**1. Информация**

**Информация** - сведения независимо от формы их представления.

**Информационная деятельность** — деятельность человека, связанная с процессами:

• Получения информации;

• Преобразования информации;

• Накопления информации;

• Передачи информации.

**Информационная технология** (ИТ) — процесс, использующий совокупность средств и методов сбора, обработки и передачи данных (первичной информации) для получения информации нового качества о состоянии объекта, процесса или явления. Информационная технология характеризуется на каждом этапе инструментарием, видами коммуникаций и целями.

1. "Ручная" информационная технология (до второй половины XIXв.)

2. "Механическая" информационная технология (с конца XIXв.)

3. "Электрическая" информационная технология (40 - 60-е гг. XXв.)

4. "Электронная" информационная технология (с начала 70-х гг. XXв.)

5. "Компьютерная" информационная технология (с середины 80-х гг. ХХ в.)

**Носители информации:**

• внемашинные;

• машиночитаемые;

• электронные.

**Способы обработки информации:**

• Неавтоматизированные;

• Автоматизированные;

• Автоматические.

**Классификация информационных технологий**

• По виду представления обрабатываемой информации;

• В зависимости от класса решаемых задач;

• По степени охвата задач управления.

**Обработка информации** состоит в получении одних «информационных объектов» из других «информационных объектов» путем выполнения некоторых алгоритмов и является одной из основных операций, осуществляемых над информацией, и главным средством увеличения ее объема и разнообразия.

**2. Данные**

**Да́нные —** первичная информация; представление фактов, понятий или инструкций в форме, приемлемой для общения, интерпретации, или обработки человеком или с помощью автоматических средств

**Данные** - необработанный материал, предоставляемый поставщиками данных и используемый потребителями для формирования информации на основе данных

**Данные**:

-Создание данных

-Модификация данных

-Безопасность

-Поиск информации

-Принятие решений

-Создание отчетов

-Создание документов

**Цели**:

1.Сбор информации (различные формы)

2. Определение существенной информации (на данный момент)

3. Предоставление существенной информации (в нужном виде)

**Задачи**

• Сбор данных

➢ Оценка качества данных

• Ввод данных в ИС

➢ Ручной

➢ Автоматический

➢ Контроль и исправление ошибок ввода

• Накопление данных

• Обеспечение доступа к данным

➢ Поиск

➢ Контроль и защита

**3. Знания**

**Знания** — это обработанная информация, записанная и подтвержденная практикой, которая использовалась и может быть повторно использована для принятия некоторых решений.

**Знания** — это тип информации, который хранится в базе знаний и отображает знания специалиста в определенной предметной области.

**Знания** — это интеллектуальный капитал. Знания могут разделяться на **формальные** и **неформальные**. Формальные знания могут храниться в виде документов (стандартов, нормативов), которые регламентируют принятие решений. Также знания могут храниться в виде учебников или инструкций с описанием решения конкретных задач. Неформальные знания — это знания и опыт специалистов в конкретной предметной области.

**4. Понятие интеллектуального анализа данных (ИАД)**

**Интеллектуальный анализ данных** — это обработка информации и выявление в ней моделей и тенденций, которые помогают принимать решения

**Интеллектуальный анализ данных** — “извлечение зерен знаний из гор данных”

**Интеллектуальный анализ данных** представляет собой процесс обнаружения пригодных к использованию сведений в крупных наборах данных. В интеллектуальном анализе данных применяется математический анализ для выявления закономерностей и тенденций, существующих в данных. Обычно такие закономерности нельзя обнаружить при традиционном просмотре данных, поскольку связи слишком сложны, или из-за чрезмерного объема данных.

Эти закономерности и тренды можно собрать вместе и определить, как модель интеллектуального анализа данных. Модели интеллектуального анализа данных могут применяться к конкретным сценариям, а именно:

**Прогнозирование**: оценка продаж, прогнозирование нагрузки сервера или времени простоя сервера

**Риск и вероятность**: выбор наиболее подходящих заказчиков для целевой рассылки, определение точки равновесия для рискованных сценариев, назначение вероятностей диагнозам или другим результатам

**Рекомендации**: определение продуктов, которые с высокой долей вероятности могут быть проданы вместе, создание рекомендаций

**Поиск последовательностей**: анализ выбора заказчиков во время совершения покупок, прогнозирование следующего возможного события

**Группирование**: разделение заказчиков или событий на кластеры связанных элементов, анализ и прогнозирование общих черт



Логические правила типа

«ЕСЛИ …‚ ТО …»

ЕСЛИ Профессия = «Программист», ТО Возраст <= 30 в 61% случаев;

ЕСЛИ Профессия = «Программист», ТО Возраст <= 60 в 98% случаев.

Стадия свободного поиска может выполняться с помощью:

• индукции правил условной логики;

• индукции правил ассоциативной логики;

• определения трендов и колебаний в динамических процессах.

**5. Data Mining**

**Data Mining** — это процесс обнаружения в сырых данных знаний, необходимых для принятия решений в различных сферах человеческой деятельности

**Data Mining** — это методология и процесс обнаружения в больших массивах данных, накапливающихся в информационных системах компаний, ранее неизвестных, нетривиальных, практически полезных и доступных для интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности

Знания, обнаруженные в процессе Data Mining, должны быть нетривиальными и ранее неизвестными. Нетривиальность предполагает, что такие знания не могут быть обнаружены путем простого визуального анализа. Они должны описывать связи между свойствами бизнес-объектов, предсказывать значения одних признаков на основе других и т.д. Найденные знания должны быть применимы и к новым объектам.

Практическая полезность знаний обусловлена возможностью их использования в процессе поддержки принятия управленческих решений и совершенствовании деятельности компании.

Знания должны быть представлены в виде, понятном для пользователей, которые не имеют специальной математической подготовки. Например, проще всего воспринимаются человеком логические конструкции «если, то». Более того, такие правила могут быть использованы в различных СУБД в качестве [SQL](https://wiki.loginom.ru/articles/sql.html)-запросов. В случае, когда извлеченные знания непрозрачны для пользователя, должны существовать методы постобработки, позволяющие привести их к интерпретируемому виду.

Data Mining — это не один, а совокупность большого числа различных методов обнаружения знаний. Все задачи, решаемые методами Data Mining, можно условно разбить на шесть видов:

* [Классификация](https://wiki.loginom.ru/articles/classification.html);
* [Регрессия](https://wiki.loginom.ru/articles/regression.html);
* [Кластеризация](https://wiki.loginom.ru/articles/clustering.html);
* [Ассоциация](https://wiki.loginom.ru/articles/association-rules.html);
* [Последовательные шаблоны](https://wiki.loginom.ru/articles/sequence-pattern.html);
* [Анализ отклонений](https://wiki.loginom.ru/articles/deviation-detection.html).



**Современные требования к интеллектуальной переработке сырых данных**

• Данные имеют неограниченный объем

• Данные являются разнородными (количественными, качественными, текстовыми)

• Результаты должны быть конкретны и понятны

• Инструменты для обработки сырых данных должны быть просты в использовании

**6. Business Intelligence**

**Business Inteligence/BI-систем (по гуглу)** – это набор инструментов и технологий для сбора, анализа и обработки данных. Например, в компании для приёма заявок используют несколько каналов и нужно собрать единую статистику продаж. Или рекламные кампании охватывают несколько площадок и необходимо сравнить их эффективность. Все эти процессы можно настроить через BI-систему. Необработанную информацию из разных источников посредством BI преобразуют в удобную и понятную аналитику. BI-системы (Microsoft Power BI, Tableau, Qlik) можно применять в любой отрасли или сфере деятельности — как на уровне компании в целом, так и для подразделений или отдельных продуктов.

**Buisness Intelligence (по лекции) —** программные средства, функционирующие в рамках предприятия и обеспечивающие функции доступа и анализа информации, которая находится в хранилище данных, а также обеспечивающие принятие правильных и обоснованных управленческих решений.

**Data scientist —** это специалист, владеющий тремя группами навыков:

* IТ-грамотность — программирование, придумывание и решение алгоритмических задач, владение софтом;
* математические и статистические знания;
* содержательный опыт в какой-то области — понимание бизнес-запросов своей организации или задач своей отрасли науки.

В этой профессии нужно уметь строить гипотезы и ставить вопросы. А затем еще и переводить с языка данных на язык бизнеса, выстраивать совместную работу аналитиков, разработчиков и тех, кто занимается собственно развитием бизнеса.

**Основные задачи, решаемые посредством систем BI:**

* Сбор данных из разных источников, их структурирование и хранение в единой системе.
* Анализ больших объёмов данных для формирования и подтверждения гипотез или для разработки бизнес-решений с учётом аналитики.
* Моделирование возможных решений для оценки их влияния на итоговые показатели деятельности и прогнозирование последующего развития на основе имеющихся данных.
* Формирование оперативной и стратегической отчётности, в том числе оповещение об отклонении показателей от допустимых норм.
* Сохранение и систематизация знаний с целью последующей передачи новым сотрудникам, чтобы опыт сохранялся, и качество работы стабильно повышалось.

**7. Пирамида управления и связь с BI**

**Пирамида управления** – это графическая иллюстрация структуры подчиненности и разграничения обязанностей в каждой отдельной компании. В зависимости от специфики бизнеса, задач предприятия и его величины различается построение «вертикалей власти».



**8. Задачи систем поддержки принятия решений**

В состав BI-решения входят:

* Инструменты интеграции и очистки данных (ETL). ETL извлекают информацию из внешних систем-источников, трансформируют её, очищают и загружают в единое хранилище.
* Аналитическое хранилище данных. Это информационная база, которая умеет структурировать и анализировать данные.
* Средства Data Mining. Эти инструменты обрабатывают данные и анализируются по различным срезам. Система выявляет зависимости и тренды. При этом могут использоваться самые разные методы обработки информации — от статистики и прогнозирования до семантического анализа.
* Инструменты визуализации данных. Это отчёты, с которыми работают пользователи. В зависимости от задач отчёты могут строиться по утверждённому формату или быть аналитическими. При построении аналитических отчётов пользователи самостоятельно устанавливают перечень отображаемых показателей, сортируют данные и выстраивают фильтры.

**9. Понятие модели предметной области**

**Модель предметной области** — это система абстракций, которая описывает отдельные аспекты сферы знаний, влияния или деятельности. Затем она может быть использована для решения проблем, связанных с этой сферой. Модель предметной области — это представление значимых концепций реального мира, относящихся к материальным аспектам, которые необходимо моделировать в программном обеспечении. Понятия включают данные, используемые в бизнесе, и правила, которые организация применяет в отношении этих компонентов.

**Модель предметной области** обычно использует профессиональный словарь. Это позволяет передавать представления заинтересованным сторонам. Он не должен ссылаться на какие-либо технические реализации.

Логическая модель предметной области из существующих проектов также может быть использована для улучшения совместимости. Физическая конструкция включает в себя детали, специфичные для системы, такие как типы данных, характерные для языка программирования, ограничения доступа и т. д. Все конкретные реализации будут легко прослежены до эталонного стандарта.

**10. Понятие информационного объекта**

**Информационный объект** – это описание некоторой сущности в виде совокупности логически связанных информационных элементов.

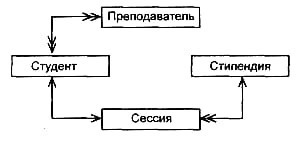
Информационный объект имеет множество реализаций, каждый из которых представлен совокупностью конкретных значений атрибутов и идентифицируется значением ключевого атрибута.

**Сущность** – любой различимый объект (объект, который мы можем отличить от другого), информацию о котором необходимо хранить в базе данных. Сущностями могут быть люди, места, самолеты, рейсы, вкус, цвет и т.д. Необходимо различать такие понятия, как тип сущности и экземпляр сущности. Понятие тип сущности относится к набору однородных личностей, предметов, событий или идей, выступающих как целое. Экземпляр сущности относится к конкретной вещи в наборе. Например, типом сущности может быть ГОРОД, а экземпляром – Москва, Киев и т.д.

**Атрибут** – поименованная характеристика сущности. Его наименование должно быть уникальным для конкретного типа сущности, но может быть одинаковым для различного типа сущностей (например, ЦВЕТ может быть определен для многих сущностей: СОБАКА, АВТОМОБИЛЬ, ДЫМ и т.д.). Атрибуты используются для определения того, какая информация должна быть собрана о сущности. Примерами атрибутов для сущности АВТОМОБИЛЬ являются ТИП, МАРКА, НОМЕРНОЙ ЗНАК, ЦВЕТ и т.д. Здесь также существует различие между типом и экземпляром. Тип атрибута ЦВЕТ имеет много экземпляров или значений: Красный, Синий, Банановый, Белая ночь и т.д., однако, каждому экземпляру сущности присваивается только одно значение атрибута.

**11. Понятие информационно-логической модели**

Информационно-логическая модель отображает данные предметной области в виде совокупности информационных объектов и связей между ними



Информационный объект - это информационное описание некоторого реального объекта, процесса, явления или события. Информационный объект образуется совокупностью взаимосвязанных реквизитов, представляющих качественные и количественные характеристики предметной области. Примерами информационных объектов могут быть Сотрудник, Ученик, Учитель, Методическая комиссия и т. п. Каждому информационному объекту нужно присвоить уникальное имя, соответствующее этому объекту.

Информационный объект может иметь множество реализаций - экземпляров. Например, каждый экземпляр объекта Ученик представляет конкретного ученика. Экземпляр характеризуется совокупностью конкретных значений реквизитов и должен однозначно идентифицироваться значением ключа информационного объекта, который может состоять из одного или нескольких ключевых реквизитов. Таким образом, реквизиты подразделяются на описательные и ключевые.

Реквизиты каждого информационного объекта должны отвечать определенным требованиям:

• информационный объект должен содержать уникальный идентификатор (ключ);

• все описательные реквизиты должны быть взаимонезависимыми;

• все реквизиты, входящие в составной ключ, должны быть также взаимонезависимыми;

• каждый описательный реквизит должен функционально зависеть от ключа, т. е. каждому значению ключа соответствует только одно значение описательного реквизита;

• при составном ключе описательные реквизиты должны зависеть целиком от всей совокупности реквизитов, образующих ключ;

• каждый описательный реквизит не может зависеть от ключа опосредовано, т. е. через другой промежуточный реквизит.

Процесс выделения информационных объектов может производиться на основе интуитивного или формального подхода. При интуитивном подходе легко могут быть выявлены информационные объекты, соответствующие реальным. Однако, получаемая при этом ИЛМ, как правило, требует дальнейших преобразований. При таком подходе возможны существенные ошибки. Последующая проверка выполнения требований обычно приводит к необходимости уточнения информационных объектов.

Следующим шагом проектирования после выявления информационных объектов является определение связей между ними. Связь устанавливается между двумя информационными объектами, если логически взаимосвязаны экземпляры этих информационных объектов. Связи информационных объектов могут быть разного типа:

• одно-однозначные (1:1);

• одно-многозначные (1:М);

• много-многозначные (М:N)

Одно-однозначные связи имеют место, когда каждому экземпляру первого объекта соответствует только один экземпляр второго объекта и наоборот, каждому экземпляру второго объекта соответствует только один экземпляр первого объекта. Следует отметить, что такие объекты могут быть объединены в один, структура которого образуется объединением реквизитов обоих объектов, а ключевым реквизитом может быть выбран любой из ключей исходных объектов.

Одно-многозначные связи - это такие связи, когда каждому экземпляру одного объекта может соответствовать несколько экземпляров другого объекта, а каждому экземпляру второго объекта может соответствовать только один экземпляр первого объекта. В такой связи первый объект является главным, а второй подчиненным.

Много-многозначные связи - это такие связи, когда каждому экземпляру первого объекта соответствуют несколько экземпляров второго объекта и каждому экземпляру второго объекта может соответствовать несколько экземпляров первого объекта. Такие связи не могут непосредственно реализовываться в реляционной базе данных. Их можно реализовать путем введения дополнительного объекта «связка». Объект связка должен иметь идентификатор, образованный из идентификаторов исходных объектов.

Проектирование базы данных заканчивается созданием на основе информационно-логической модели (ИЛМ) логической структуры реляционной базы данных, которая отображается Access как схема данных.

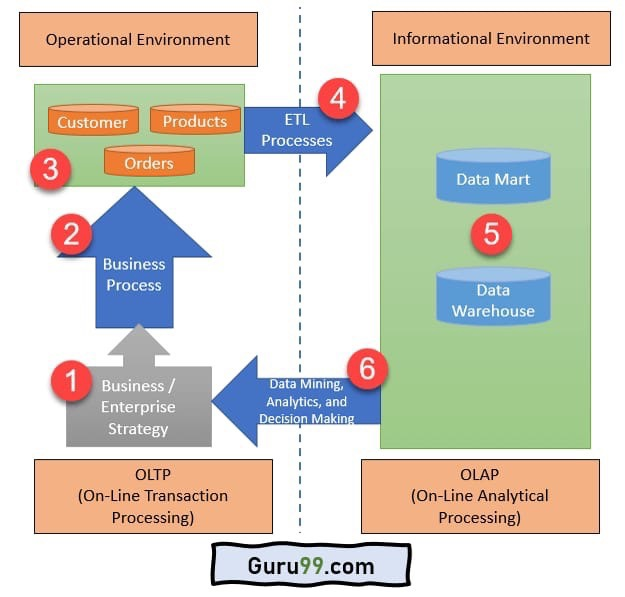
**12. OLTP-системы**

OLTP (online transaction processing - оперативная обработка транзакций) - система обработки транзакций в реальном времени. Способ организации БД, при котором система работает с небольшими по размерам транзакциями, но идущими большим потоком, и при этом клиенту требуется от системы минимальное время отклика.

OLTP использует полностью нормализованную схему для согласованности базы данных

OLTP система — это система смены базы данных в режиме онлайн. Поэтому он поддерживает запрос к базе данных, такой как вставка, обновление и удаление информации из базы данных.

**Архитектура:**



**Стратегия бизнеса / предприятия**. Стратегия предприятия касается вопросов, которые влияют на организацию в целом. В OLTP он обычно разрабатывается на высоком уровне внутри компании советом директоров или высшим руководством.

**Бизнес-процесс**. Бизнес-процесс OLTP представляет собой набор действий и задач, которые после его выполнения будут выполнять организационную задачу.

**Клиенты, заказы и продукты**. В базе данных OLTP хранится информация о продуктах, заказах (транзакциях), клиентах (покупателях), поставщиках (продавцах) и сотрудниках.

**Процессы ETL**: он отделяет данные от различных исходных систем РСУБД, затем преобразует данные (например, применяет конкатенации, вычисления и т. Д.) И загружает обработанные данные в систему хранилища данных.

**Маркетинг данных и хранилище данных:** витрина данных — это структура / схема доступа, характерная для сред хранилищ данных. Он используется OLAP для хранения обработанных данных.

**Сбор данных, аналитика и принятие решений**. Данные, хранящиеся в витрине данных и хранилище данных, могут использоваться для анализа данных, анализа и принятия решений. Эти данные помогают вам обнаруживать шаблоны данных, анализировать необработанные данные и принимать аналитические решения для роста вашей организации.

Примером системы OLTP является Онлайн бронирование авиабилетов

различие между OLTP и OLAP

|  |  |
| --- | --- |
| **OLTP** | **OLAP** |
| OLTP — это транзакционная онлайн-система. | OLAP — это онлайн-анализ и поиск данных. |
| Он характеризуется большим количеством коротких онлайн-транзакций. | Характеризуется большим объемом данных. |
| OLTP — это онлайновая система модификации баз данных. | OLAP — это онлайновая система управления запросами к базам данных. |
| OLTP использует традиционные СУБД. | OLAP использует хранилище данных. |
| Вставка, обновление и удаление информации из базы данных. | В основном выберите операции |
| OLTP и его транзакции являются источниками данных. | Различные базы данных OLTP становятся источником данных для OLAP. |
| База данных OLTP должна поддерживать ограничения целостности данных. | База данных OLAP не часто модифицируется. Следовательно, целостность данных не является проблемой. |
| Это время отклика в миллисекундах. | Время отклика в секундах до минут. |
| Данные в базе данных OLTP всегда детализированы и упорядочены. | Данные в процессе OLAP могут быть не организованы. |
| Разрешить операции чтения / записи. | Только читаю и редко пишу. |
| Это рыночный процесс. | Это ориентированный на клиента процесс. |
| Запросы в этом процессе стандартизированы и просты. | Сложные запросы с участием агрегатов. |
| Полное резервное копирование данных в сочетании с инкрементным резервным копированием. | OLAP время от времени требуется только резервное копирование. Резервное копирование не важно по сравнению с OLTP |
| Разработка БД — это пример, ориентированный на приложения: дизайн базы данных меняется в зависимости от отрасли, такой как розничная торговля, авиакомпания, банковское дело и т. Д. | Дизайн БД является предметно-ориентированным. Пример: изменение дизайна базы данных по таким темам, как продажи, маркетинг, закупки и т. Д. |
| Он предназначен для бизнес-операций в реальном времени. | Он предназначен для анализа бизнес-показателей по категориям и атрибутам. |
| Пропускная способность транзакций является метрикой производительности | Пропускная способность запросов — это показатель производительности. |
| Этот вид пользователя базы данных позволяет тысячи пользователей. | Этот вид базы данных позволяет только сотням пользователей. |
| Это помогает повысить самообслуживание пользователя и производительность | Помогите увеличить производительность бизнес-аналитиков. |
| OLTP имеет быстрое время отклика, низкую избыточность данных и нормализован. | Хранилище данных создается уникальным образом, чтобы в него можно было интегрировать разные источники данных для построения консолидированной базы данных. |

**13.-14. Концепция хранилища данных. Организация ХД**

Хранилище данных - это электронное хранилище большого объема информации из различных источников, предназначенное для анализа и управления данными. Это процесс преобразования данных в информацию и своевременного предоставления их пользователям.

Основная концепция хранилища данных состоит в том, чтобы упростить для компании процесс принятия решений и прогнозирования. Хранилище данных — это информационная система, которая содержит исторические и коммутативные данные из одного или нескольких источников. Концепция хранилища данных, упрощает процесс отчетности и анализа организации.

**Как работает хранилище данных?**

Хранилище данных работает как центральное хранилище, куда информация поступает из одного или нескольких источников данных. Данные поступают в хранилище данных из транзакционной системы и других реляционных баз данных (Данные могут быть: Структурированные, Полуструктурированный, Неструктурированные данные).

Данные обрабатываются и преобразуются, так что пользователи могут получить доступ к обработанным данным в хранилище данных с помощью инструментов бизнес-аналитики, клиентов SQL и электронных таблиц. Хранилище данных объединяет информацию, поступающую из разных источников, в одну комплексную базу данных.

**Преимущества:**

\* Хранилище данных позволяет бизнес-пользователям быстро получать доступ к критически важным данным из некоторых источников в одном месте. Таким образом, это экономит время пользователя на получение данных из нескольких источников.

\* Хранилище данных предоставляет согласованную информацию о различных межфункциональных действиях. Он также поддерживает специальные отчеты и запросы.

\* Хранилище данных помогает интегрировать множество источников данных, чтобы снизить нагрузку на производственную систему.

\* Хранилище данных помогает сократить общее время обработки для анализа и отчетности.

\* Хранилище данных хранит большое количество исторических данных. Это помогает пользователям анализировать различные периоды времени и тенденции, чтобы делать прогнозы на будущее.

**Недостатки:**

\* Не идеальный вариант для неструктурированных данных.

\* Создание и внедрение хранилища данных — это, безусловно, запутанное время.

\* Хранилище данных может сравнительно быстро устареть

\* Трудно вносить изменения в типы данных и диапазоны, схему источника данных, индексы и запросы.

\* Хранилище данных может показаться простым, но на самом деле оно слишком сложное для обычных пользователей.

\* Несмотря на все усилия по управлению проектами, объем проекта хранилищ данных всегда будет увеличиваться.

\* Иногда пользователи склада разрабатывают различные бизнес-правила.

\* Организации должны тратить много своих ресурсов на обучение и внедрение.

**Разница между базой данных и хранилищем данных:**

|  |  |  |
| --- | --- | --- |
| параметр | База данных | Хранилище данных |
| Цель | Предназначен для записи | Предназначен для анализа |
| Метод обработки | База данных использует онлайн-обработку транзакций (OLTP) | Хранилище данных использует онлайн-аналитическую обработку (OLAP). |
| Применение | База данных помогает выполнять фундаментальные операции для вашего бизнеса | Хранилище данных позволяет анализировать ваш бизнес. |
| Таблицы и соединения | Таблицы и объединения базы данных являются сложными, поскольку они нормализованы. | Таблицы и объединения просты в хранилище данных, потому что они денормализованы. |
| ориентация | Является ли прикладной сбор данных | Это предметно-ориентированный сбор данных |
| Предел хранения | Обычно ограничивается одним приложением | Хранит данные из любого количества приложений |
| Доступность | Данные доступны в режиме реального времени | Данные обновляются из исходных систем по мере необходимости |
| Применение | Методы ER моделирования используются для проектирования. | Методы моделирования данных используются для проектирования. |
| Тип данных | Данные, хранящиеся в базе данных, актуальны. | Текущие и исторические данные хранятся в хранилище данных. Может быть не в курсе. |
| Хранение данных | Для хранения данных используется метод плоского реляционного подхода. | Data Ware House использует размерный и нормализованный подход к структуре данных. Пример: схема «звезда» и «снежинка». |
| Тип запроса | Используются простые транзакционные запросы. | Сложные запросы используются для целей анализа. |

Подводя итог, можно сказать, что база данных помогает выполнять основную деятельность бизнеса, а хранилище данных помогает анализировать ваш бизнес. Вы выбираете любой из них в зависимости от ваших бизнес-целей.

**15. Данные и атрибуты**

Данные - это информация, которую человек хранит и по потребности может извлекать и каким-то образом манипулировать ею (редактировать, добавлять новые данные, удалять)

Атрибут - свойство, характеризующее объекты (объект - это объект реального мира. Например, объект - преподаватель, его атрибуты - фио, кафедра, зарплата)

**16.-20. Шкала измерений: номинальная, порядковая, интервальная, относительная, дихотомическая**

Шкала измерения в статистике — это способ представления переменных (признаков, атрибутов) и их группировки в различные категории.

1. Номинальная шкала — это шкала измерения, которая используется для идентификации. Она является самой «слабой» из четырех видов шкал в смысле возможности обработки данных. Она присваивает номера атрибутам для удобства идентификации и может использоваться только как метка. Единственный вид статистического анализа, который можно выполнить с использованием номинальной шкалы, это вычисление процентных долей и частот. Данные в номинальной шкале можно проанализировать графически с помощью гистограммы и круговой диаграммы. Например, если измерить атрибут «Товар» в номинальной шкале, то она будет выглядеть так: 1 — мороженное; 2 — соки; 4 — выпечка. При этом значения шкалы не определяют какого-либо приоритета между товарами, а просто идентифицируют их. Очевидно, что такая шкала может использоваться только для самого просто анализа.

2. Порядковая шкала — предполагает упорядочивание значений переменной в зависимости от масштабирования. Атрибуты в порядковой шкале обычно располагаются в порядке возрастания или убывания. Порядковая шкала может быть использована в исследованиях рынка, рекламы и опросов удовлетворенности клиентов. Она использует квалификаторы, такие как «очень», «высоко», «больше», «меньше» и т. д. В порядковой шкале можно использовать для статистического анализа такие статистики как медиана, но не среднее значение. Существуют и другие виды анализа, которые могут быть проведены с использованием порядковой шкалы. Например, компания-разработчик ПО может провести опрос пользователей для оценки нового приложения в шкале: «Отлично», «Очень хорошо», «Хорошо», «Плохо», «Очень плохо». Атрибуты в этом примере перечислены в порядке убывания.

3. Интервальная шкала (разностей) — это шкала, в которой уровни упорядочены, а интервалы между ними равны. Её можно рассматривать как расширение порядковой шкалы. Основным отличием является свойство равных интервалов. Интервальная шкала не только позволяет однозначно определить, какое значение больше (меньше), но и на сколько. Кроме того, в отличие от порядковой и номинальной шкал, в интервальной могут выполняться арифметические операции. Типичным примером является измерение температуры по шкале Фаренгейта. Интервальную шкалу можно использовать при расчете среднего значения, медианы, моды, стандартного отклонения и других статистик.

5. Шкала отношений (абсолютная) является «наивысшим» уровнем представления данных. Она может рассматриваться как расширение интервальной шкалы, и следовательно, удовлетворяет четырем свойствам шкалы измерения: идентифицируемостью, величиной, равноинтервальностью и наличием абсолютного нуля. Примерами шкал отношения являются длина, вес, время и т. д. В исследованиях рынка примерами шкалы отношений являются цена, количество клиентов, суммы продаж и т. д. Она широко используется в маркетинге и рекламе. Шкала отношений совместима со всеми методами статистического анализа и может использовать как показатели центральной тенденции (среднее значение, медиана, мода и т. д.), так и разброса значения (дисперсии, размаха, стандартного отклонения и т. д.).

6. Дихотомическая шкала (dichotomous scale) - шкала, содержащая только две категории. Пример такой шкалы: пол (мужской и женский)

**21. Табличные данные**

**Табличные данные** – это просто данные, содержащиеся в таблице.

**Таблица** — это совокупность связанных данных, хранящихся в структурированном виде в [базе данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). Она состоит из столбцов и строк.

В [реляционных базах данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) и плоских [файлах баз данных](https://en.wikipedia.org/wiki/Flat_file_database), таблица — это набор элементов данных (значений), использующий модель вертикальных столбцов (имеющих уникальное имя) и горизонтальных строк. Ячейка — место, где строка и столбец пересекаются.Таблица содержит определенное число столбцов, но может иметь любое количество строк. Каждая строка однозначно определяется одним или несколькими уникальными значениями, которые принимают её ячейки из определенного подмножества столбцов. Подмножество столбцов, которое уникально идентифицирует строку, называется [первичным ключом](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B2%D0%B8%D1%87%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BB%D1%8E%D1%87).

«Таблица» — это ещё один термин для «[отношения](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BD%D0%BE%D1%88%D0%B5%D0%BD%D0%B8%D0%B5_(%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C))»; разница между ними в том, что таблица обычно представляет собой [мультимножество](https://ru.wikipedia.org/wiki/%D0%9C%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BC%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2%D0%BE) (набор) строк, тогда как отношение представляет собой [множество](https://ru.wikipedia.org/wiki/%D0%9C%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2%D0%BE_(%D1%82%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)) и не допускает дубликатов. Помимо обычных данных, таблицы, как правило, имеют связанные с ними [метаданные](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%B0%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5), такие как [ограничения](https://en.wikipedia.org/wiki/Check_constraint), относящиеся к таблицам в целом или к значениям в определенных столбцах.

Данные в таблицах не обязательно физически хранятся в базе данных. [Представления](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D0%B4%D1%81%D1%82%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_(%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)) также функционируют, как реляционные таблицы, но их данные вычисляются во время выполнения запроса. Внешние таблицы (например, в СУБД [Informix](https://ru.wikipedia.org/wiki/Informix) или [Oracle](https://ru.wikipedia.org/wiki/Oracle_Database),) также можно рассматривать как представления.

**Табличная** информация является основой географических объектов, позволяющей визуализировать, строить запросы и анализировать ваши **данные**. Проще говоря, таблицы состоят из строк и столбцов, все строки имеют одинаковые столбцы. ... Каждое поле может содержать **данные** одного определенного типа, такие как числа, даты или текст.

**22. Транзакционные данные**

**Транзакционные данные** – это данные, которые образовались в результаты выполнения предприятием каких-либо бизнес-транзакций. Например, для коммерческого предприятия: продажи продуктов и услуг, закупки, поступления/списания денежных средств, поступления на склад и т.п. Обычно такие данные базируются в системе управления ресурсами предприятия (ERP) или других отраслевых системах. Естественно, транзакционные системы широко используют мастер-данные при выполнении транзакций.

Транзакционные данные не требуют особых дополнительных пояснений. Они представляют собой более тактический ресурс и сфокусированы на достижении каких-то целей. Каждая транзакция генерирует транзакционные данные, используя определенную часть мастер-данных и операционных данных вместе с информацией, описывающей конкретную транзакцию (время, место, количество, сумма и т.д.). Когда вы готовите отчеты – результатом часто является перечень или обобщение данных транзакционного уровня (например, отчеты о продажах основаны на транзакциях, объединенных по некоторому параметру данных). В некоторых подразделениях, таких как Закупки, транзакционные данные могут стать триггером, инициирующим изменения в операционных данных. Хорошим примером может быть, например, обновление средней стоимости закупаемого вида товара/услуги (которая является частью операционных данных) по мере осуществления новых закупок, т.е. на основании транзакционных данных из заказов на закупки.

**23. Метаданные**

**Метаданные** - это данные о данных, информация об информации, описание контента. Хранение и доставка информации в электронном виде порождает много проблем. Пользователи должны иметь возможность найти нужную информацию, получить доступ к ней в приемлемой для них форме. Создатели информации должны быть уверены, что их права на интеллектуальную собственность будут защищены, а администраторы и иные специалисты должны иметь возможности по сопровождению электронной информации, например, обеспечение ее сохранности в течение длительного времени. Метаданные являются ключевым компонентом для решения этих проблем. Учитывая, что значительная часть служебных задач может решаться и реально решается без участия человека, метаданных подразделяют на предназначенные для использования приложениями и для использования человеком.

Метаданные – это средство классификации, упорядочивания и характеристики данных или *содержимого*. Национальная организация по информационным стандартам (NISO) предлагает [классификацию](https://www.niso.org/publications/press/UnderstandingMetadata.pdf), которую можно применить для всех типов данных или репозиториев данных, от библиотек до веб-сайтов, для текстовых и нетекстовых данных, в цифровой или материальной форме.

**NISO описывает три типа метаданных.**

*Описательные метаданные* включают такую информацию, как точки контакта, заголовок или автор публикации, аннотация работы, используемые в работе ключевые слова, географическое местоположение или даже пояснение методологии. Эти данные служат для обнаружения, сбора или группирования ресурсов по общим для них характеристикам.

*Структурные метаданные* поясняют состав или организацию ресурсов. Например, цифровую книгу можно публиковать в виде изображений отдельных страниц, файла PDF или HTML. Эти страницы или *компоненты* обычно группируют в главы. Данные о главах, содержание или сведения о макете страниц считаются [структурными метаданными](https://www.digitizationguidelines.gov/term.php?term=metadatastructural). К структурным метаданным относятся также [такие записи](https://www.cisco.com/c/en/us/td/docs/security/firesight/540/api/estreamer/EventStreamerIntegrationGuide/IS-DCRecords.html), как структурная карта страниц или иных ресурсов веб-сайта, событие вторжения или [записи сведений о голосовых вызовах](https://en.wikipedia.org/wiki/Call_detail_record).

*Административные метаданные* используются для управления ресурсом. Даты создания или получения, права доступа, права или происхождение, либо правила утилизации, такие как хранение или удаление, являются примерами прав, которые может применять [цифровой архивист](https://journals.ala.org/lrts/article/view/5031/6088), куратор. Сходные метаданные окажутся полезными для администратора базы данных или для администраторов, отвечающих за получение данных из [трафика](https://qosient.com/argus/presentations/Argus.FloCon.2014.Metadata.Tutorial.pdf) телекоммуникационных сетей или сетей передачи данных, либо журналов систем безопасности или данных о событиях.

# **24. Многомерная модель данных**

В многомерной модели данные представляются в виде гиперкубов, используя которые, можно получать различные срезы при аналитической обработке данных. Оси гиперкуба содержат параметры, а ячейки — зависящие от них агрегатные данные. Вдоль каждой оси данные могут быть организованы в виде иерархии, представляющей различные уровни их детализации. Благодаря такой модели данных пользователи могут формулировать сложные запросы, генерировать отчеты, получать подмножества данных.

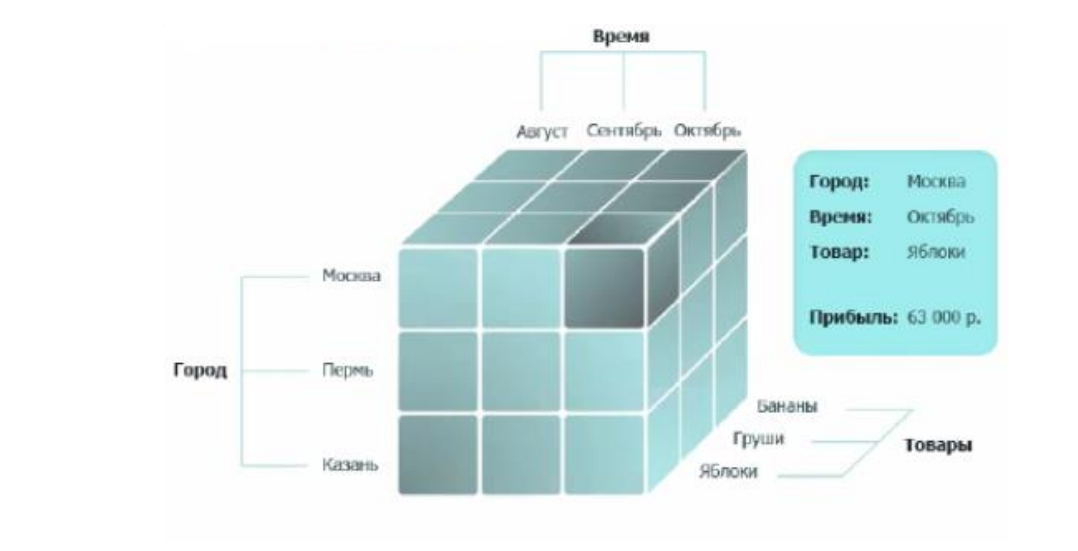
Многомерная модель называется полукубической, если предметная область может быть отображена в виде нескольких гиперкубов с различной размерностью и разными измерениями в качестве граней.

Если же модель может состоять из одного гиперкуба или нескольких гиперкубов с одинаковой размерностью и совпадающими измерениями, то она называется гиперкубической.

# **25. OLAP-системы**

OLAP (online analytical processing, интерактивная аналитическая обработка) — это технология комплексного многомерного анализа данных.

**Кубом OLAP** называют структуру, в которой хранятся совокупности данных, полученные путем всех возможных сочетаний измерений в таблице фактов



OLAP системы включают ключевые компоненты:

* базу данных (БД) - источник, из которого берется информационный материал для обработки. Тип БД определяется разновидностью OLAP системы и порядком выполнения действий OLAP сервера. Чаще всего пользуются реляционными и многомерными БД и хранилищами данных;
* OLAP сервер - ядро системы, с помощью которого проводится обработка многомерных структур данных, и обеспечивается связь между БД и пользователями систем;
* приложения для работы пользователей, в которых формируются запросы и визуализируются полученные ответы.

Специфика обработки данных OLAP системами состоит в построении многомерных, то есть имеющих большое количество связей между отдельными элементами, массивов информации. Для формирования таких массивов OLAP система собирает данные из различных источников (например, из хранилищ данных, из информационных систем управления предприятием (ERP) или из системы взаимодействия с клиентами (CRM)). После этого информация обрабатывается на OLAP сервере и передается в пользовательские приложения.

Хранение и обработка данных с применением OLAP систем могут осуществляться:

* непосредственно на рабочих местах пользователей;
* в форме реляционных баз данных - при совместной работе OLAP систем с ERP или CRM-системами;
* в форме многомерных баз данных на обособленных серверах.

OLAP системы применяются главным образом в сфере принятия стратегических управленческих решений и используются для бюджетирования, построения прогностических моделей, подготовки финансовой отчетности, хранения результатов.

Характеристики OLAP систем делятся на основные и специальные. Основные представлены:

* многомерностью моделей данных, то есть их многовариантностью и большим количеством проекций;
* интуитивностью механизмов работы с информацией - все манипуляции проводятся без задействования сложных меню;
* клиент-серверной архитектурной ориентированностью;
* доступностью данных - с помощью таких систем выстраивается надежная взаимосвязь между разнородными источниками информации и рабочим интерфейсом;
* пакетным извлечением данных, которое позволяет обеспечить не только хранение анализируемых данных, но и динамический доступ к их источникам;
* прозрачностью - определяется возможностью полного доступа к инструментам OLAP системы и к разнородным источникам данных;
* многопользовательской работой, которая гарантирует одновременный доступ к информации с ее извлечением, обновлением при условиях сохранения ее в безопасности и целостности.

Специальные характеристики OLAP систем включают:

* обработку ненормализованной (избыточно введенной в БД) информации;
* хранение OLAP результатов с разграничением от исходных данных;
* выделение отсутствующих данных (то есть данных, которые не определяются или не имеют смысла), отличающихся по значению от нуля;
* обработку отсутствующих значений, которые (независимо от их источника) игнорируются в ходе аналитики.

1. Многомерность: OLAP-система на концептуальном уровне должна представлять данные в виде многомерной модели, что упрощает процессы анализа и восприятия информации.
2. Прозрачность: OLAP-система должна скрывать от пользователя реальную реализацию многомерной модели, способ организации, источники, средства обработки и хранения.
3. Доступность - OLAP-система должна предоставлять пользователю единую, согласованную и целостную модель данных, обеспечивая доступ к данным независимо от того, как и где они хранятся.
4. Постоянная производительность при разработке отчетов — производительность OLAP-систем не должна значительно уменьшаться при увеличении количества измерений, по которым выполняется анализ.
5. Клиент-серверная архитектура - OLAP-система должна быть способна работать в среде "клиент-сервер", т. к. большинство данных, которые сегодня требуется подвергать оперативной аналитической обработке, xpaнятся распределенно. Главной идеей здесь является то, что серверный компонент инструмента OLAP должен быть достаточно интеллектуальным и позволять строить общую концептуальную схему на основе обобщения и консолидации различных логических и физических схем корпоративных БД для обеспечения эффекта прозрачности.
6. Равноправие измерений - OLAP-система должна поддерживать многомерную модель, в которой все измерения равноправны. При необходимости дополнительные характеристики могут быть предоставлены отдельным измерениям, но такая возможность должна быть предоставлена любому измерению.
7. Динамическое управление разреженными матрицами - OLAP-система должна обеспечивать оптимальную обработку разреженных матриц. Скорость доступа должна сохраняться вне зависимости от расположения ячеек данных и быть постоянной величиной для моделей, имеющих разное число измерений и различную степень разреженности данных.
8. Поддержка многопользовательского режима - OLAP-система должна предоставлять возможность работать нескольким пользователям совместно с одной аналитической моделью или создавать для них различные модели из единых данных. При этом возможны как чтение, так и запись данных, поэтому система должна обеспечивать их целостность и безопасность.
9. Неограниченные перекрестные операции - OLAP-система должна обеспечивать сохранение функциональных отношений, описанных с помощью определенного формального языка между ячейками гиперкуба при выполнении любых операций среза, вращения, консолидации или детализации. Система должна самостоятельно (автоматически) выполнять преобразование установленных отношений, не требуя от пользователя их переопределения.
10. Интуитивная манипуляция данными - OLAP-система должна предоставлять способ выполнения операций среза, вращения, консолидации и детализации над гиперкубом без необходимости пользователю совершать множество действий с интерфейсом. Измерения, определенные в аналитической модели, должны содержать всю необходимую информацию для выполнения вышеуказанных операций.
11. Гибкие возможности получения отчетов - OLAP-система должна поддерживать различные способы визуализации данных, т. е. отчеты должны представляться в любой возможной ориентации. Средства формирования отчетов должны представлять синтезируемые данные или информацию, следующую из модели данных в ее любой возможной ориентации. Это означает, что строки, столбцы или страницы должны показывать одновременно от 0 до N измерений, где N - число измерений всей аналитической модели. Кроме того, каждое измерение содержимого, показанное в одной записи, колонке или странице, должно позволять показывать любое подмножество элементов (значений), содержащихся в измерении, в любом порядке.
12. Неограниченная размерность и число уровней агрегации - исследование о возможном числе необходимых измерений, требующихся в аналитической модели, показало, что одновременно может использоваться до 19 измерений. Отсюда вытекает настоятельная рекомендация, чтобы аналитический инструмент мог одновременно предоставить хотя бы 15, а предпочтительно - 20 измерений. Более того, каждое из общих измерений не должно быть ограничено по числу определяемых пользователем-аналитиком уровней агрегации и путей консолидации.

# **26. Концептуальное многомерное представление данных**

Концептуальное многомерное представление данных представляет собой множественную перспективу, состоящую из нескольких независимых измерений, вдоль которых могут быть проанализированы определенные совокупности данных. Одновременный анализ по нескольким измерениям определяется как многомерный анализ. Осями многомерной системы координат служат основные атрибуты анализируемого процесса.

Например, для продаж это могут быть тип товара, регион, тип покупателя. В качестве одного из измерений используется время. На пересечениях осей - измерений - находятся данные, количественно характеризующие процесс - меры: суммы и иные агрегатные функции. Каждое измерение включает направления консолидации данных, состоящие из серии последовательных уровней обобщения (уровней иерархии), где каждый вышестоящий уровень соответствует большей степени агрегации данных по соответствующему измерению (различные уровни их детализации). В этом случае становится возможным произвольный выбор желаемого уровня детализации информации по каждому из измерений.

Описываемая система наиболее наглядна и удобна в обращении, так как позволяет отражать и рассматривать любые взаимосвязи в самых сложных многокомпонентных системах. Детальные и агрегированные данные содержатся в многомерной базе. Хранение данных в многомерных структурах позволяет манипулировать данными как многомерным массивом, благодаря чему скорость вычисления агрегатных значений одинакова для любого из измерений. MOLAP предполагает создание явного, физически хранимого многомерного куба (или нескольких кубов) с выполнением аналитических запросов только над ними, без обращения к реляционной БД. В этом случае достигается наибольшая производительность, однако в этом случае многомерная база данных оказывается избыточной, так как многомерные данные полностью содержат детальные реляционные данные.

# **27. Практическое применение ИАД**

Существует целый ряд областей, в которых применяется ИАД:

* Торговля. Анализ потребительской корзины, исследование временных шаблонов, создание прогнозирующих моделей, оптимизация складских запасов.
* Банковское дело. Сегментация клиентов, выявление мошенничества с кредитными картами, прогнозирование изменения клиентуры, анализ финансовых рисков.
* Страховой бизнес. Сегментация клиентов, выявление фактов мошенничества, анализ страховых рисков, разработка новых продуктов, расчет страховых премий.
* Телекоммуникации. Анализ лояльности клиентов, сегментирование клиентской базы и услуг, анализ внешних факторов на отказы оборудования, выявление случаев несанкционированного доступа к сети.
* Производственные предприятия. Оптимизация закупок, диагностика брака на ранних стадиях, диагностика оборудования, маркетинг.

# **28. Основные этапы интеллектуального анализа и обработки данных**

В общем случае процесс интеллектуального анализа и обработки данных состоит из следующих шести этапов:

* отбор данных;
* очистка;
* обогащение;
* кодирование;
* извлечение знаний;
* сообщение.

# **29. Отбор данных**

Как правило, для решения конкретной задачи нужны не все данные из хранилища данных. Сначала необходимо выбрать то их подмножество, которое будет подвергнуто анализу. При этом возможно, потребуется объединить несколько таблиц, а полученные записи отфильтровать.

Отбор данных — это выбор наблюдений по определенным критериям. Так, например, при опросе избирателей можно отобрать только мужчин, голосующих за определенную партию, а при опросе студентов – только студенток, изучающих психологию и медицину. После этого все вычисления будут проводиться только с этими отобранными наблюдениями.

# **30. Очистка данных**

Очистка данных занимается выявлением и удалением ошибок и несоответствий в данных с целью улучшения качества данных.

Существуют несколько типов очистки данных (удаление дублирующих записей, исправление типографских ошибок, добавление отсутствующей информации и т.д.), некоторые из которых могут выполняться заранее, в то время как другие вызываются только после обнаружения загрязнения на этапах кодирования или обнаружения.

# **31. Обогащение данных**

Обогащение данных — процесс насыщения данных новой информацией, которая позволяет сделать их более ценными и значимыми с точки зрения решения той или иной аналитической задачи. Можно выделить два основных метода обогащения данных — внешнее обогащение и внутреннее.

Внешнее обогащение предполагает привлечение дополнительной информации из внешних источников, что позволит повысить ценность и значимость данных с точки зрения их анализа.

Внутреннее обогащение не предполагает привлечения внешней информации. Внутреннее обогащение обычно связано с получением и включением в набор данных полезной информации, которая отсутствует в явном виде, но может быть тем или иным способом получена с помощью манипуляций с имеющимися данными.

# **32. Кодирование данных**

На этом этапе мы выделяем только те записи, которые имеют достаточно информации, чтобы быть ценными.

Хотя трудно дать детализированные правила для этого вида операции, это ситуация, которая часто происходит на практике. В большинстве таблиц, которые собраны из операционных данных, отсутствует множество желательных данных и большинство из них невозможно восстановить. Поэтому необходимо принять обдуманное решение или пропустить эту информацию, или удалить её. Общее правило говорит, что любое удаление данных должно быть сознательным решением после всестороннего анализа возможных последствий.

В некоторых случаях, особенно в задачах определения мошенничества, недостаток информации может быть ценным указанием наличия значимых образцов. До сих пор фаза кодирования состояла из простых операций SQL, но теперь для данного этапа требуется творческое преобразование данных.

# **33. Извлечение знаний**

Этап извлечения знаний является ядром процесса интеллектуального анализа и обработки знаний. Технология обнаружения знаний включает много методов и основана на идее, что существует больше знаний, скрытых в данных, чем видно на поверхности.

В настоящее время специалисты выделяют следующие основные методы извлечения знаний: инструментальные средства запроса, статистическая техника, визуализация, интерактивная аналитическая обработка (OLAP), обучение, основанное на прецедентах (k-ближайший сосед), деревья решений, ассоциативные правила, нейронные сети, генетические алгоритмы.

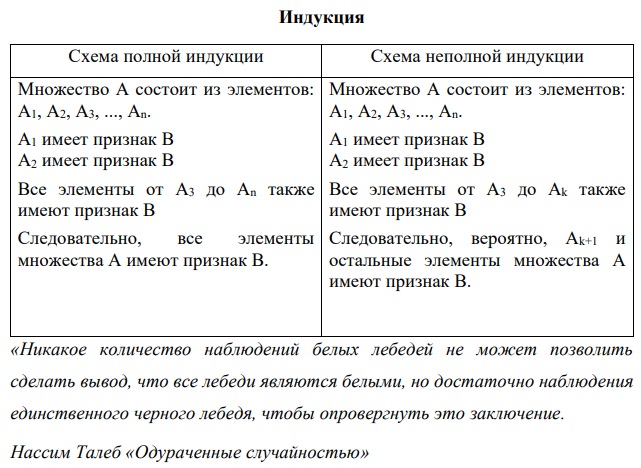
Фактически, в технологии обнаружения знаний необходимо различать четыре различных типа знания, которые могут быть извлечены из данных:

1. Поверхностное знание. Это информация, которая может быть легко найдена из баз данных, использующих инструментальное средство запроса типа структурированного языка запросов (SQL).
2. Многомерное знание. Это информация, которая может быть проанализирована, используя интерактивные аналитические инструментальные средства обработки OLAP. С помощью инструментальных средств OLAP можно быстро исследовать все виды кластеризации и различные упорядочения данных, но важно понимать, что большинство операций, которые можно делать с инструментом OLAP, могут также быть выполнены, используя SQL. Преимущество инструментальных средств OLAP состоит в том, что они оптимизированы для этого вида операций поиска и анализа. Однако, процедуры OLAP не так мощны, как процедуры обнаружения знаний, ибо они не могут искать оптимальные решения.
3. Скрытое знание. Это информация, которая может быть найдена относительно легко, используя алгоритмы распознавания образцов или машинного обучения. Для нахождения этих образцов также можно было бы использовать средства SQL, но это потребовало бы невероятно много времени. Алгоритм распознавания образцов может найти регулярности в базе данных за минуты или, в крайнем случае, всего за несколько часов, и в то же время чтобы достигнуть близкий результат, используя SQL средства, необходимо затратить месяцы.
4. Глубокое знание. Это информация, которая хранится в базе данных, но может быть обнаружена только, если имеется ключ, который сообщит нам, где смотреть. Различие между глубоким и скрытым знанием лучше всего можно объяснить в терминах пространства поиска. Скрытое знание - результат поиска в пространстве с пологим холмистым ландшафтом; алгоритм поиска может легко найти приемлемое оптимальное решение. Глубокое знание - это обычно результат поиска в пространстве, где существует только локальный оптимум, и отсутствуют какие-либо указания о любых возвышенностях по соседству. Алгоритм поиска может передвигаться вокруг этого ландшафта сколь угодно долго, не достигая хоть какого-либо значительного результата. Примером этого может служить зашифрованная информация, хранимая в базе данных. Почти невозможно декодировать сообщение, которое зашифровано, если вы не имеете ключа, который указывает что искать.

**34. Индукция. Дедукция**

Дедукция – переход от общего к частному; в более специальном смысле термин «дедукция» обозначает процесс логического вывода, т. е. перехода по тем или иным правилам логики от некоторых данных предложений-посылок к их следствиям (заключениям).

Индукция - это умозаключение, в результате которого на основе знания об отдельных предметах какого-либо класса делается вывод обо всем классе этих предметов.

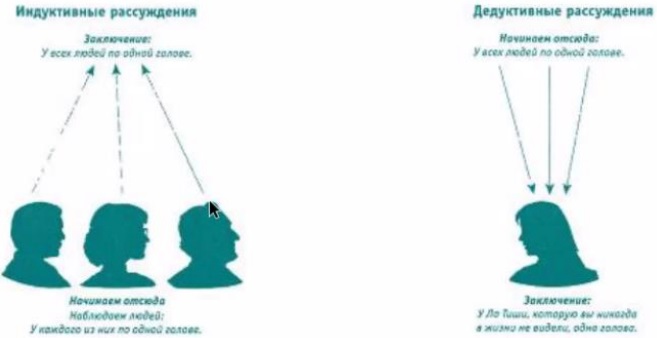
****

**35. Стадии ИАД**

• Свободный поиск — индуктивен.

• Прогнозирование — дедуктивно.

**Прогнозирование** (Forecasting) — это разработка прогноза; в узком значении — специальное научное исследование конкретных перспектив дальнейшего развития какого-либо процесса.



**36. Методы ИАД**

• Исходные данные могут храниться в явном детализированном виде и непосредственно использоваться для прогностического моделирования и/или анализа исключений.

• Информация вначале извлекается из первичных данных и преобразуется в некоторые формальные конструкции.



• Обучающее множество (training set) — множество, которое включает данные, использующиеся для обучения (конструирования) модели.

• Тестовое (test set) множество используется для проверки работоспособности модели.

Несколько основных методов, которые используются для интеллектуального анализа данных, описывают тип анализа и операцию по восстановлению данных. К сожалению, разные компании и решения не всегда используют одни и те же термины, что может усугубить путаницу и кажущуюся сложность.

Рассмотрим некоторые ключевые методы и примеры того, как использовать те или иные инструменты для интеллектуального анализа данных.

### **Ассоциация**

Ассоциация (или отношение), вероятно, наиболее известный, знакомый и простой метод интеллектуального анализа данных. Для выявления моделей делается простое сопоставление двух или более элементов, часто одного и того же типа. Например, отслеживая привычки покупки, можно заметить, что вместе с клубникой обычно покупают сливки.

### **Классификация**

Классификацию можно использовать для получения представления о типе покупателей, товаров или объектов, описывая несколько атрибутов для идентификации определенного класса. Например, автомобили легко классифицировать по типу (седан, внедорожник, кабриолет), определив различные атрибуты (количество мест, форма кузова, ведущие колеса). Изучая новый автомобиль, можно отнести его к определенному классу, сравнивая атрибуты с известным определением. Те же принципы можно применить и к покупателям, например, классифицируя их по возрасту и социальной группе.

Кроме того, классификацию можно использовать в качестве входных данных для других методов. Например, для определения классификации можно применять деревья принятия решений. Кластеризация позволяет использовать общие атрибуты различных классификаций в целях выявления кластеров.

### **Кластеризация**

Исследуя один или более атрибутов или классов, можно сгруппировать отдельные элементы данных вместе, получая структурированное заключение. На простом уровне при кластеризации используется один или несколько атрибутов в качестве основы для определения кластера сходных результатов. Кластеризация полезна при определении различной информации, потому что она коррелируется с другими примерами, так что можно увидеть, где подобия и диапазоны согласуются между собой.

Метод кластеризации работает в обе стороны. Можно предположить, что в определенной точке имеется кластер, а затем использовать свои критерии идентификации, чтобы проверить это. Метод кластеризации можно применить и в обратную сторону: учитывая определенные входные атрибуты, выявлять различные артефакты. Например, недавнее исследование четырехзначных PIN-кодов выявили кластеры чисел в диапазонах 1-12 и 1-31 для первой и второй пар. Изобразив эти пары на графике, можно увидеть кластеры, связанные с датами (дни рождения, юбилеи).

### **Прогнозирование**

Прогнозирование ― это широкая тема, которая простирается от предсказания отказов компонентов оборудования до выявления мошенничества и даже прогнозирования прибыли компании. В сочетании с другими методами интеллектуального анализа данных прогнозирование предполагает анализ тенденций, классификацию, сопоставление с моделью и отношения. Анализируя прошлые события или экземпляры, можно предсказывать будущее.

Например, используя данные по авторизации кредитных карт, можно объединить анализ дерева решений прошлых транзакций человека с классификацией и сопоставлением с историческими моделями в целях выявления мошеннических транзакций. Если покупка авиабилетов в США совпадает с транзакциями в США, то вполне вероятно, что эти транзакции подлинны.

### **Последовательные модели**

Последовательные модели, которые часто используются для анализа долгосрочных данных, ― полезный метод выявления тенденций, или регулярных повторений подобных событий. Например, по данным о покупателях можно определить, что в разное время года они покупают определенные наборы продуктов. По этой информации приложение прогнозирования покупательской корзины, основываясь на частоте и истории покупок, может автоматически предположить, что в корзину будут добавлены те или иные продукты.

### **Деревья решений**

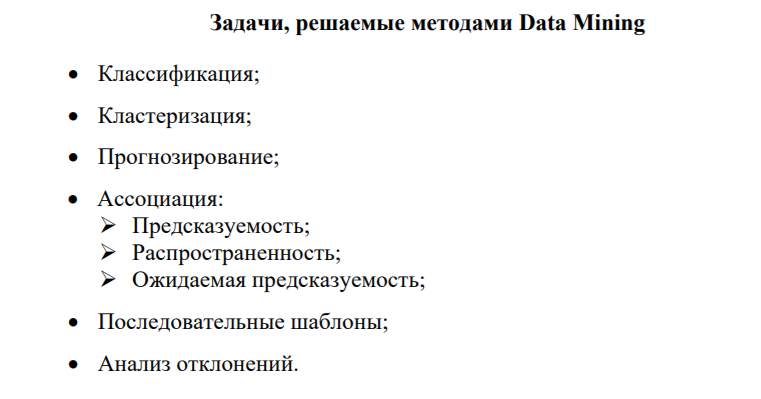
Дерево решений, связанное с большинством других методов (главным образом, классификации и прогнозирования), можно использовать либо в рамках критериев отбора, либо для поддержки выбора определенных данных в рамках общей структуры. Дерево решений начинают с простого вопроса, который имеет два ответа (иногда больше). Каждый ответ приводит к следующему вопросу, помогая классифицировать и идентифицировать данные или делать прогнозы.

##### 

### **Комбинации**

На практике очень редко используется только один из этих методов. Классификация и кластеризация ― подобные методы. Используя кластеризацию для определения ближайших соседей, можно дополнительно уточнить классификацию. Деревья решений часто используются для построения и выявления классификаций, которые можно прослеживать на исторических периодах для определения последовательностей и моделей.

**37. Классификация задач ИАД (Data Mining)**

****

**Классификация (Classification)**

Краткое описание. Наиболее простая и распространенная задача Data Mining. В результате решения задачи классификации обнаруживаются признаки, которые характеризуют группы объектов исследуемого набора данных - классы; по этим признакам новый объект можно отнести к тому или иному классу.

Методы решения. Для решения задачи классификации могут использоваться методы: ближайшего соседа (Nearest Neighbor); k-ближайшего соседа (k-Nearest Neighbor); байесовские сети (Bayesian Networks); индукция деревьев решений; нейронные сети (neural networks).

**Кластеризация (Clustering)**

Краткое описание. Кластеризация является логическим продолжением идеи классификации. Это задача более сложная, особенность кластеризации заключается в том, что классы объектов изначально не предопределены. Результатом кластеризации является разбиение объектов на группы.

Пример метода решения задачи кластеризации: обучение "без учителя" особого вида нейронных сетей - самоорганизующихся карт Кохонена.

**Ассоциация (Associations)**

Краткое описание. В ходе решения задачи поиска ассоциативных правил отыскиваются закономерности между связанными событиями в наборе данных.

Отличие ассоциации от двух предыдущих задач Data Mining: поиск закономерностей осуществляется не на основе свойств анализируемого объекта, а между несколькими событиями, которые происходят одновременно.

Наиболее известный алгоритм решения задачи поиска ассоциативных правил - алгоритм Apriori.

**Последовательность (Sequence)**, или последовательная ассоциация (sequential association)

Краткое описание. Последовательность позволяет найти временные закономерности между транзакциями. Задача последовательности подобна ассоциации, но ее целью является установление закономерностей не между одновременно наступающими событиями, а между событиями, связанными во времени (т.е. происходящими с некоторым определенным интервалом во времени). Другими словами, последовательность определяется высокой вероятностью цепочки связанных во времени событий. Фактически, ассоциация является частным случаем последовательности с временным шагом, равным нулю. Эту задачу Data Mining также называют задачей нахождения последовательных шаблонов (sequential pattern).

Правило последовательности: после события X через определенное время произойдет событие Y.

Пример. После покупки квартиры жильцы в 60% случаев в течение двух недель приобретают холодильник, а в течение двух месяцев в 50% случаев приобретается телевизор. Решение данной задачи широко применяется в маркетинге и менеджменте, например, при управлении циклом работы с клиентом (Customer Lifecycle Management).

**Прогнозирование (Forecasting)**

Краткое описание. В результате решения задачи прогнозирования на основе особенностей исторических данных оцениваются пропущенные или же будущие значения целевых численных показателей.

Для решения таких задач широко применяются методы математической статистики, нейронные сети и др.

**Определение отклонений или выбросов** **(Deviation Detection)**, анализ отклонений или выбросов

Краткое описание. Цель решения данной задачи - обнаружение и анализ данных, наиболее отличающихся от общего множества данных, выявление так называемых нехарактерных шаблонов.

**Оценивание (Estimation)**

Задача оценивания сводится к предсказанию непрерывных значений признака.

**Анализ связей** (Link Analysis) - задача нахождения зависимостей в наборе данных.

**Визуализация (Visualization, Graph Mining)**

В результате визуализации создается графический образ анализируемых данных. Для решения задачи визуализации используются графические методы, показывающие наличие закономерностей в данных.

Пример методов визуализации - представление данных в 2D и 3D измерениях.

Подведение итогов (Summarization) - задача, цель которой - описание конкретных групп объектов из анализируемого набора данных

**38. Стратегии Data Mining**

Согласно классификации по стратегиям, задачи Data Mining подразделяются на следующие группы:

• обучение с учителем;

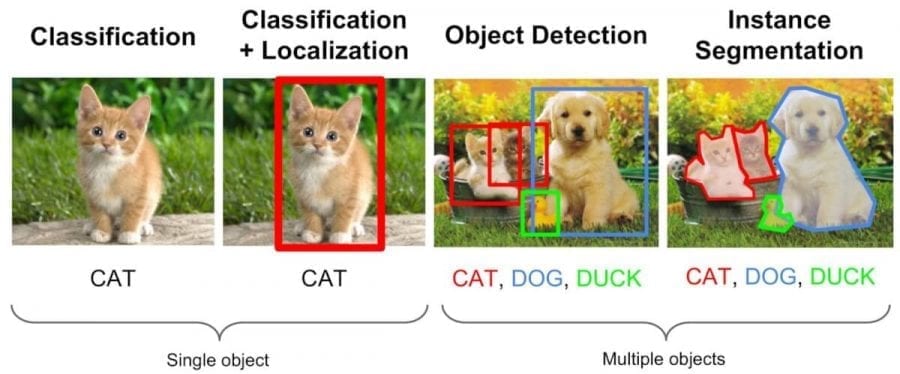
• обучение без учителя;

• другие.

**39. Обучение с учителем**

Категория обучение с учителем представлена следующими задачами Data Mining: классификация, оценка, прогнозирование.  
Обучение с учителем (supervised learning) предполагает наличие полного набора размеченных данных для тренировки модели на всех этапах ее построения.

Наличие полностью размеченного датасета означает, что каждому примеру в обучающем наборе соответствует ответ, который алгоритм и должен получить. Таким образом, размеченный датасет из фотографий цветов обучит нейронную сеть, где изображены розы, ромашки или нарциссы. Когда сеть получит новое фото, она сравнит его с примерами из обучающего датасета, чтобы предсказать ответ.



Пример обучения с учителем — классификация (слева), и дальнейшее ее использование для сегментации и распознавания объектов

В основном обучение с учителем применяется для решения двух типов задач: классификации и регрессии.

В задачах классификации алгоритм предсказывает дискретные значения, соответствующие номерам классов, к которым принадлежат объекты. В обучающем датасете с фотографиями животных каждое изображение будет иметь соответствующую метку — «кошка», «коала» или «черепаха». Качество алгоритма оценивается тем, насколько точно он может правильно классифицировать новые фото с коалами и черепахами.

Таким образом, обучение с учителем больше всего подходит для задач, когда имеется внушительный набор достоверных данных для обучения алгоритма. Но так бывает далеко не всегда. Недостаток данных — наиболее часто встречающаяся проблема в машинном обучении на 2018 год.

**40. Обучение без учителя**

Категория обучение без учителя представлена задачей кластеризации.  
Идеально размеченные и чистые данные достать нелегко. Поэтому иногда перед алгоритмом стоит задача найти заранее не известные ответы. Вот где нужно обучение без учителя.

В обучении без учителя (**unsupervised learning**) у модели есть набор данных, и нет явных указаний, что с ним делать. Нейронная сеть пытается самостоятельно найти корелляции в данных, извлекая полезные признаки и анализируя их.



Кластеризация данных на основе общих признаков

В зависимости от задачи модель систематизирует данные по-разному.

* [**Кластеризация**](https://neurohive.io/ru/papers/deep-claster/)**.** Даже без специальных знаний эксперта-орнитолога можно посмотреть на коллекцию фотографий и разделить их на группы по видам птиц, опираясь на цвет пера, размер или форму клюва. Именно в этом заключается кластеризация — наиболее распространенная задача для обучения без учителя. Алгоритм подбирает похожие данные, находя общие признаки, и группируют их вместе.
* **Обнаружение аномалий.** Банки могут обнаружить мошеннические операции, выявляя необычные действия в покупательском поведении клиентов. Например, подозрительно, если одна кредитная карта используется в Калифорнии и Дании в один и тот же день. Похожим образом, обучение без учителя используют для нахождения выбросов в данных.
* **Ассоциации.** Выберете в онлайн-магазине подгузники, яблочное пюре и детскую кружку-непроливайку и сайт порекомендует вам добавить нагрудник и радионяню к заказу. Это пример ассоциаций: некоторые характеристики объекта коррелируют с другими признаками. Рассматривая пару ключевых признаков объекта, модель может предсказать другие, с которыми существует связь.
* **Автоэнкодеры.** Автоэнкодеры принимают входные данные, кодируют их, а затем пытаются воссоздать начальные данные из полученного кода. Не так много реальных ситуаций, когда используют простой автоэнкодер. Но стоит добавить слои и возможности расширятся: используя зашумленные и исходные версии изображений для обучения, автоэнкодеры могут удалять шум из видеоданных, изображений или медицинских сканов, чтобы повысить качество данных.

В обучении без учителя сложно вычислить точность алгоритма, так как в данных отсутствуют «правильные ответы» или метки. Но размеченные данные часто ненадежные или их слишком дорого получить. В таких случаях, предоставляя модели свободу действий для поиска зависимостей, можно получить хорошие результаты.

**41. Задача классификации**

**Задача классификации** — предсказание категориальной зависимой переменной (т.е. зависимой переменной, являющейся категорией) на основе выборки непрерывных и/или категориальных переменных. (лекция)

**Классификация** — один из разделов [машинного обучения](http://www.machinelearning.ru/wiki/index.php?title=%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5), посвященный решению следующей задачи. Имеется множество объектов (ситуаций), разделённых некоторым образом на классы. Задано конечное множество объектов, для которых известно, к каким классам они относятся. Это множество называется [обучающей выборкой](http://www.machinelearning.ru/wiki/index.php?title=%D0%92%D1%8B%D0%B1%D0%BE%D1%80%D0%BA%D0%B0). Классовая принадлежность остальных объектов не известна. Требуется построить [алгоритм](http://www.machinelearning.ru/wiki/index.php?title=%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC), способный классифицировать произвольный объект из исходного множества.



**42. Методы классификации**

***Классификация* - системное распределение изучаемых предметов, явлений, процессов по родам, видам, типам, по каким-либо существенным признакам для удобства их исследования; группировка исходных понятий и расположение их в определенном порядке, отражающем степень этого сходства.**

***Классификация* - упорядоченное по некоторому принципу множество объектов, которые имеют сходные классификационные признаки (одно или несколько свойств), выбранных для определения сходства или различия между этими объектами.**

***Классификация* требует соблюдения следующих правил:**

**· в каждом акте деления необходимо применять только одно основание;**

**· деление должно быть соразмерным, т.е. общий объем видовых понятий должен равняться объему делимого родового понятия;**

**· члены деления должны взаимно исключать друг друга, их объемы не должны перекрещиваться;**

**· деление должно быть последовательным.**

**Различают:**

**· вспомогательную (искусственную) *классификацию*, которая производится по внешнему признаку и служит для придания множеству предметов (процессов, явлений) нужного порядка;**

**· естественную *классификацию*, которая производится по существенным признакам, характеризующим внутреннюю общность предметов и явлений. Она является результатом и важным средством научного исследования, т.к. предполагает и закрепляет результаты изучения закономерностей классифицируемых объектов.**

**В зависимости от выбранных признаков, их *сочетания* и процедуры деления понятий *классификация* может быть:**

**· простой - деление родового понятия только по признаку и только один раз до раскрытия всех видов. Примером такой *классификации* является дихотомия, при которой членами деления бывают только два понятия, каждое из которых является противоречащим другому (т.е. соблюдается принцип: "А и не А");**

**· сложной - применяется для деления одного понятия по разным основаниям и синтеза таких простых делений в единое целое. Примером такой *классификации* является периодическая система химических элементов.**

***Классификация* может быть *одномерной* (по одному признаку) и *многомерной* (по двум и более признакам).**

**Для *классификации* используются различные методы. Основные из них:**

**· *классификация* с помощью деревьев решений;**

**· байесовская (наивная) *классификация* ;**

**· *классификация* при помощи *искусственных нейронных сетей*;**

**· *классификация* методом *опорных векторов*;**

**· статистические методы, в частности, линейная регрессия;**

**· *классификация* при помощи метода ближайшего соседа;**

**· *классификация* *CBR*-методом;**

**· *классификация* при помощи *генетических алгоритмов*.**

**43. Метод опорных векторов**

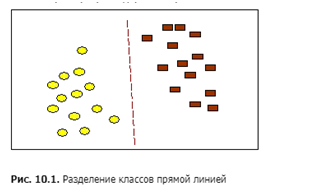
**Метод *опорных векторов* (*Support* *Vector Machine* - SVM) относится к группе граничных методов. Она определяет классы при помощи границ областей.**

**При помощи данного метода решаются задачи *бинарной классификации*. (зависимая переменная принимает только 2 значения)**

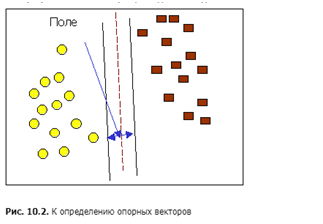
**В основе метода лежит понятие *плоскостей* решений.**

***Плоскость* (*plane*) решения разделяет объекты с разной классовой принадлежностью.**

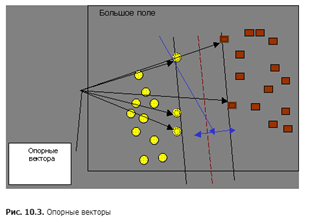
**На** [**рис.10.1**](https://intuit.ru/studies/courses/6/6/lecture/176?page=1#image.10.1) **приведен пример, в котором участвуют объекты двух типов. Разделяющая линия задает границу, справа от которой - все объекты типа brown (коричневый), а слева - типа yellow (желтый). Новый *объект*, попадающий направо, классифицируется как *объект* класса brown или - как *объект* класса yellow, если он расположился *по* левую сторону от разделяющей *прямой*. В этом случае каждый *объект* характеризуется двумя измерениями.**

****

**Цель метода *опорных векторов* - найти *плоскость*, разделяющую два *множества* объектов; такая *плоскость* показана на** [**рис. 10.2**](https://intuit.ru/studies/courses/6/6/lecture/176?page=1#image.10.2)**. На этом рисунке множество образцов поделено на два класса: желтые объекты принадлежат классу А, коричневые - классу В.**

****

**Метод отыскивает образцы, находящиеся на границах между двумя классами, т.е. *опорные вектора* ; они изображены на** [**рис. 10.3**](https://intuit.ru/studies/courses/6/6/lecture/176?page=1#image.10.3)**.**

****

***Опорными векторами* называются объекты *множества*, лежащие на границах областей.**

**Классификация считается хорошей, если область между границами пуста.**

**На** [**рис. 10.3**](https://intuit.ru/studies/courses/6/6/lecture/176?page=1#image.10.3)**.показано пять векторов, которые являются опорными для данного *множества*.**

**44. Метод «ближайшего соседа»**

**Следует сразу отметить, что метод "ближайшего соседа" ("nearest neighbour") относится к классу методов, работа которых основывается на хранении данных в памяти для сравнения с новыми элементами. При появлении новой записи для прогнозирования находятся отклонения между этой записью и подобными наборами данных, и наиболее подобная (или ближний сосед) идентифицируется.**

**Например, при рассмотрении нового клиента банка, его атрибуты сравниваются со всеми существующими клиентами данного банка (доход, возраст и т.д.). Множество "ближайших соседей" потенциального клиента банка выбирается на основании ближайшего значения дохода, возраста и т.д.**

**При таком подходе используется термин "k-ближайший сосед" ("k-nearest neighbour"). Термин означает, что выбирается k "верхних" (ближайших) соседей для их рассмотрения в качестве *множества* "ближайших соседей". Поскольку не всегда удобно хранить все данные, иногда хранится только множество "типичных" случаев. В таком случае используемый метод называют рассуждением *по* аналогии (*Case* Based *Reasoning*, *CBR*), рассуждением на основе аналогичных случаев, рассуждением *по* прецедентам.**

#### **Преимущества метода**

**· Простота использования полученных результатов.**

**· Решения не уникальны для конкретной ситуации, возможно их использование для других случаев.**

**· Целью поиска является не гарантированно верное решение, а лучшее из возможных.**

#### **Недостатки метода "ближайшего соседа"**

**· Данный метод не создает каких-либо моделей или правил, обобщающих предыдущий опыт, - в выборе решения они основываются на всем массиве доступных исторических данных, поэтому невозможно сказать, на каком основании строятся ответы.**

**· Существует сложность выбора меры "близости" (метрики). От этой меры главным образом зависит объем множества записей, которые нужно хранить в памяти для достижения удовлетворительной классификации или прогноза. Также существует высокая зависимость результатов классификации от выбранной метрики.**

**· При использовании метода возникает необходимость полного перебора *обучающей выборки* при распознавании, следствие этого - вычислительная трудоемкость.**

**· Типичные задачи данного метода - это задачи небольшой размерности по количеству классов и переменных.**

**С помощью данного метода решаются задачи классификации и регрессии.**

**45. Прецедент**

***Прецедент* - это описание ситуации в сочетании с подробным указанием действий, предпринимаемых в данной ситуации.**

**Подход, основанный на *прецедентах*, условно можно поделить на следующие этапы:**

**· сбор подробной информации о поставленной задаче;**

**· сопоставление этой информации с деталями *прецедентов*, хранящихся в базе, для выявления аналогичных случаев;**

**· выбор *прецедента*, наиболее близкого к текущей проблеме, из базы *прецедентов* ;**

**· адаптация выбранного решения к текущей проблеме, если это необходимо;**

**· проверка корректности каждого вновь полученного решения;**

**· занесение детальной информации о новом *прецеденте* в базу *прецедентов*.**

**Таким образом, *вывод*, основанный на *прецедентах*, представляет собой такой метод анализа данных, который делает заключения относительно данной ситуации *по* результатам поиска аналогий, хранящихся в базе *прецедентов*.**

**Данный метод *по* своей сути относится к категории "обучение без учителя", т.е. является "самообучающейся" технологией, благодаря чему рабочие характеристики каждой базы *прецедентов* с течением времени и накоплением примеров улучшаются. Разработка баз *прецедентов* *по* конкретной *предметной области* происходит на естественном для человека языке, следовательно, может быть выполнена наиболее опытными сотрудниками компании - экспертами или аналитиками, работающими в данной *предметной области*.**

**Однако это не означает, что *CBR*-системы самостоятельно могут принимать решения. Последнее всегда остается за человеком, данный метод лишь предлагает возможные варианты решения и указывает на самый "разумный" с ее точки зрения.**

**46. Деревья решений**

**Метод деревьев решений (*decision trees*) является одним из наиболее популярных методов решения задач классификации и прогнозирования. Иногда этот метод *Data Mining* также называют деревьями решающих *правил*, деревьями классификации и регрессии.**

**Как видно из последнего названия, при помощи данного метода решаются задачи классификации и прогнозирования.**

**Если зависимая, т.е. целевая *переменная* принимает дискретные значения, при помощи метода дерева решений решается задача классификации.**

**Если же зависимая *переменная* принимает непрерывные значения, то *дерево* решений устанавливает зависимость этой переменной от независимых переменных, т.е. решает задачу численного прогнозирования.**

**В наиболее простом виде *дерево* решений - это способ представления *правил* в иерархической, последовательной структуре. Основа такой структуры - ответы "Да" или "Нет" на ряд вопросов.**

**В узлах бинарных деревьев *ветвление* может вестись только в двух направлениях, т.е. существует возможность только двух ответов на поставленный вопрос ("да" и "нет").**

**Бинарные деревья являются самым простым, частным случаем деревьев решений. В остальных случаях, ответов и, соответственно, *ветвей* дерева, выходящих из его *внутреннего узла*, может быть больше двух.**

**Задача классификации решается в два этапа: построение классификационной модели и ее использование.**

**На этапе построения модели, собственно, и строится *дерево* классификации или создается набор неких *правил*. На этапе использования модели построенное *дерево*, или *путь* от его корня к одной из вершин, являющийся набором *правил* для конкретного клиента, используется для ответа на поставленный вопрос "Выдавать ли *кредит*?"**

***Правилом* является логическая конструкция, представленная в виде "если : то :".**

**Каждая *ветвь* дерева, идущая от *внутреннего узла*, отмечена *предикатом расщепления*. Последний может относиться лишь к одному *атрибуту расщепления* данного узла. Характерная особенность *предикатов расщепления*: каждая *запись* использует уникальный *путь* от корня дерева только к одному узлу-решению. Объединенная *информация* об *атрибутах расщепления* и *предикатах расщепления* в узле называется *критерием расщепления* (splitting *criterion*)**

**Качество построенного дерева решения весьма зависит от правильного выбора *критерия расщепления*. Над разработкой и усовершенствованием критериев работают многие исследователи.**

**Метод деревьев решений часто называют "наивным" подходом . Но благодаря целому ряду преимуществ, данный метод является одним из наиболее популярных для решения задач классификации.**

**Преимущества метода**

**Интуитивность деревьев решений. Классификационная модель, представленная в виде дерева решений, является интуитивной и упрощает понимание решаемой задачи.**

**Деревья решений дают возможность извлекать *правила* из *базы данных* на естественном языке.**

***Алгоритм* конструирования дерева решений не требует от пользователя выбора входных атрибутов (независимых переменных).**

**Точность моделей, созданных при помощи деревьев решений, сопоставима с другими методами построения классификационных моделей (*статистические методы*, нейронные сети).**

**Быстрый процесс обучения. На построение классификационных моделей при помощи алгоритмов конструирования деревьев решений требуется значительно меньше времени, чем, например, на обучение нейронных сетей.**

**Большинство алгоритмов конструирования деревьев решений имеют возможность специальной обработки пропущенных значений.**

**Многие классические *статистические методы*, при помощи которых решаются задачи классификации, могут работать только с числовыми данными, в то время как деревья решений работают и с числовыми, и с категориальными типами данных.**

**47. Регрессионный анализ**

**Регрессионный анали**з -­ статистический аналитический метод, позволяющий вычислить предполагаемые отношения между зависимой переменной одной или несколькими независимыми переменными. Используя регрессионный анализ, вы можете моделировать отношения между выбранным переменными, а также прогнозируемыми значениями на основе модели.

**Регрессионный анализ** — метод моделирования измеряемых данных и исследования их свойств. Данные состоят из пар значений зависимой переменной (переменной отклика) и независимой переменной (объясняющей переменной). [Регрессионная модель](http://www.machinelearning.ru/wiki/index.php?title=%D0%A0%D0%B5%D0%B3%D1%80%D0%B5%D1%81%D1%81%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C) есть функция независимой переменной и параметров с добавленной [случайной переменной](http://www.machinelearning.ru/wiki/index.php?title=%D0%A1%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B5%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F&action=edit). Параметры модели настраиваются таким образом, что модель наилучшим образом приближает данные. Критерием качества приближения (целевой функцией) обычно является [среднеквадратичная ошибка](http://www.machinelearning.ru/wiki/index.php?title=%D0%A1%D1%80%D0%B5%D0%B4%D0%BD%D0%B5%D0%BA%D0%B2%D0%B0%D0%B4%D1%80%D0%B0%D1%82%D0%B8%D1%87%D0%BD%D0%B0%D1%8F_%D0%BE%D1%88%D0%B8%D0%B1%D0%BA%D0%B0&action=edit): сумма квадратов разности значений модели и зависимой переменной для всех значений независимой переменной в качестве аргумента. Регрессионный анализ — раздел [математической статистики](http://www.machinelearning.ru/wiki/index.php?title=%D0%9C%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%81%D1%82%D0%B0%D1%82%D0%B8%D1%81%D1%82%D0%B8%D0%BA%D0%B0) и [машинного обучения](http://www.machinelearning.ru/wiki/index.php?title=%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5). Предполагается, что зависимая переменная есть сумма значений некоторой модели и [случайной величины](http://www.machinelearning.ru/wiki/index.php?title=%D0%A1%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D0%B0%D1%8F_%D0%B2%D0%B5%D0%BB%D0%B8%D1%87%D0%B8%D0%BD%D0%B0). Относительно характера распределения этой величины делаются предположения, называемые гипотезой порождения данных. Для подтверждения или опровержения этой гипотезы выполняются [статистические тесты](http://www.machinelearning.ru/wiki/index.php?title=%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%81%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D1%82%D0%B5%D1%81%D1%82), называемые [анализом остатков](http://www.machinelearning.ru/wiki/index.php?title=%D0%90%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7_%D0%BE%D1%81%D1%82%D0%B0%D1%82%D0%BA%D0%BE%D0%B2&action=edit). При этом предполагается, что независимая переменная не содержит ошибок. Регрессионный анализ используется для [прогноза](http://www.machinelearning.ru/wiki/index.php?title=%D0%9F%D1%80%D0%BE%D0%B3%D0%BD%D0%BE%D0%B7), [анализа временных рядов](http://www.machinelearning.ru/wiki/index.php?title=%D0%90%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7_%D0%B2%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D1%8B%D1%85_%D1%80%D1%8F%D0%B4%D0%BE%D0%B2&action=edit), [тестирования гипотез](http://www.machinelearning.ru/wiki/index.php?title=%D0%A2%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%B3%D0%B8%D0%BF%D0%BE%D1%82%D0%B5%D0%B7&action=edit) и выявления скрытых взаимосвязей в данных.

**48. Задача поиска ассоциативных правил**

В последнее время неуклонно растет интерес к методам 'обнаружения знаний в базах данных'. Объемы современных баз данных, которые весьма внушительны, вызвали устойчивый спрос на новые масштабируемые алгоритмы анализа данных. Одним из популярных методов обнаружения знаний стали алгоритмы поиска ассоциативных правил.

Ассоциативные правила позволяют находить закономерности между связанными событиями. Примером такого правила, служит утверждение, что покупатель, приобретающий 'Хлеб', приобретет и 'Молоко' с вероятностью 72%. Первый алгоритм поиска ассоциативных правил, называвшийся AIS был разработан в 1993 году сотрудниками исследовательского центра IBM Almaden. С этой пионерской работы возрос интерес к ассоциативным правилам; на середину 90-х годов прошлого века пришелся пик исследовательских работ в этой области, и с тех пор каждый год появлялось по несколько алгоритмов.

1.1 Ассоциативные правила

Впервые это задача была предложена поиска ассоциативных правил для нахождения типичных шаблонов покупок, совершаемых в супермаркетах, поэтому иногда ее еще называют анализом рыночной корзины (market basket analysis).

Пусть имеется база данных, состоящая из покупательских транзакций. Каждая транзакция – это набор товаров, купленных покупателем за один визит. Такую транзакцию еще называют рыночной корзиной.

Пусть I = {i1, i2, i3, …in} – множество (набор) товаров, называемых элементами. Пусть D – множество транзакций, где каждая транзакция T – это набор элементов из I, T I. Каждая транзакция представляет собой бинарный вектор, где t[k]=1, если ik элемент присутствует в транзакции, иначе t[k]=0. Мы говорим, что транзакция T содержит X, некоторый набор элементов из I, если X T. Ассоциативным правилом называется импликация X Y, где X I, Y I и X Y = . Правило X Y имеет поддержку s (support), если s% транзакций из D, содержат X Y, supp(X Y) = supp (X Y). Достоверность правила показывает какова вероятность того, что из X следует Y. Правило X Y справедливо с достоверностью (confidence) c, если c% транзакций из D, содержащих X, также содержат Y, conf(X Y) = supp(X Y)/supp(X ).

Покажем на конкретном примере: '75% транзакций, содержащих хлеб, также содержат молоко. 3% от общего числа всех транзакций содержат оба товара'. 75% – это достоверность (confidence) правила, 3% это поддержка (support), или 'Хлеб' 'Молоко' с вероятностью 75%. Другими словами, целью анализа является установление следующих зависимостей: если в транзакции встретился некоторый набор элементов X, то на основании этого можно сделать вывод о том, что другой набор элементов Y также же должен появиться в этой транзакции. Установление таких зависимостей дает нам возможность находить очень простые и интуитивно понятные правила.

Алгоритмы поиска ассоциативных правил предназначены для нахождения всех правил X Y, причем поддержка и достоверность этих правил должны быть выше некоторых наперед определенных порогов, называемых соответственно минимальной поддержкой (minsupport) и минимальной достоверностью (minconfidence).

Задача нахождения ассоциативных правил разбивается на две подзадачи:

1. Нахождение всех наборов элементов, которые удовлетворяют порогу minsupport. Такие наборы элементов называются часто встречающимися.

2. Генерация правил из наборов элементов, найденных согласно п.1. с достоверностью, удовлетворяющей порогу minconfidence.

Значения для параметров минимальная поддержка и минимальная достоверность выбираются таким образом, чтобы ограничить количество найденных правил. Если поддержка имеет большое значение, то алгоритмы будут находить правила, хорошо известные аналитикам или настолько очевидные, что нет никакого смысла проводить такой анализ. С другой стороны, низкое значение поддержки ведет к генерации огромного количества правил, что, конечно, требует существенных вычислительных ресурсов. Тем не менее, большинство интересных правил находится именно при низком значении порога поддержки. Хотя слишком низкое значение поддержки ведет к генерации статистически необоснованных правил.

Поиск ассоциативных правил совсем не тривиальная задача, как может показаться на первый взгляд. Одна из проблем – алгоритмическая сложность при нахождении часто встречающих наборов элементов, т.к. с ростом числа элементов в I (| I |) экспоненциально растет число потенциальных наборов элементов.

**49. Алгоритм A`priori**

# Apriori — масштабируемый алгоритм поиска ассоциативных правил

Алгоритм Apriori использует стратегию [поиска в ширину](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B8%D1%81%D0%BA_%D0%B2_%D1%88%D0%B8%D1%80%D0%B8%D0%BD%D1%83) для подсчёта объектов и использует функцию генерации кандидата, которая основана на свойстве нисходящего замыкания поддержки.

Современные базы данных имеют очень большие размеры, достигающие гига- и терабайтов, и тенденцию к дальнейшему увеличению. И поэтому, для нахождения [ассоциативных правил](https://wiki.loginom.ru/articles/association-rules.html) требуются эффективные [масштабируемые алгоритмы](https://wiki.loginom.ru/articles/scalable-algorithm.html), позволяющие решить задачу за приемлемое время. Мы опишем алгоритм [Apriori](https://wiki.loginom.ru/articles/apriori.html).

Для того чтобы было возможно применить алгоритм, необходимо провести предобработку данных:

* привести все данные к бинарному виду;
* изменить структуру данных.

Алгоритм Apriori ищет ассоциативные правила и применяется по отношению к базам данных, содержащим огромное количество транзакций.

Как же работает алгоритм Apriori? Перед тем, как перейти к сути алгоритма, вам нужно определить 3 параметра:

1. Во-первых, нужно установить **размер** набора. Вы хотите определить двухэлементный, трёхэлементный набор или какой-нибудь еще?
2. Во-вторых, определить **поддержку** – это число транзакций, входящих в набор, разделенное на общее количество транзакций. Набор, который равен поддержке, является самым часто встречаемым набором.
3. В-третьих, определить **достоверность**, то есть условную вероятность определенного товара оказаться в корзине с другими товарами. Пример: чипсы в вашем наборе имеют 67%-ную вероятность оказаться в одной корзине с газировкой.

Простой алгоритм Apriori состоит из трех шагов:

1. **Объединение**. Просмотр базы данных и определение частоты вхождения отдельных товаров.
2. **Отсечение**. Те наборы, которые удовлетворяют поддержке и достоверности, переходят на следующую итерацию с двухкомпонентными наборами,
3. **Повторение**. Предыдущие два шага повторяются для каждой величины набора, пока не будет повторно получен ранее определенный размер.

**50. Задача прогнозирования**

**Прогнозирование** (от греческого Prognosis), в широком понимании этого слова, определяется как опережающее отражение будущего. Целью прогнозирования является предсказание будущих событий.

**Прогнозирование** (forecasting) является одной из задач Data Mining и одновременно одним из ключевых моментов при принятии решений.

**Прогностика** (prognostics) - теория и практика прогнозирования.

**Прогнозирование** направлено на определение тенденций динамики конкретного объекта или события на основе ретроспективных данных, т.е. анализа его состояния в прошлом и настоящем. Таким образом, решение задачи прогнозирования требует некоторой обучающей выборки данных.

**Прогнозирование** - установление функциональной зависимости между зависимыми и независимыми переменными.

**Прогнозирование** является распространенной и востребованной задачей во многих областях человеческой деятельности. В результате прогнозирования уменьшается риск принятия неверных, необоснованных или субъективных решений.

**Главная цель прогнозирования** – провести научный анализ социально-экономических процессов и тенденций, предвидеть новые экономические ситуации и выявить основные экономические проблемы.

При этом в прогнозирование входит:

1.Исследование объективных связей выбранного явления (процесса) для конкретных условий и при соответствующем уровне экономического и общественного развития,

2.Оценка объектов прогнозирования,

3.Выявление всех возможных будущих альтернатив развития,

4.Принятие оптимального решения.

Принципы прогнозирования

В современном мире выделяют следующие принципы, на которых основан процесс прогнозирования:

Научное обоснование прогнозов (применение при разработке научных методов, сообразование с закономерностями развития природы и общественной жизнью), Непрерывное прогнозирование, с осуществлением постоянных корректировок (например, в ситуации изменений в экономике);

Перспективное и текущее прогнозирование, с приоритетом на перспективу,

Согласованность прогнозирования (взаимосвязь со смежными планами);

Альтернативность прогнозов (наличие нескольких его вариантов с целью внесения корректив при наступлении одного из них);

Учет основных факторов прогнозирования, которые оказывают влияние на прогноз;

Системность в разработке прогноза (прогноз стоит рассматривать как целое, но состоящее из взаимосвязанных частей);

Верифицируемость (достоверность и обоснованность);

Адекватность прогноза (прогнозная модель должна быть по максимуму приближена к фактам реальной действительности);

Рентабельность прогнозирования (затраты не должны превышать затраты на прогнозирование)

**51. Задача кластеризации**

**Кластеризация** (англ. *cluster analysis*) — задача группировки множества объектов на подмножества (кластеры) таким образом, чтобы объекты из одного кластера были более похожи друг на друга, чем на объекты из других кластеров по какому-либо критерию.

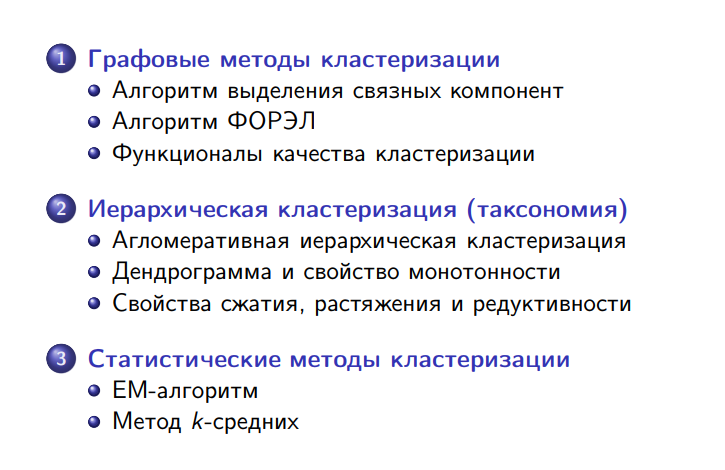
Задача кластеризации (то же самое, что кластерный анализ) решена, если получено разбиение множества объектов на подмножества, называемых кластерами. Основное условие решения: объекты, принадлежащие одному кластеру должны быть больше похожи друг на друга, чем объекты из других кластеров. Критерий «похожести» может быть один, но их может быть и несколько.

В процессе кластеризации вводится понятие меры расстояний между объектами (различий значений признака). Для объектов, которые окажутся в одном кластере, расстояние должно быть одного порядка малости.

Вопреки распространенному заблуждению, популярное средство для статистической обработки данных Excel не обладает функциями кластерного анализа. Весьма эффективно задача кластеризации решается с применением языков программирования Python и R в профессиональных средах разработки.

**52. Методы кластеризации**

* Алгоритмы, основанные на разделении данных, в т.ч. итеративные:
  + разделение объектов на k кластеров;
  + итеративное перераспределение объектов для улучшения кластеризации.
* Иерархические алгоритмы:
  + агломерация: каждый объект первоначально является кластером, кластеры, соединяясь друг с другом, формируют больший кластер и т.д.
* Методы, основанные на концентрации объектов:
  + основаны на возможности соединения объектов;
  + игнорируют шумы, нахождение кластеров произвольной формы.
* Грид-методы:
  + квантование объектов в грид-структуры.
* Модельные методы:
  + использование модели для нахождения кластеров, наиболее соответствующих данным**.**

****

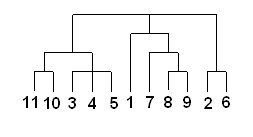
**53. Дендрограммы**

Дендрограмма (dendrogram) - древовидная диаграмма, содержащая n уровней, каждый из которых соответствует одному из шагов процесса последовательного укрупнения кластеров.

Дендрограмму также называют древовидной схемой, деревом объединения кластеров, деревом иерархической структуры.

Дендрограмма представляет собой вложенную группировку объектов, которая изменяется на различных уровнях иерархии.

Существует много способов построения дендрограмм. В дендрограмме объекты могут располагаться вертикально или горизонтально.

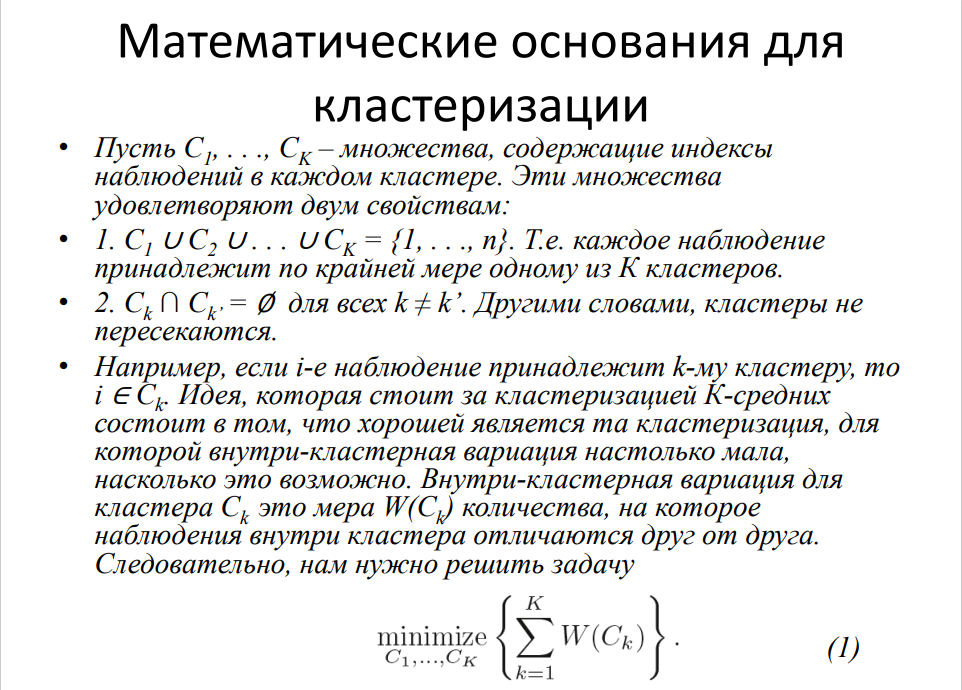


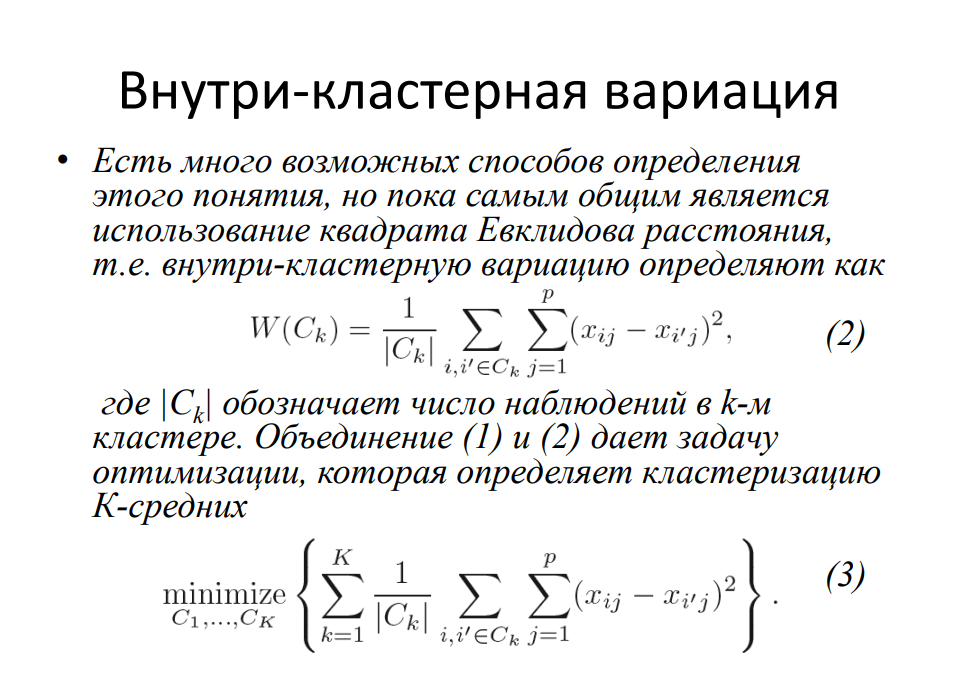
**Пример дендрограммы**

**Объяснение примера:** Числа 11, 10, 3 и т.д. соответствуют номерам объектов или наблюдений исходной выборки. Мы видим, что на первом шаге каждое наблюдение представляет один кластер (вертикальная линия), на втором шаге наблюдаем объединение таких наблюдений: 11 и 10; 3, 4 и 5; 8 и 9; 2 и 6. На втором шаге продолжается объединение в кластеры: наблюдения 11, 10, 3, 4, 5 и 7, 8, 9. Данный процесс продолжается до тех пор, пока все наблюдения не объединятся в один кластер.

**54. Алгоритм k-средних**

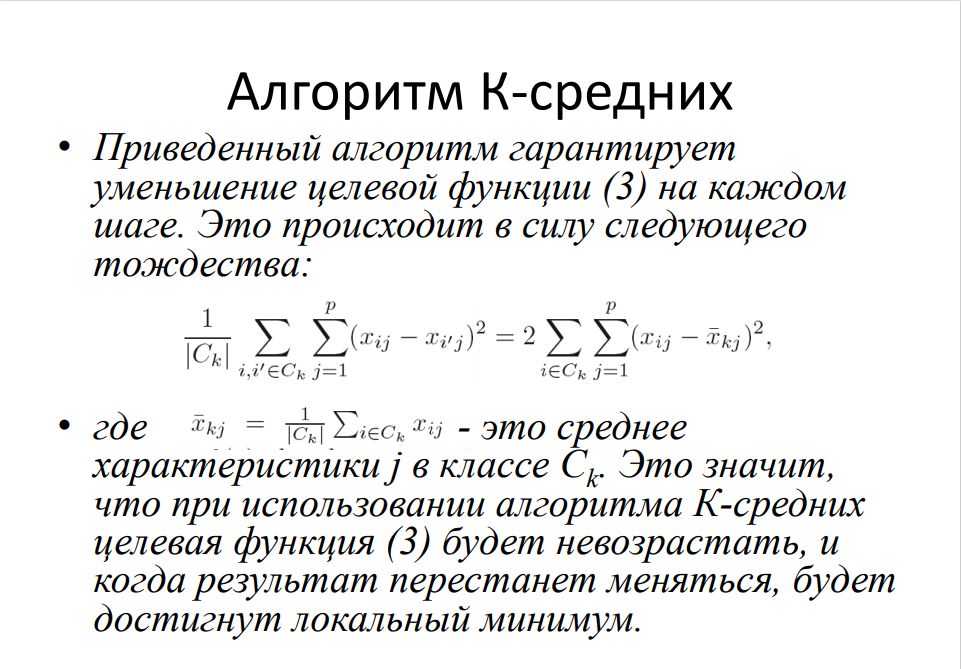
Кластеризация К-средних – простой метод разделения множества данных на К различных непересекающихся кластеров. Для выполнения кластеризации сначала нужно определить желаемое число кластеров К, затем алгоритм К-средних будет относить каждое наблюдение в точности к одному из К кластеров.

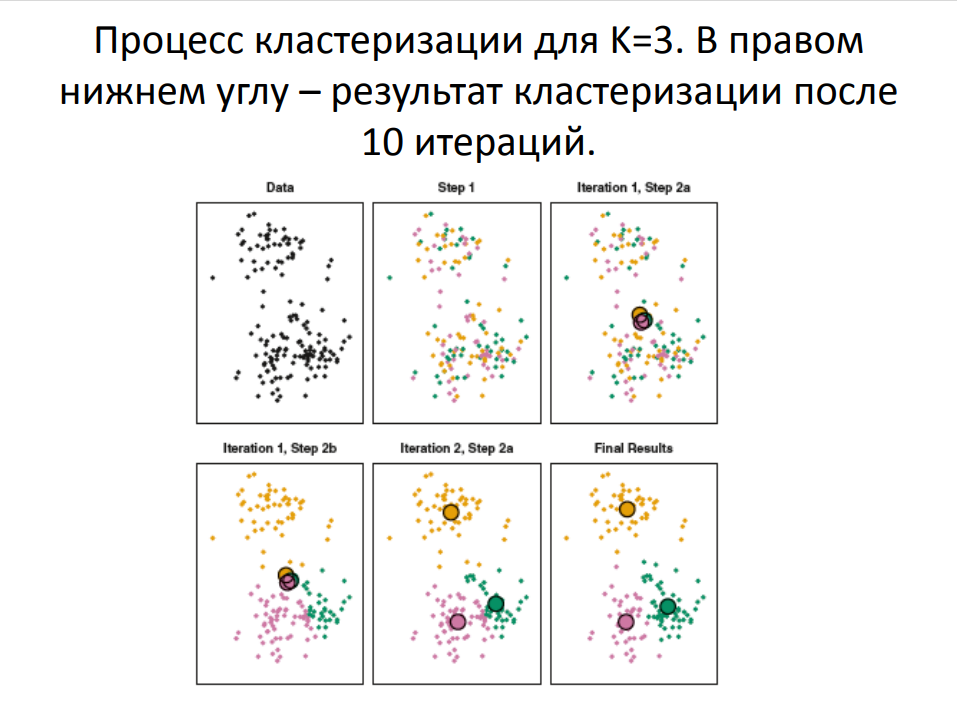




Следующий алгоритм находит локальный минимум задачи (3) – достаточно хорошее решение.

1. Произвольным образом присваиваем номер от 1 до К каждому из наблюдений. Это послужит начальным разделением на классы.
2. Повторяем, пока разбиение на классы не перестанет меняться:
   1. Для каждого из К кластеров вычисляем кластерный центроид. Центроид k-го кластера – это вектор средних p характеристик для наблюдений в k-м кластере.
   2. Относим каждое наблюдение к тому кластеру, чей центроид ближайший (где ближайший определяется с использованием Евклидова расстояния).





Так как алгоритм К-средних находит локальный, а не глобальный минимум, его результат зависит от выбора начальных (случайных) кластеров. Поэтому важно запустить алгоритм несколько раз для различных случайных начальных конфигураций. Затем выбирается наилучшее решение, т.е. то, для которого значение целевой функции (3) будет минимальным.

**55. Прогнозирование временных рядов**

Временной ряд — последовательность наблюдаемых значений какого либо признака, упорядоченных в неслучайные моменты времени.

* Члены временного ряда, в отличие от элементов случайной выборки, не являются статистически независимыми.
* Члены временного ряда не являются одинаково распределенными.

Анализ временного ряда осуществляется с целью:

* Определение природы ряда.
* Прогнозирование будущих значений ряда.

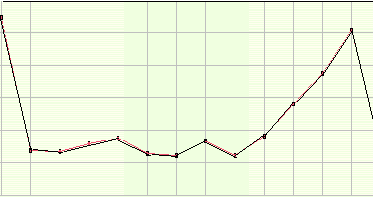
**56. Тренд, сезонность, цикл**

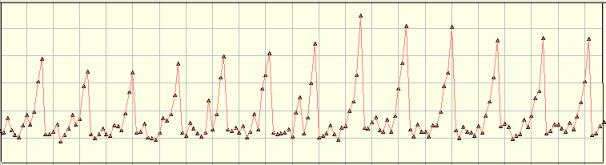
Тренд — неслучайная функция, которая формируется под действием общих или долговременных тенденций, влияющих на временной ряд.

* прогнозирование односерийных рядов;
* прогнозирование мультисерийных, или взаимовлияющих, рядов.

**Сезонность** - периодически колебания, наблюдаемые на временных рядах.

Свойство сезонности означает, что через примерно равные промежутки времени форма кривой, которая описывает поведение зависимой переменной, повторяет свои характерные очертания.





Отличия циклической компоненты от сезонной:

* Продолжительность цикла, как правило, больше, чем один сезонный период;
* Циклы, в отличие от сезонных периодов, не имеют определенной продолжительности.

# **57. Период, горизонт, интервал прогнозирования**

Горизонт прогнозирования — число периодов в будущем, которые покрывают прогноз.

Горизонт прогнозирования — это временной интервал, в пределах которого прогноз выполняется с заданной точностью.

Период прогнозирования — основная единица времени на который делается прогноз.

Интервал прогнозирования — это оценка интервала, в который с определенной вероятностью попадет будущее наблюдение, учитывая то, что уже наблюдалось.

Интервалы прогнозирования часто используются в регрессионном анализе.

# **58. Виды прогнозов**

* Краткосрочный прогноз (до 1 года; содержат общие количественные показатели);
* Среднесрочный прогноз (разрабатываются на период 1-5 лет и содержат не только количественные, но также и общие качественные оценки);
* Долгосрочный прогноз (разрабатываются на 5-15 лет; содержащие как общие количественные, так и общие качественные показатели).

# **59. Виды визуализации результатов анализа данных**

Визуализация — это инструментарий, который позволяет увидеть конечный результат вычислений, организовать управление вычислительным процессом и даже вернуться назад к исходным данным, чтобы определить наиболее рациональное направление дальнейшего движения.

Виды визуализации результатов анализа данных:

* Графики и диаграммы;
* Инфорграфика и схемы;
* Презентация и анализ данных;
* Интерактивный сторителлинг;
* Бизнес аналитика и дашборды;
* Научная и медицинская визуализация;
* Карты и картограммы.

# **60. Уровни интеграции данных**

**Интеграция данных** (англ. — Data integration) — процесс объединения данных из различных источников для получения их согласованного представления, в широком смысле — процесс организации регулярного обмена данными между различными ИС предприятия.

* Интеграция данных на физическом уровне (конверсия данных из различных источников в требуемый единый формат их физического представления);
* Интеграция данных на логическом уровне (доступ к данным, содержащимся в различных источниках, в терминах единой глобальной схемы свойств данных;
* Интеграция данных на семантическом уровне (единое представление данных с учетом их семантических свойств в контексте единой онтологии предметной области).

Сложности интеграции данных

• Физический уровень — разные форматы

• Логический уровень — разные модели/схемы данных

• Семантический уровень — разные онтологии/понятийная система

Проблемы интеграции данных

• Разнородность (гетерогенность)

• Автономность

• Распределенность

# **61. ETL**

ETL (Extract, Transform, Load — «извлечение, преобразование, загрузка», 60-80% времени) — один из основных процессов в управлении хранилищами данных, который включает в себя:

* извлечение данных из внешних источников;
* их трансформация и очистка, чтобы они соответствовали потребностям бизнес-модели;
* и загрузка их в хранилище данных.

В процессе преобразования данных в рамках ETL чаще всего выполняются следующие операции(по лекции):

* Извлечение и очистка данных;
* Трансформации данных;
* Загрузка данных.

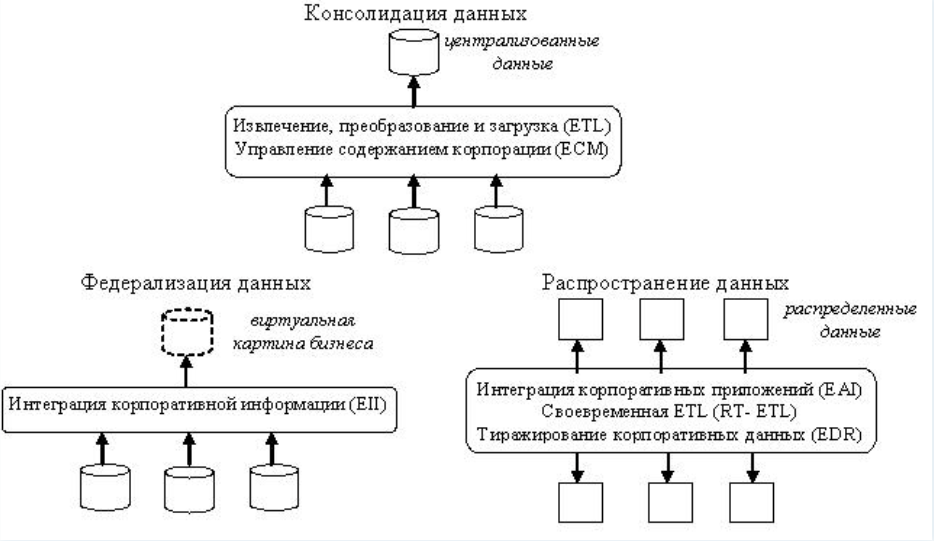
Извлечение, Преобразование, Загрузка — ETL.

Извлечение, Загрузка, Преобразование — E-LT.

# **62. Проблемы интеграции данных**

* Разнородность (гетерогенность) источники данных используют разные модели (и даже метамодели).
* Автономность, то есть источники разработаны и эксплуатируются независимо друг от друга, независимо спроектированы под решение конкретных, различных задач, различными методами.
* Распределенность источники физически или логически доступны только через сетевые протоколы удаленного доступа, в частности, информационные источники могут быть распределены в сети Интернет.

# **63. Методы интеграции/объединения данных**



* Консолидация;
* Федерализация;
* Распространение;
* SOA (Сервисный подход);
* Гибридный подход(?).
* Семантическая интеграция;

Интеграция делится на:

* Синтаксис (консолидация, федерализация, распространение, SOA, гибридный подход);
* Семантика.

К интеграции данных есть два альтернативных подхода: старый – *синтаксический* и новый – *семантический*. Первый основывается на внешнем сходстве объединяемых данных, второй на содержательном. Например, если мы, следуя первому, объединяем две обычные таблицы, то предполагаем, что в поле «Температура» значения выражены по одной шкале, а если мы смогли бы каким-то образом организовать семантическое хранение, то тогда могли бы рассчитывать на то, что в поле «Температура» находятся данные и метаданные, то есть запись физической величины и указание на то, по какой шкале она измерена, и смогли бы объединить массивы данных, записанных и по Цельсию, и по Фаренгейту.

Семантическая интеграция основывается на знании и учете природы данных. Разумеется, хранение данных вместе с метаданными создает дополнительные сложности, но обеспечивает большее удобство работы

# **64. Консолидация**

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

Основные критерии оптимальности с точки зрения консолидации данных:

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

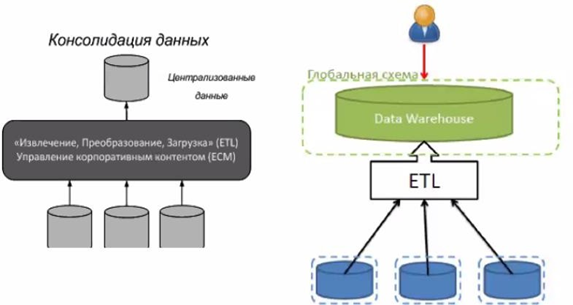
Ключевым понятием консолидации является источник данных — объект, содержащий структурированные данные, которые могут оказаться полезными для решения аналитической задачи. Необходимо, чтобы используемая аналитическая платформа могла осуществлять доступ к данным из этого объекта непосредственно либо после их преобразования в другой формат. В противном случае очевидно, что объект не может считаться источником данных.

Аналитические приложения, как правило, не содержат развитых средств ввода и редактирования данных, а работают с уже сформированными выборками. Таким образом, формирование массивов данных для анализа в большинстве случаев ложится на плечи заказчиков аналитических решений.

### **Основные задачи консолидации данных**

В процессе консолидации данных решаются следующие задачи:

* выбор источников данных;
* разработка стратегии консолидации;
* оценка качества данных;
* обогащение;
* очистка;
* перенос в хранилище данных.

****

Преимущество: трансформация довольно больших объемов данных.

ETL процессы

(Extraction, Transformation, Load) 60-80% времени

* Извлечение и очистка данных
* Трансформации данных
* Загрузка данных

При использовании этого метода данные собираются из нескольких первичных систем и интегрируются в одно постоянное место хранения.

# **65. Федерализация**

Федерализация — это обеспечение единой виртуальной картины одного или нескольких источников исходных данных. Для реализации данного подхода используется технология интеграции корпоративной информации (Enterprise Information Integration), которая позволяет извлекать данные из различных источников данных, объединять их и представлять аналитику в режиме реального времени.

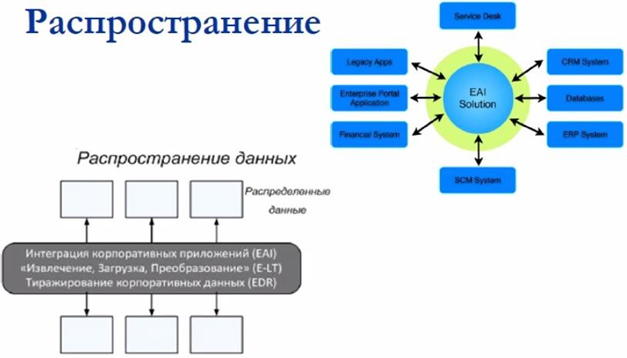
****

Преимущество: обеспечивает доступ к текущим данным. Не подходит для больших объемов — нужна консолидация. Используется, когда стоимость консолидации превышает ее преимущества.

При использовании этого метода данные виртуально фиксируются в одной системе и формируют целевые запросы в другую систему.

# **66. Распространение**

Метод распространения данных предполагает копирование данных из одного места в другое. Такие приложения обычно работают в оперативном режиме и производят перемещение данных к местам назначения, т. е. зависят от определенных событий. Обновления в первичной системе могут передаваться в конечную систему синхронно или асинхронно, причем метод распространения гарантирует доставку данных в систему назначения, что является ключевым признак распространения данных. Большинство технологий синхронного распространения данных поддерживают двусторонний обмен данными между первичными и конечными системами. Необходимым компонентом в этом методе является брокер обмена сообщений, управляющий семантикой обмена и обеспечивающий гарантированную доставку сообщений.

****

**!!!**

Извлечение, Преобразование, Загрузка — ETL

Извлечение, Загрузка, Преобразование — E-LT

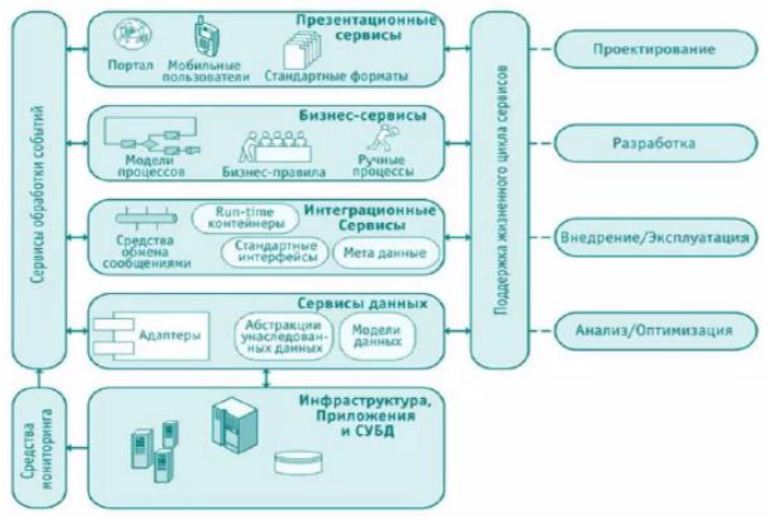
При использовании этого метода данные физически копируются из одного места в другое, и пока не попадут в некоторую целевую систему.

# **67. Сервисно-ориентированная архитектура**

SOA (service-oriented architecture) **—** это прикладная архитектура, в которой все функции определены как независимые сервисы с вызываемыми интерфейсами. Обращение к этим сервисам в определенной последовательности позволяет реализовать тот или иной бизнес-процесс.

SOA характеризуют следующие основные принципы:

* Все сервисы независимы друг от друга. Они выполняют определенные действия по запросам, полученные действия по запросам, полученным от других сервисов, и возвращают результаты. Все детали этого полностью скрыты: в концепции SOA сервисы – это «черные ящики»;
* Каждый, составляющий информационную систему сервис реализует отдельную бизнес функцию, которая является логически обособленной, повторяющейся задачей, являющейся составной частью бизнес-процесса предприятия;
* SOA предполагает наличие единой схемы обращения к сервису независимо от того, находится ли они в том же самом приложении, в другом адресном пространстве многопроцессорной системы, на другой аппаратной платформе в корпоративной intranet-сети или в приложении в системе партнера.

****

# **68. Классификация синтаксической интеграции данных**

Синтаксическая интеграция данных:

* Консолидация;
* Федерализация;
* Распространение;
* SOA;
* Гибридный подход.

**69. Универсальные форматы обмена данными**XML (Extensible Markup Language) – простой, очень гибкий текстовый формат, являющийся подмножеством SGML (ISO 8879), который позволяет определять собственные теги и атрибуты. Язык называется расширяемым, поскольку он не фиксирует разметку, используемую в документах: разработчик волен создать ее в соответствии с особенностями конкретной предметной области, будучи ограниченным лишь синтаксическими правилами языка. Возможность создания собственных тегов делает XML универсальным.

JSON (Java Script Object Notation) – представляет собой облегченный формат обмена данными между компьютерами. В соответствии с определением стандарта сценарного языка программирования ECMA (Европейской ассоциации производителей компьютеров), он является производным от литералов Java Script. JSON более компактен, чем XML, его конструкции легче анализируются средствами Java Script, для которого JSON является внутренним используемым типом данных. Основная сфера применения JSON – программирование web-приложений, где он служит альтернативой XML.

YAML – человекочитаемый формат сериализации данных, концептуально близкий к языкам разметки, но ориентированный на удобство ввода-вывода типичных структур данных многих языков программирования. В настоящее время акроним YAML интерпретируется как «YAML Ain’t Markup Language» («YAML – не язык разметки»). В названии отражена история развития: на ранних этапах акроним представлял собой аббревиатуру выражения «Yet Another Markup Language» («Ещё один язык разметки») и даже рассматривался как конкурент XML, но позже был переименован, чтобы акцентировать внимание на данных, а не на разметке документов.

# **70. Семантическая интеграция**

Семантическая интеграция подразумевает, что информационные системы обмениваются данными [в такой форме](http://www.business-semantic.ru/tech/semantic/), которая полностью абстрагирована от внутренней структуры каждой из них. Информация записывается в виде "высказываний" при помощи определенного "словаря"; системы отправляют и получают эти сообщения по различным правилам. При этом технически информация может передаваться путем записи в файлы, или через веб-сервисы.

Ориентирована на отношения между единицами данных вне зависимости от их схожести. Устанавливаются связи между единицами данных в соответствии с определениями в общих для них онтологиях.

****

# **71. Онтологии**

**Онтология** – это точная спецификация некоторой предметной области

Онтология — это попытка всеобъемлющей и подробной формализации некоторой области знаний с помощью концептуальной схемы. Обычно такая схема состоит из структуры данных, содержащей все релевантные классы объектов, их связи и правила (теоремы, ограничения), принятые в этой области.

Онтологии строятся по большей части одинаково, независимо от языка написания. Обычно они состоят из:

* экземпляров;
* понятий;
* атрибутов;
* отношений.

Сферы применения: моделирование бизнес-процессов, семантическая паутина, искусственный интеллект.

**72. Анализ текстовой информации**

**Информационный поис**к — отыскание информации слабо структурированного типа, отвечающей информационной потребности, среди большого объёма информации. Информации в данном случае присуще то, что она обычно представлена в виде текстовых документов и хранится в электронном виде. Структурирование — потеря полезной информации

Интеллектуальный анализ текста, или text mining — автоматизация извлечения сведений из текстовых данных. Его особенность (в отличие от анализа других данных) заключается в не формализованности исходной информации: ее не описать простой математической функцией. Ключевыми группами задач ИАТ являются: категоризация текстов, извлечение информации и информационный поиск, обработка изменений в коллекциях текстов, а также разработка средств представления информации для пользователя.

По сути, text mining, или извлечение информации из неструктурированных текстов – это текстовое направление data mining, то есть интеллектуального анализа данных. Основное отличие text mining от последнего заключается не в цели (а цель у этих направлений одна – извлечение информации для принятия некоего решения), а в задачах.

**Релевантность** — устанавливаемое при информационном поиске соответствие содержания документа информационному запросу или поискового образа документа поисковому предписанию.

**Поисковый образ документа** — это выраженное в терминах формализованного информационно-поискового языка основное смысловое содержание этого документа

**Элементы Text Mining**

1. классификация

2. кластеризация

3. прогнозирование

4. нахождение исключений

5. поиск связанных признаков

6. извлечение фактов, понятий

7. реферирование

8. ответ на запросы

9. тематическое индексирование

10. поиск по ключевым словам

**73. Задачи Text Mining**

1. Извлечение из текста характерных элементов или признаков, которые могут использоваться в качестве ключевых слов, метаданных, аннотаций.
2. Отнесение документов к некоторым категориям из заданной схемы их систематизации

Задача text mining – находить новые знания в больших объемах *неструктурированной* текстовой информации. С другой стороны, text mining можно рассматривать как расширение интеллектуального анализа данных, поскольку к процессу добавляется дополнительный этап – превращение неструктурированного текста в структурированный текстовый массив, чтобы впоследствии к нему можно было применить стандартные методы data mining.

Концепция довольно прозрачна – но для чего text mining нужен в прикладном смысле? Среди самых распространенных областей применения направления:

* пополнение онтологических баз знаний – всеобъемлющих и детальных формализаций в виде концептуальной схемы;
* вопросно-ответные системы и фактографический поиск – еще один шаг на пути к созданию «разумного» Интернета;
* определение семантической (смысловой) близости текстов – в целях анализа данных и обработки естественного языка, в том числе применимой в разработке искусственного интеллекта и машинном переводе.

**Контент-анализ**

**качественный**

***•*** основан на самом факте присутствия или отсутствия в тексте одной или нескольких характеристик содержания

**количественный**

**•** частота появления в документах определенных характеристик содержания (понятий, феноменов)

**74. Стратегии поиска информации**

Стратегия поиска информации в базах данных (БД) – общий план, искусство ведения поиска в БД с использованием закономерностей, присущих этому виду деятельности.

Парадигматический поиск – это дробление заглавного дескриптора на составляющие и поиск по этим составляющим. Например, по запросу «Нужны материалы о поэтах Серебряного века» мы обнаружим в записях помимо заглавного дескриптора «Поэты Серебряного века» его составляющие: Волошин, Гумилев, Пастернак, Ахматова и т.д. Поиск по этим фамилиям драматически увеличит количество найденных релевантных запросу записей.

Синтагматический поиск – поиск с использованием параллельного дескриптора. Например, по запросу «Нужны материалы об угро-финских народах» мы обнаружим мало записей с дескриптором «Угро-финны», а в тех, которые мы обнаружим, будет параллельный дескриптор «Финно-угры». Поиск с использованием этого дескриптора будет гораздо эффективнее. По этому запросу необходимо также использовать другие стратегии поиска: закон Мерфи (на тот случай, если дефис в слове отсутствует, например, с использованием маскирующего оператора «финн$угр\*»); и парадигматический поиск (раздробить понятие на составляющие: марийцы, удмурты, мордва и т.д.).

Поиск узко-специфической информации используется тогда, когда другие стратегии не приносят желаемого эффекта. Эта стратегия обратна стратегии Парадигматический поиск. Заключается она в расширении ключевого понятия. В этом случае мы можем найти источник, в котором искомая информация может содержаться в качестве составной части (главы, доклада, части и т.д.). Например, поиск по запросу «Нужны материалы о главной героине романа Фаулза «Женщина французского лейтенанта» не очень эффективен. Однако, если расширить ключевое понятие до поиска всех критических материалов на творчество Фаулза, то можно найти несколько дополнительных релевантных записей (поисковое предписание “Фаулз\* ИЛИ Fowles\*”). Такой записью может быть книга о женских образах в творчестве Фаулза. В этой книге наверняка имеется релевантная информация.

**75. Полнотекстовый поиск**

Полнотекстовый поиск — автоматизированный поиск документов, при котором поиск ведётся не по именам документов, а по их содержимому, всему или существенной части.

Полнотекстовый поиск – это поиск слов или фраз в текстовых данных. Обычно такой вид поиска используется для поиска текста в большом объёме данных, например, таблица с миллионом и более строк, так как он значительно быстрей обычного поиска.

Полнотекстовый поиск подразумевает создание специального индекса (он отличается от обычных индексов) текстовых данных, который представляет собой некий словарь слов, которые встречаются в этих данных.

С помощью полнотекстового поиска можно реализовать своего рода поисковую систему документов (т.е. строк), по словам или фразам в базе данных своего предприятия. Так как помимо своей быстрой работы он обладает еще и возможностью ранжировать найденные документы, т.е. выставлять ранг каждой найденной строке, другими словами, можно найти самые релевантные записи, т.е. самые подходящие под Ваш запрос.

**76. Поиск по метаданным**

Метаданные - информация о другой информации, или данные, относящиеся к дополнительной информации о содержимом или объекте. Метаданные раскрывают сведения о признаках и свойствах, характеризующих какие-либо сущности, позволяющие автоматически искать и управлять ими в больших информационных потоках.

Поиск по метаданным - это поиск по неким атрибутам документа, поддерживаемым системой - название документа, дата создания, размер, автор и т.д.

**77. Оценка эффективности поиска**

При оценке эффективности информационного поиска традиционно используется ряд коэффициентов качества, определяющих эффективность ИПС, и её сравнение с существующими аналогами. К **критериям** оценки эффективности поиска отнесем такие понятия, как **полнота поиска, точность поиска, уровень потери информации, уровень поискового шума, нормализованные коэффициенты точности и полноты**.

Рассматриваемые характеристики информационного поиска при проведении процедуры поиска в статичных массивах небольших размеров приводят к адекватным результатам. Но существует некоторая ограниченность применения данных характеристик при рассмотрении в качестве среды поиска массивов гипертекстовой информации больших размеров, что обусловлено:

1) существующим количеством документов, что вызывает погрешность в определении коэффициента полноты вследствие использования в расчетах формальной релевантности;

2) значительным числом существующих релевантных документов, размещенных в глобальной сети, что, в свою очередь, не требует высокого значения коэффициента полноты, и данный коэффициент перестает являться адекватной характеристикой эффективности информационного поиска;

3) нечеткостью понятия релевантности в связи с возможностью наличия ситуаций либо частичной релевантности документа, либо релевантности запроса всему документу в целом и нерелевантность части документа, либо релевантность запросу совокупности нескольких документов.

**78. Релевантность**

Релевантность — устанавливаемое при информационном поиске соответствие

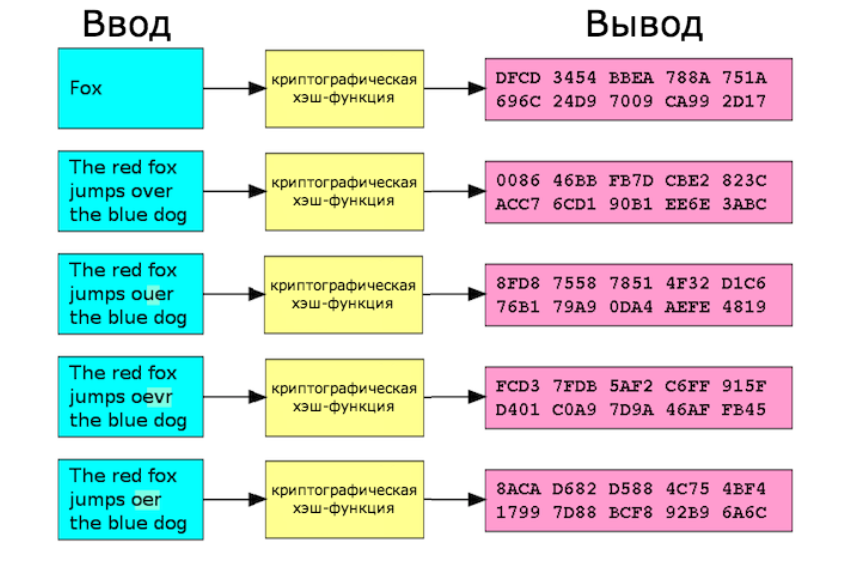
содержания документа информационному запросу или поискового образа

документа поисковому предписанию.

**79. Хэширование**

Если коротко, то криптографическая **хеш-функция,** чаще называемая просто хешем, — это математический алгоритм, преобразовывающий произвольный массив данных в состоящую из букв и цифр строку фиксированной длины. Причем при условии использования того же типа хеша длина эта будет оставаться неизменной, вне зависимости от объема вводных данных. Криптостойкой хеш-функция может быть только в том случае, если выполняются главные требования: стойкость к восстановлению хешируемых данных и стойкость к коллизиям, то есть образованию из двух разных массивов данных двух одинаковых значений хеша. Интересно, что под данные требования формально не подпадает ни один из существующих алгоритмов, поскольку нахождение обратного хешу значения — вопрос лишь вычислительных мощностей. По факту же в случае с некоторыми особо продвинутыми алгоритмами этот процесс может занимать чудовищно много времени.

Хеш-функция (англ. hash function от hash — «превращать в фарш», «мешанина»), или функция свёртки — функция, осуществляющая преобразование массива входных данных произвольной длины в выходную битовую строку установленной длины, выполняемое определённым алгоритмом. Преобразование, производимое хеш-функцией, называется хешированием. Исходные данные называются входным массивом, «ключом» или «сообщением». Результат преобразования называется «хешем», «хеш-кодом», «хеш-суммой», «сводкой сообщения».



Хеш-функции применяются в следующих случаях:

* при построении ассоциативных массивов;
* при поиске дубликатов в сериях наборов данных;
* при построении уникальных идентификаторов для наборов данных;
* при вычислении контрольных сумм от данных (сигнала) для последующего обнаружения в них ошибок (возникших случайно или внесённых намеренно), возникающих при хранении и/или передаче данных;
* при сохранении паролей в системах защиты в виде хеш-кода (для восстановления пароля по хеш-коду требуется функция, являющаяся обратной по отношению к использованной хеш-функции);
* при выработке электронной подписи (на практике часто подписывается не само сообщение, а его «хеш-образ»);
* и др.

В общем случае (согласно принципу Дирихле) нет однозначного соответствия между хеш-кодом и исходными данными. Возвращаемые хеш-функцией значения менее разнообразны, чем значения входного массива. Случай, при котором хеш-функция преобразует более чем один массив входных данных в одинаковые сводки, называется «коллизией». Вероятность возникновения коллизий используется для оценки качества хеш-функций.

Существует множество алгоритмов хеширования, отличающихся различными свойствами. Примеры свойств:

* разрядность;
* вычислительная сложность;
* криптостойкость.

Выбор той или иной хеш-функции определяется спецификой решаемой задачи.

**80. Задачи Web Mining**

Необходимость автоматического анализа информации из интернета вызвана высокой доступностью огромного количества постоянно пополняющейся информации, а также растущей популярностью веб-услуг среди всех категорий пользователей. Развитие Веба в глобальную информационную инфраструктуру позволило обычным пользователям быть не только потребителями информации, но ее создателями и распространителями. В этой связи для эффективного решения задач поиска, структурирования и анализа в основном хаотично организованной информации в сети предназначено новое направление в методологии анализа данных - Web Mining.

Web Mining — это использование методов интеллектуального анализа данных для автоматического обнаружения веб-документов и сервисов, извлечения информации из веб-ресурсов и выявления общих закономерностей в Интернете.

Web Mining развивается на пересечении таких дисциплин как обнаружение знаний в базах данных, эффективный поиск информации, искусственный интеллект, машинное обучение и обработка естественных языков.

**Категории:**

1. Анализ использования веб-ресурсов (Web Usage Mining).
2. Извлечение веб-структур (Web Structure Mining).
3. Извлечение веб-контента (Web Content Mining).

**Задачи Web Mining**

* **Поиск информации**

Для нахождения необходимой информации пользователи обычно пользуются поисковыми ресурсами. При этом часто используются простые запросы по ключевым словам. Результатом выполнения запроса является список страниц, отсортированный по некому индексу релевантности, описывающему степень совпадения результата с запросом. Однако существующие поисковые механизмы обладают недостатками. Основным из них является низкая точность результата, вызванная недостаточным учетом семантических связей и контекста найденных в тексте выражений. Индексация интересующих сегментов сети с использованием интеллектуального анализа данных, применяющего алгоритмы математической лингвистики и обработки естественных языков, является перспективным направлением Web Mining в области поиска информации.

* **Анализ структуры сегмента сети**

Этот метод заключается в анализе структуры ссылок между различными веб-страницами, внутренними и внешними сайтами в выделенном сетевом сегменте. Появление этого метода было вызвано необходимостью решения задач, возникающих при анализе социальных сетей или специфических областей человеческой деятельности или знаний, например, в анализе цитирования авторов. Результатом такого анализа может служить выявленный набор специфичных страниц следующих типов:

1. хабы - из такой страницы ссылки идут на наиболее значимые ресурсы в данной области знаний или на "знакомства" с наиболее значимыми пользователями социальной сети;
2. авторитеты - страницы, на которые ссылаются большое количеством авторов по данной тематике или пользователи социальной сети, к "дружбе" с которыми стремится большое количество пользователей.

Топология структуры ссылок представляется в виде направленного графа с помеченными узлами в соответствии с их функциональной классификацией и дугами с весами, описывающими, например, частоты переходов по ссылке. Для моделирования топологии веб-ссылок используется несколько алгоритмов, например HITS (Jon M. Kleinberg, "Authoritative sources in hyperlink environment").

* **Выявление знаний из веб-ресурсов**

Эта задача пересекается с уже описанной проблемой поиска информации. Только здесь у исследователя уже имеется набор веб-страниц, полученных в результате запроса. Далее требуется произвести их обработку с точки зрения автоматической классификации, составления оглавлений, выявления ключевых слов и общих тем. Выявленные знания могут представляться в виде деревьев, описывающих структуры документов или в виде логических и семантических выражений. Решение части этих проблем предлагает Text Mining - технология автоматического извлечения знаний в больших объемах текстового материала, основанная на сочетании лингвистических, семантических, статистических и машинных обучающихся методик (http://www.megaputer.ru/text\_mining.php, Soumen Chakrabarti "Data mining for hypertext", Helena Ahonen-Myka, "Finding co-occuring text phrases by combining sequence and frequent set discovery")

* **Персонализация информации**

Персонализации веб-пространства - задача по созданию веб-систем, адаптирующих свои возможности (навигация, контент, баннеры и другие рекламные предложения) под пользователя на основании собранной и проанализированной информации о пользовательских предпочтениях.

Классическим примером может являться ресурс http://www.amazon.com/, на котором один раз заказав дорогую книгу в твердом переплете, пользователь начинает регулярно получать предложения о покупке подарочных изданий по схожей тематике. Другой пример - на основании анализа корзин заказов пользователя ему предлагаются **товары,** которые он никогда не заказывал, но которые входят в корзины других покупателей, схожих с ним по транзакционному поведению.

Для анализа информации о пользователе следует в наименьшей степени использовать декларируемую о себе информацию, а скорее основываться на стойких шаблонах его "поведения" в сети - последовательности кликов внутри ресурса, переходах на другие под-ресурсы, периодах сетевой активности, осуществляемых покупках и т.д. См. B. Masand, Redwood, "Web Usage Analysis and User Profiling", Miha Gr?ar, "User profiling: Web usage mining".

* **Поиск шаблонов в поведении пользователей**

Эта задача связана с предыдущей, но ее целью является не адаптация ресурса к предпочтениям индивидуальных пользователей, а поиск закономерностей в шаблонах взаимодействия пользователя с веб-ресурсом с целью прогнозирования его последующих действий. Анализируемые действия пользователей могут включать не только переходы по ссылкам, но и отправку форм, прокрутку страниц, добавление в избранные страницы и т.д. Найденные шаблоны используются в дальнейшем для оптимизации структуры сайта, изучения целевой аудитории и для прямого маркетинга.

Разработано множество подходов к решению задачи по выявлению знаний из шаблонов навигации пользователей (Jose Borges и Mark Levene "Data Mining of User Navigation Patterns", A. G. Buechner "Navigation Pattern Discovery from Internet Data").

С точки зрения применения алгоритмов интеллектуального анализа данных при поиске шаблонов пользовательского поведения чаще всего используются следующие методики:

1. Кластеризация - поиск групп похожих посетителей, сайтов, страниц и т.д.
2. Ассоциации - поиск совместно запрашиваемых страниц, заказываемых товаров.
3. Анализ последовательностей - поиск последовательностей действий. Наиболее часто применяется вариант алгоритма apriori, разработанного для анализа частых наборов, но модифицированного для выявления частых фрагментов последовательностей и переходов.

Особенно интересен подход кластеризации последовательностей - поиск групп пользователей со схожими последовательностями действий. На первом этапе в этом подходе выделяются последовательности классифицированных действий пользователя, например, в рамках одной сессии. Затем подсчитываются частоты переходов между различными действиями для составления Марковской цепи заданного порядка. На заключительном этапе полученные Марковские цепи кластеризуются для выявления групп с похожими частотами переходов. Для прогнозирования следующего действия пользователя сначала на основании истории его действий в рамках сессии определяется группа, к которой он принадлежит с наибольшей вероятностью. Затем определяется действие, которое выполняется с наибольшей вероятностью в этой группе с учетом последних действий данного пользователя. Для реализации такого анализа можно, например, использовать алгоритм Microsoft Sequential Clustering, входящий в Microsoft Analysis Services 2005/2008. Недостатком алгоритма Microsoft является то, что до настоящего времени реализован алгоритм, использующий Марковские цепи только первого порядка.

В качестве примера применения метода анализа последовательности действий можно привести задачу по оптимизации рубрикации одного книжного интернет-магазина, проведенную компанией spellabs. Была выявлена группа, состоящая из пользователей, переходящих долгими путями по ссылкам на книги из разных рубрик и заказывающих в конечном итоге "изотерическую" литературу , до этого отдельно не выделенную в рубрику. Так была выявлена неучтенная целевая аудитория и оптимизирована структура сайта.

**Вывод**

Web Mining является новым перспективным направлением анализа интернет-ресурсов для оптимизации структуры веб-сайтов, получения знаний о посетителях сайта, описания социальных сетей и сообществ, а также для автоматического поиска и структуризации информации из интернета.

**81. SEO**

Поисковая оптимизация (англ. search engine optimization, SEO) — комплекс мероприятий по внутренней и внешней оптимизации для поднятия позиций сайта в результатах выдачи поисковых систем по определённым запросам пользователей, с целью увеличения сетевого трафика (для информационных ресурсов) и потенциальных клиентов (для коммерческих ресурсов) и последующей монетизации (получение дохода) этого трафика. SEO может быть ориентировано на различные виды поиска, включая поиск информации, товаров, услуг, изображений, видеороликов, новостей и специфические отраслевые поисковые системы.

Обычно чем выше позиция сайта в результатах поиска, тем больше заинтересованных посетителей переходит на него с поисковых систем. При анализе эффективности поисковой оптимизации оценивается стоимость целевого посетителя с учётом времени вывода сайта на указанные позиции и конверсии сайта.

***Основные направления работы***

Поисковые системы учитывают множество внутренних и внешних параметров сайта при вычислении его релевантности (степени соответствия введённому запросу):

1. *плотность ключевых слов* (сложные алгоритмы современных поисковых систем позволяют производить семантический анализ текста, чтобы отсеять поисковый спам, в котором ключевое слово встречается слишком часто (терм. сленг «тошнота»);
2. *индекс цитирования* («ИЦ») и *тематический индекс цитирования* («ТИЦ») зависят от количества и авторитетности веб-ресурсов, ссылающихся на данный сайт; многими поисковиками не учитываются взаимные ссылки (друг на друга). Способ наращивания числа сайтов-доноров ссылающихся на продвигаемый сайт, называется Линкбилдинг. Примером метрики индекса цитирования являлся PageRank;
3. *водность текста* — показатель, определяющий наличие малозначимых слов, которые не несут никакой полезной информации и служат для разбавления текста (стоп-слова);
4. *поведенческие факторы* (внутренние) — ряд всевозможных действий пользователей, которые они могут произвести на сайте: вход, общее время проведённое пользователем на сайте, количество сессий одного пользователя на сайте, просмотр страниц, количество просмотренных пользователем страниц, возврат пользователя на сайт, клики на ссылки в тексте, переходы по ссылкам в меню;
5. *поведенческие факторы* (внешние) — основной внешний показатель качества поведения пользователя при взаимодействии с сайтом является отказ от дальнейшего поиска по ключевой фразе в поисковой системе;
6. *индекс качества сайта* («ИКС») — это показатель того, насколько полезен конкретный сайт для пользователей с точки зрения Яндекса. ИКС введён в 2018 году взамен тематического индекса цитирования (ТИЦ), учитывавшего тематику ссылающихся сайтов.
7. *скорость загрузки сайта* — показатель скорости загрузки сайта. Используется несколько параметров для характеристики скорости загрузки сайта — загрузка до появления первого контента, загрузка первого контент до взаимодействия, скорость ответа сервера на запрос, длина html кода. Общепринятым стандартом скорости загрузки сайта принято считать сервис Google PageSpeed.

**82. Инструментальные средства анализа данных**

Excel

Excel – это не только создание формы, сводные таблицы и VBA. Система настолько велика, что ни один инструмент аналитики не может превзойти ее, гарантируя, что люди могут анализировать данные в соответствии с их потребностями.

Тем не менее, многие люди равнодушны к Excel в виде инструмента для такой задачи. Мы считаем, что Excel – это универсальный игрок. Он справится как с небольшими данными, так и обработкой бигдаты (используя плагин).

Основываясь на мощных возможностях Excel и его пользовательском охвате, можно сказать, что это незаменимый инструмент. Если тебе нужно изучить анализ данных, то Excel, безусловно, является отличным выбором.

Tableau

Основное направление Tableau – сводные таблицы и диаграммы Excel. Судя по отзывам на рынке и истории развития, этот продукт лучше подходит для визуализации. Крутые графики, дизайн, цвет и пользовательский интерфейс дают ощущение свежести и легкости.

Не так давно разработчики добавили функцию очистки данных и более интеллектуальные функции анализа, т. е. продукт постоянно поддерживается и развивается, что немаловажно.

Orange

Orange – это опенсорсный проект, содержащий набор инструментов для визуализации, обработки и анализа данных. Огромный пакет инструментов поможет тебе создать интерактивный рабочий процесс для анализа и визуализации данных с помощью точечных диаграмм, гистограмм, деревьев, дендрограмм, сетей и тепловых карт. Подойдет как для профессиональных аналитиков, так и для обычных юзеров.

**Сокращения:**

**ИТ** - информационная технология

**BI** -Business Intelligence

**DS** - Data Science

**ИАД** - интеллектуальный анализ данных

**EIS** - Enterprise Information Systems

**OLTP** - online transaction processing

**OLAP** - online analytical processing

**ИАС** - [Информационно-аналитическая система](https://grant.rscf.ru/)

**DWH** - data warehousing (хранилище данных, ХД)

**KDD** -Knowledge Discovery in Databases



**DB** - data base

**DM** - Data Mining

**SOA** -service-oriented architecture ( Сервисориентированная архитектура)

**ETL** - Extraction, Transformation, Load

!!!

**Извлечение, Преобразование, Загрузка — ETL.**

**Извлечение, Загрузка, Преобразование — E-LT.**

!!!

**CDM** - [Common Data Model](https://powerplatform.microsoft.com/ru-ru/common-data-model/) ( общая модель данных)

**TM** -Text Mining

**TF-IDF** :

**ТF** - term frequency(частотность термина, которая измеряет, насколько часто термин встречается в документе)

**IDF** — inverse document frequency (это обратная частотность документов)

**ИАТ** -Интеллектуальный анализ текста (?)

**ERP** - **Enterprise Resource Planning**, планирование ресурсов предприятия

**CRM -** Customer Relationship Management, (Управление отношениями с клиентами)

XML - Extensible Markup Language

JSON -Java Script Object Notation

**SEO -**search engine optimization