental Concepts and a Heuristic Example. Data Science, Classification, and Related Methods. Studies in Classification, Data Analysis, and Knowledge Organization. Springer Japan. pp. 40–51. doi:10.1007/978-4-431-65950-1_3. ISBN 9784431702085.

[Hey] Hey T., Tansley S., Tolle K.M. (2009). The Fourth Paradigm: Data-intensive Scientific Discovery. Microsoft Research. ISBN 978-0-9825442-0-4. Archived from the original on 20 March 2017.

[Hilbert] Hilbert M. Scale-free power-laws as interaction between progress and diffusion. Complexity : journal. 2013. Vol. 19, no. 4. P. 56–65. doi:10.1002/cplx.21485.

[Hilbert] Hilbert M. Scale-free power-laws as interaction between progress and diffusion. Complexity. 2013. doi: 10.1002/cplx.21485.

[Huang] Huang H., Mehrotra K., Mohan C.K. (2013). Rank-based outlier detection. Journal of Statistical Computation and Simulation, 83(3), 518–531. https://doi.org/10.1080/00949655.2011.621124.

[Hubert] Hubert M., Vandervieren E. An Adjusted Boxplot for Skewed Distributions.. Computational Statistics & Data Analysis 52(12):5186-5201. August 2008. DOI:10.1016/j.csda.2007.11.008.

[Jain] Jain P., Puneet S. (November 2014). Behind Every Good Decision: How Anyone Can Use Business Analytics to Turn Data Into Profitable Insight. American Management Association. ISBN 978-0-8144-4921-9.

[JTBD] Что такое концепция Jobs to Be Done и как использовать её в маркетинге. Блог компании Яндекс-бизнес. (Электронный ресурс) https://business.yandex/praktika/jobs-to-be-done/ [Дата обращения – 0.11.2023].

[Karpathy] Karpathy A., et al. Generative models. Блог компании OpenAI. June 16, 2016. https://openai.com/index/generative-models/ [Дата обращения – 11.08.2024].

[Kibardin] Kibardin E. AI for AI (artificial insemination) — Deep Topological Analysis for sensor data. Блог компании DataRefiner. March 10, 2020. https://datarefiner.com/feed/ai-for-ai [Дата обращения – 11.08.2024].

[Kibardin] Kibardin E. Why you should use Topological Data Analysis over t-SNE or UMAP? Блог компании DataRefiner. April 17, 2023. https://datarefiner.com/feed/why-tda [Дата обращения – 11.08.2024].

[Koehrsen] Koehrsen W. Histograms and Density Plots in Python. Mar 23, 2018. (Электронный ресурс) https://towardsdatascience.com/histograms-and-density-plots-in-python-f6bda88f5ac0 [Дата обращения – 11.08.2024].

[Kozyrkov] Kozyrkov C. What’s the difference between analytics and statistics? Aug 24, 2019. (Электронный ресурс) https://towardsdatascience.com/whats-the-

difference-between-analytics-and-statistics-cd35d457e17 [Дата обращения – 11.08.2024].
[Lee] Lee S.Y., Kim J.H.T. Exponentiated Generalized Pareto Distribution: Properties and applications towards Extreme Value Theory. arXiv:1708.01686v1 [math.ST] 4 Aug 2017.
[Loginom Skills] Метод главных компонент (Principal component analysis). Блог компании Loginom. 15.08.2024. (Электронный ресурс) https://wiki.loginom.ru/articles/principal-component-analysis.html [Дата обращения – 11.08.2024].
[Maaten] van der Maaten L.J.P., Hinton G.E. Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research 9(Nov):2579-2605, 2008. [Хмельков] Хмельков И. Препарируем t-SNE. 21 сен 2015. https://habr.com/ru/articles/267041/.
[McInnes] McInnes L., Healy J., Melville J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426v3 [stat.ML] 18 Sep 2020.
[Mike] Mike K., Hazza O. (2023). "What is Data Science?". Communications of the ACM. 66 (2): 12–13. doi:10.1145/3575663. ISSN 0001-0782.
[NEERC] Метод главных компонент (PCA). Викиконспекты. (Электронный ресурс) 15.08.2024. https://neerc.ifmo.ru/wiki/index.php?titleМетод_главных_компонент%28PCA%29&mobileaction=toggle_view_desktop [Дата обращения – 11.08.2024].
[OLAP] Is OLAP still relevant today? 16.08.2024. (Электронный ресурс) https://www-quora-com.translate.goog/Is-OLAP-still-relevant-today?_x_tr_sl=en&_x_tr_tl=ru&_x_tr_hl=ru&_x_tr_pto=rq#:~:text=OLAP%20is%20usually%20referred%20to,build%20quite%20complex%20analytics%20models [Дата обращения – 11.08.2024].
[Ou] Ou G., et al. Introduction to Data Science. Beijing: World Scientific Publishing, 2024. ISBN 978981126390.
[Peng] Peng X., Li J. & Jiang S. Unified uncertainty representation and quantification based on insufficient input data. Structural and Multidisciplinary Optimization. December 2017. DOI: 10.1007/s00158-017-1722-4.
[Piatetsky] Piatetsky G. CRISP-DM, still the top methodology for analytics, data mining, or data science projects. October 28, 2014. (Электронный ресурс) https://www.kdnuggets.com/2014/10/crisp-dm-top-methodology-analytics-data-mining-data-science-projects.html [Дата обращения – 11.08.2024].
[Piccini] Piccini N. 101 machine learning algorithms for data science with cheat sheets. June 9, 2022. (Электронный ресурс) https://datasciencedojo.com/blog/machine-learning-algorithms/ [Дата обращения – 11.08.2024].
[Pimentel] Pimentel M.A.F., Clifton D.A., Clifton L., Tarassenko L. A review of novelty detection. Signal Processing 99, 2014, Pp. 215–249.

[Plotnikova] Plotnikova V, Dumas M, Milani F. Adaptations of data mining methodologies: a systematic literature review. PeerJ Comput Sci. 2020 May 25;6:e267. doi: 10.7717/peerj-cs.267. PMID: 33816918; PMCID: PMC7924527.

[Pratt] Pratt M.K. 18 data science tools to consider using in 2024. 25 Jan 2024. (Электронный ресурс) https://www.techtarget.com/searchbusinessanalytics/feature/15-data-science-tools-to-consider-using [Дата обращения – 11.08.2024].

[Pudjihartono] Pudjihartono N, Fadason T, Kempa-Liehr AW and O'Sullivan JM (2022) A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction. Front. Bioinform. 2:927312. doi: 10.3389/fbinf.2022.927312.

[Rahm] Rahm, E. and Do, H.H. (2003) Data Cleaning Problems and Current Approaches. IEEE Bulletin on Data Engineering, 7, Pp. 182-185.

[Reed] Reed W.J. (2002). "On the rank-size distribution for human settlements". Journal of Regional Science. 42 (1): 1–17. Bibcode:2002JRegS..42....1R. doi:10.1111/1467-9787.00247.

[Rousseeuw] Rousseeuw P.J., Croux C. Alternatives to the Median Absolute Deviation. Journal of the American Statistical Association. Volume 88, 1993. Issue 424. Pp. 1273-1283.

[Sargent] Sargent T.J. and Stachurski J. Heavy-Tailed Distributions. In: A First Course in Quantitative Economics with Python. (Электронный ресурс) https://intro.quantecon.org/heavy_tails.html [Дата обращения – 11.08.2024].

[SAS] SAS Enterprise Miner. Overview. (Электронный ресурс) https://www.sas.com/en_gb/software/enterprise-miner.html [Дата обращения – 11.08.2024].

[Scheler] Scheler G. (2017). "Logarithmic distributions prove that intrinsic learning is Hebbian". F1000Research. 6: 1222. doi:10.12688/f1000research.12130.2.

[SendPulse]. Дашборд как интерактивная альтернатива табличным отчетам. Блог компании SendPulse. 15 мая 2024. (Электронный ресурс) https://sendpulse.com/ru/blog/dashboard#:~:text=Так_в_чем_же_разница,отслеживать_данные_в_реальном_времени [Дата обращения – 11.08.2024].

[Sevcik] Sevcik C. First derivatives at the optimum analysis (fdao): An approach to estimate the uncertainty in nonlinear regression involving stochastically independent variables. January 2019. DOI:10.13140/RG.2.2.23401.75365.

[Shahid] Shahid A. 7 Data Quality Metrics to Assess Your Data Health. April 24th, 2024. (Электронный ресурс) https://www.astera.com/type/blog/data-quality-metrics/ [Дата обращения – 11.08.2024].

[Shearer] Shearer C. The CRISP-DM model: the new blueprint for data mining, J Data Warehousing (2000); 5, Pp. 13—22.

[Sikder] Sikder N.K., Batarseh F.A. 7 - Outlier detection using AI: a survey. AI Assurance, Academic Press, 2023, P. 231-291, ISBN 9780323919197.

[SkillFactory] Практики и инструменты методологии DevOps. Блог компании SkillFactory. 14.08.2024. (Электронный ресурс) https://blog.skillfactory.ru/glossary/devops/ [Дата обращения – 11.08.2024].

[SmartRisk] Understanding SmartRisk: An Advanced Approach to Portfolio Risk Measurement. Блог компании Covisum. 25.08.2024. (Электронный ресурс) https://www.covisum.com/knowledge-base/measuring-risk [Дата обращения – 11.08.2024].

[Spradlin] Spradlin D. Are You Solving the Right Problem? Harvard Business Review, 2012. (Электронный ресурс) https://hbr.org/2012/09/are-you-solving-the-right-problem [Дата обращения – 11.08.2024].

[Stanly] Stanly H., Shalinie M.S., Paul R. A review of generative and non-generative adversarial attack on context-rich images. Engineering Applications of Artificial Intelligence. Volume 124, September 2023, 106595.

[Su] Su S., et al. (2019). An Efficient Density-Based Local Outlier Detection Approach for Scattered Data. IEEE Access, 7, Pp. 1006–1020.

[Tuychlev] Tuychlev B. A Comprehensive Introduction to Anomaly Detection. Nov 28, 2023. (Электронный ресурс) https://www.datacamp.com/tutorial/introduction-to-anomaly-detection [Дата обращения – 11.08.2024].

[Univariate] Univariate Distribution Relationships. (Электронный ресурс) 26.08.2024. https://www.math.wm.edu/~leemis/chart/UDR/UDR.html [Дата обращения – 11.08.2024].

[Urbanowicz] Urbanowicz R.J., et al. Relief-based feature selection: Introduction and review. Journal of Biomedical Informatics, Volume 85, 2018, Pages 189-203, https://doi.org/10.1016/j.jbi.2018.07.014.

[Wang B.] Wang H., Bah M.J., Hammad M. Progress in Outlier Detection Techniques: A Survey. IEEE Access, Aug. 2, 2019. DOI 10.1109/ACCESS.2019.2932769.

[Wang X.] Wang X., Varol O., Eliassi-Rad T. L2P: Learning to place for estimating heavy-tailed distributed outcomes. arXiv:1908.04628v3 [cs.LG] 12 Oct 2023.

[Wang Y.] Wang Y., et al. Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP for Data Visualization. Journal of Machine Learning Research 22, 2021, Pp. 1-73.

[Wolpert] Wolpert D. "The Lack of A Priori Distinctions between Learning Algorithms", Neural Computation, 1996, Pp. 1341–1390.

[Xiao] Xiao Z., et al. Monocular Localization with Vector HD Map (MLVHM): A Low-Cost Method for Commercial IVs. Sensors 2020, 20, 1870; doi:10.3390/s20071870.

[Zhao] Zhao, et al. Maximum Relevance and Minimum Redundancy Feature Selection Methods for a Marketing Machine Learning Platform. 2019. https://arxiv.org/abs/1908.05376.

[Zia] Zia A. et al. Topological deep learning: a review of an emerging paradigm. Artifcial Intelligence Review (2024) 57:77 https://doi.org/10.1007/s10462-024-10710-9. Published online: 29 February 2024.

[Айвазян, 1983] Айвазян С.А., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: основы моделирования и первичная обработка данных. М.: Финансы и статистика, 1983.

[Айвазян, 1983] Айвазян С.А., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: исследование зависимостей. М.: Финансы и статистика, 1985.

[Акимов] Акимов С.С., Трипкош В.А. Сравнение некоторых методов оценки тяжести хвостов при идентификации закона распределения. Современные наукоемкие технологии. 2021. № 2. С. 9–13.

[Гатман] Гатман А.Дж., Голдмейер Дж. Разберись в Data Science : как освоить науку о данных и научиться думать как эксперт. М.: Эксмо, 2023. 304 с. ISBN 978-5-04-174810-4.

[Гмурман] Гмурман В.Е. Теория вероятностей и математическая статистика: Учебное пособие для вузов. М.: Высшая школа, 2003. 479 с. ISBN 5-06-004214-6.

[Григорьев] Григорьев Ю. Д. Методы оптимального планирования эксперимента: линейные модели: Учебное пособие. — СПб.: Издательство «Лань», 2015. — 320 с.

[Дэви] Дэви С., Арно М., Мохамед А. Основы Data Science и Big Data. Python и наука о данных. СПб: Питер, 2017. 336 с.

[Елисеева] Елисеева И.И., Юзбашев М.М. Общая теория статистики: Учебник. Под ред. И. И. Елисеевой. 5-е изд., перераб. и доп. М.: Финансы и статистика, 2004.

[Жолудова] Жолудова О. Частотный и байесовский подходы к A/B тестированию: подробное сравнение | Глава 7. Блог компании Dynamic Yield. Апр 21, 2022. (Электронный ресурс) https://ux-journal.ru/bayesian-testing.html. [Дата обращения – 11.08.2024].

[Зыков] Зыков Р. Роман с Data Science. Как монетизировать большие данные. СПб.: Питер, 2021. 320 с. ISBN 978-5-4461-1879-3.

[Кацер] Кацер Ю. Обзор метрик обнаружения аномалий (плюс много дополнительной информации). Блог компании Росатом 9 сен 2022. (Электронный ресурс) https://habr.com/ru/companies/rosatom/articles/687270/ [Дата обращения – 11.08.2024].

[Келлехер, 2020] Келлехер Дж., Тирни Б. Наука о данных. М.: Альпина, 2020. 238 с.

[Кобзарь] Кобзарь А. И. Прикладная математическая статистика. Справочник для инженеров и научных работников. — М.: Физматлит, 2006. — 816 с.

[Кодкамп] 5 реальных примеров биномиального распределения/ Блог компании Кодкамп. 17 авг. 2022 г. (Электронный ресурс) https://www.codecamp.ru/blog/binomial-distribution-real-life-examples/ [Дата обращения – 11.08.2024].

[Корнелюк] Корнелюк А. Сегментация целевой аудитории. Блог компании Marquiz. (Электронный ресурс) 05.07.2024. https://www.marquiz.ru/blog/segmentaciya-celevoy-auditorii [Дата обращения – 11.08.2024].

[Коточигов] Коточигов К. CRISP-DM: проверенная методология для Data Scientist-ов. 17 мая 2017. (Электронный ресурс) https://habr.com/ru/companies/lanit/articles/328858/ [Дата обращения – 11.08.2024].

[Кузнецов] Кузнецов В.И. Статистическая совокупность. Большая Российская энциклопедия. 27.03.2023. (Электронный ресурс) https://bigenc.ru/c/statisticheskaia-sovokupnost-226cde [Дата обращения – 11.08.2024].

[Кузнецов] Кузнецов Е., Кузьмин Л., Слуцкая Ю. Чем Data Scientist отличается от аналитика данных. Блог Яндекс-практикума. 24.08.2022. (Электронный ресурс) https://practicum.yandex.ru/blog/otlichiya-analitika-dannyh-i-data-scientist/ [Дата обращения – 11.08.2024].

[Кушнарева] Кушнарева Л. Главные отличия PCA от UMAP и t-SNE. 4 мая 2024. https://habr.com/ru/articles/811437/ [Дата обращения – 11.08.2024].

[Лукьянова] Лукьянова К. Байесовский подход к АБ тестированию. Блог компании GlowByte. 11 мая 2023. (Электронный ресурс) https://habr.com/ru/companies/glowbyte/articles/732024/ [Дата обращения – 11.08.2024].

[Маккэндлесс] Маккэндлесс Д. Инфографика. Самые интересные данные в графическом представлении. М.: МИФ, 2014. 264 с. ISBN 978-5-91657-850-8.

[Николаев] Николаев М. Распределения с тяжелыми хвостами: как распознать, где встречаются, почему возникают. 12.10.2021. (Электронный ресурс) https://www.youtube.com/watch?v=U9OuOW4intQ [Дата обращения – 11.08.2024].

[Новицкий] Новицкий П.В., Зограф И.А. Оценка погрешностей результатов измерений. Л., Энергоатомиздат, 1991. 303 с.

[Описательная] Описательная статистика перформанс-распределений/ Блог компании JUG Ru Group. 14 мар 2023. (Электронный ресурс) https://habr.com/ru/companies/jugru/articles/722342/ [Дата обращения – 11.08.2024].

[Орешков] Орешков В.И. Репрезентативность выборочных данных. Блог компании Loginom. 6 сентября 2021. (Электронный ресурс) https://loginom.ru/blog/representativity [Дата обращения – 11.08.2024].

[Оценивание] Оценивание плотности распределения. Информационно-аналитический ресурс MachineLearning.ru. (Электронный ресурс). http://www.machinelearning.ru/wiki/index.php?title=Оценивание_плотности_распределения [Дата обращения – 11.08.2024].

[Пасхавер] Пасхавер И. С. Закон больших чисел и статистические закономерности. М.: Статистика, 1974.

[Петровский] Петровский С. Как мы упростили подготовку данных для тестирования. Блог компании Сбер, 27 мая 2022. (Электронный ресурс) https://habr.com/ru/companies/sberbank/articles/668160/ [Дата обращения – 11.08.2024].

[Поддержка] Поддержка ассоциативного правила (Association Rule Support). Блог компании Loginom. (Электронный ресурс) https://wiki.loginom.ru/articles/association-rule-support.html [Дата обращения – 11.08.2024].

[Пономаренко] Пономаренко А.Н. Статистика. Большая российская энциклопедия. 25.01.2024. (Электронный ресурс) https://bigenc.ru/c/statistika-ce7bad

[Регуляризация] Регуляризация. https://neerc.ifmo.ru/wiki/index.php?title=Регуляризация [Дата обращения – 11.08.2024].

[Справочник] Справочник по теории вероятностей и математической статистике. – 2-е изд., перераб. и дополн. / В.С. Королюк, Н.И. Портенко, А.В. Скороход, А.Ф. Турбин. – М.: Наука, 1985. – 640 с.

[Тырсин] Тырсин А.Н. Метод подбора наилучшего закона распределения непрерывной случайной величины на основе обратного отображения. Вестн. Южно-Ур. ун-та. Сер. Матем. Мех. Физ., 9:1 (2017), 31–38 DOI: 10.14529/mmph170104.

[Тьюки] Тьюки Дж. Анализ результатов наблюдений. Разведочный анализ. Пер. с англ. – М.: Мир, 1981. – 693 с. Tukey J. W. Exploratory data analysis. Reading, MA: Addison-Wesley, 1977. 689 p.

[Фролов] Фролов А. Н.. Краткий курс теории вероятностей и математической статистики: учебное пособие для СПО. — СПб.: Лань, 2021. — С. 189. — 316 с. — ISBN 978-5-8114-8343-3.

[Шеррингтон] Шеррингтон М. Незримые ценности бренда. М.: Вершина, 2006. ISBN: 5-9626-0112-2, 1-4039-0387-5

[Энциклопедия] Энциклопедия статистических терминов. В 8 томах. Том 2. Инструментальные методы статистики. М.: Федеральная служба государственной статистики, 2011. (Электронный ресурс) https://rosstat.gov.ru/free_doc/new_site/rosstat/stbook11/tom2.pdf [Дата обращения – 11.08.2024].

[Яу] Яу Н. Искусство визуализации в бизнесе. Как представить сложную информацию простыми образами. М.: МИФ, 2013. 352 с. ISBN 978-5-91657-737-2.

Александра Сергеевна Ватьян
Наталия Федоровна Гусарова
Наталья Викторовна Добренко

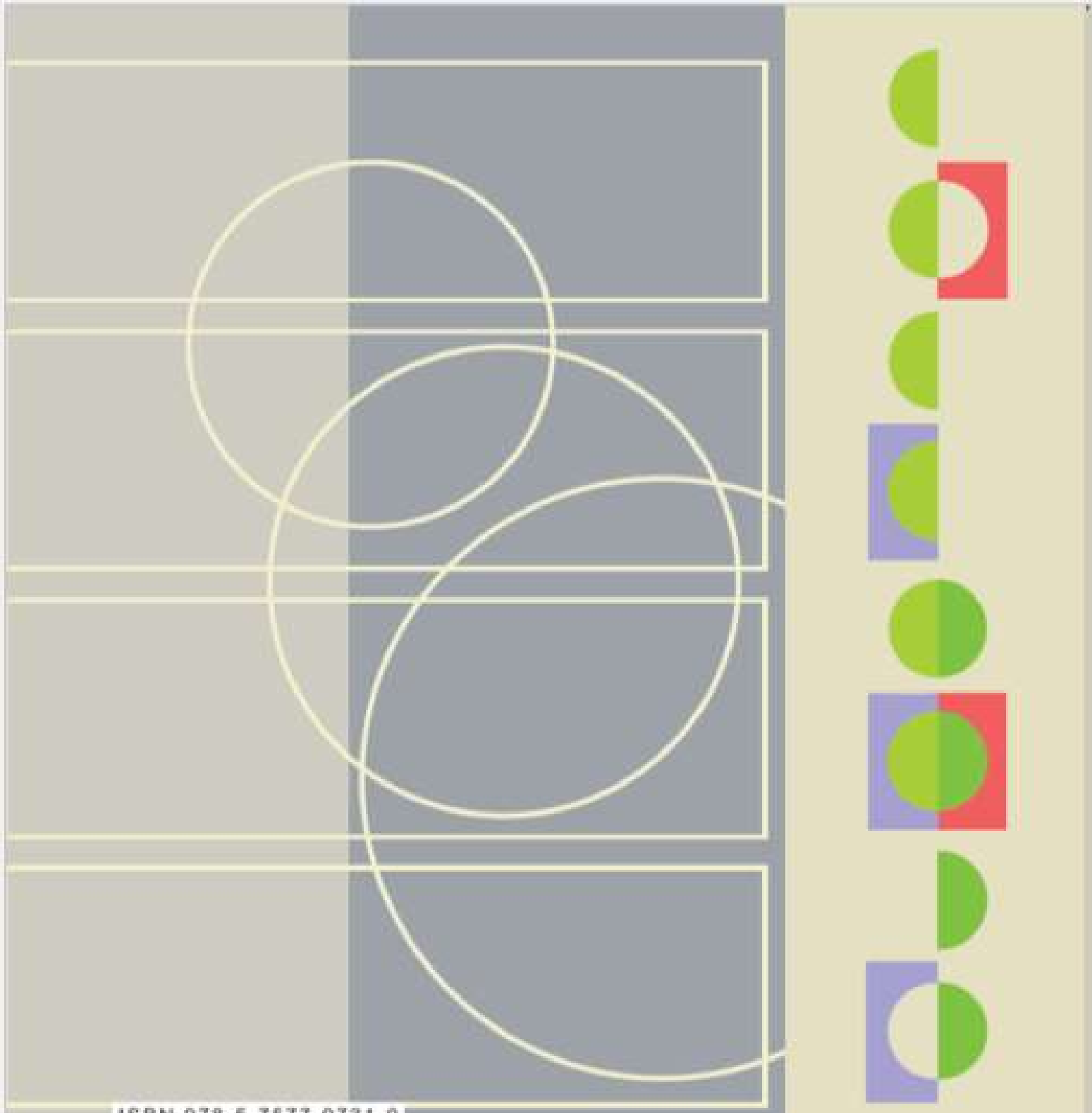
DATA SCIENCE: проблемы и решения

Научно-учебное издание

Редакционно-издательский отдел Университета ИТМО
Зав. РИО Н.Ф. Гусарова
Дизайн обложки Н.А. Потехина
Вёрстка Н.Ф. Гусарова
Подписано к печати 14.01.2025
Заказ № 4792
Тираж 200

Печатается в авторской редакции

Отпечатано: Учреждение «Университетские коммуникации»
199034, Санкт-Петербург, В.О., Биржевая линия, 16



ISBN 978-5-7577-0731-0

Редакционно-издательский отдел
Университета ИТМО
197101, Санкт-Петербург, Кронверкский пр., 49, лит, А