

Big Data and Data Science

Логинов Сергей

НФИМд-01-22

link: <https://stepik.org/lesson/584152/step/1?unit=578918>

Образовательная организация:

Русская школа программирования

Трудоемость:

26-28 часов

Блок SQL

Все примеры реализуются в БД Oracle.

В данном блоке изучаются

- Подзапросы в операторах SELECT, FROM, WHERE и их применение для решения практических задач
- Дополнительные операторы - ANY, ALL, IN, их использование в подзапросах
- СТЕ - табличные выражения, помогающие эффективнее работать при большом количестве подзапросов.
- Различные виды JOIN - INNER, LEFT, RIGHT
- Альтернативные виды объединения - UNION, UNION ALL, INTERSECT, MINUS
- Оконные функции - агрегатные и ранжирующие, смещения и аналитические
- Представления

Блок Big Data

В данном блоке представлены азы архитектуры MapReduce и файловой системы HDFS, а также Apache Hadoop.

Конспект первого видео

Большие данные - структурированные и неструктурированные данные в больших объемах.

Структурированные данные - имеют четкий формат, всегда имеют однозначное обращение к ним. Например, таблицы в БД.

Неструктурированные данные - более гибкий формат, в котором информации не представлена в однозначном виде. Например, тексты, картинки, аудио и т.д.

Суть больших данных выражена в модели 3V(VVV):

- Объем - обработка больших массивов информации, неформально от 1 Гбайт
- Скорость - быстрая генерация информации
- Разнообразие - данные в различных форматах поступают из разных источников и имеют разные сущности

Существует вертикальное и горизонтальное масштабирование

Вертикальное - увеличение и наращивание производительности

Горизонтальное - увеличение количества вычислительных центров

Технология MapReduce - разработка Google. Краткое введение.

В ее основе лежат процедуры функционального программирования:

- map - применение заданной функции к каждому элементу списка
- reduce - объединение результатов работы map

На основе данного подхода разработано множество продуктов для обработки больших данных:

- Apache Hadoop
- Apache CouchDB
- MongoDB

В действительности данная модель состоит из последовательной комбинации трех функций:

1. map - обрабатывает большое количество входных данных.

Главный узел (мастер) получает список данных, делит его на части для рабочих узлов.

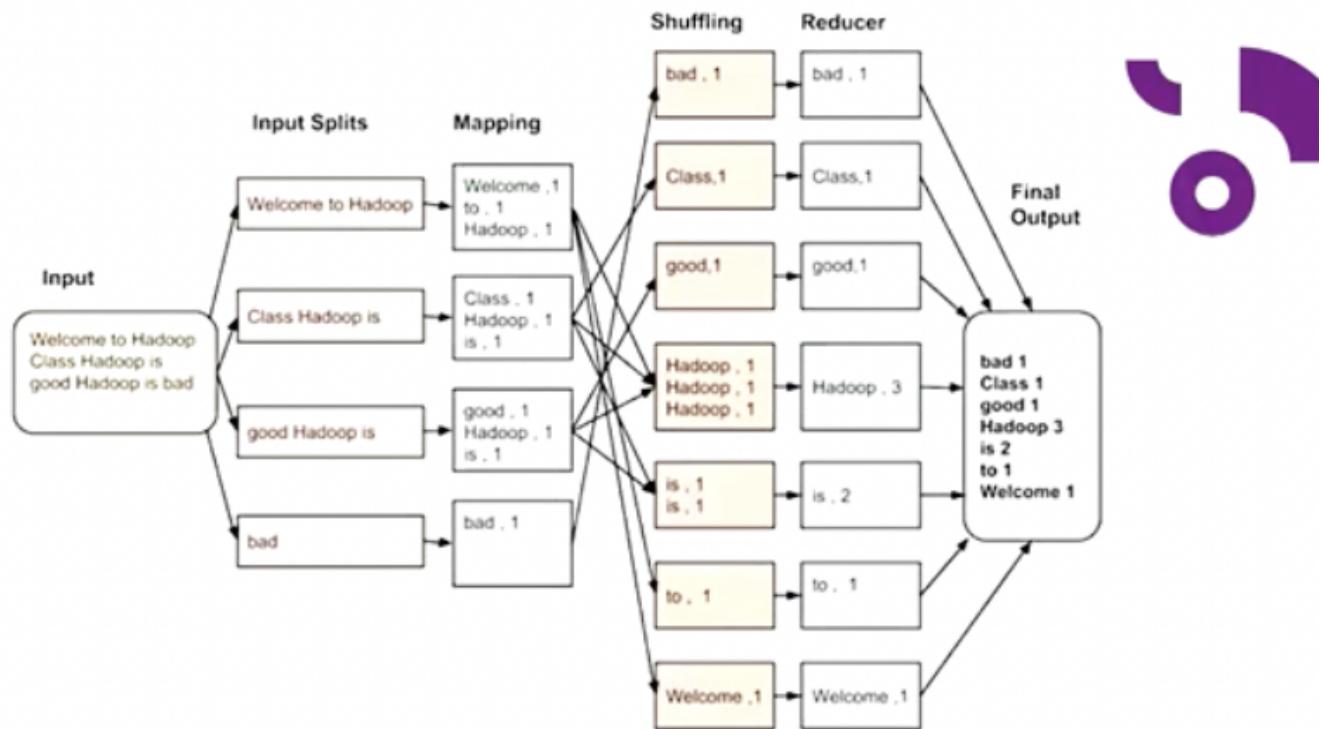
Далее рабочие узлы применяют функцию map на части данных и записывают результат в виде "ключ-значение"

2. shuffle - перераспределяет данные на основе ключей, созданных функцией map, данные одного ключа находятся в отдельном узле

3. reduce - параллельная обработка данных по порядку следования ключей и объединение результатов в мастере.

Мастер передает свободным узлам промежуточные результаты выполнения следующего шага. Результат = решение поставленной задачи

Примерная схема работы модели:



Распределенная файловая система HDFS

Данная файловая система, используемая в Hadoop, предназначена для хранения файлов больших размеров, которые блоками распределены на

разных узлах вычислительного кластера.

Файловая система имеет архитектуру master/slave. Она состоит из (утрировано):

NameNode - сервер, который управляет пространством имен файловой системы и регулирует клиентский доступ к файлам. Другие функции:

- Выполняет операции на открытие, закрытие, переименовывание файлов и каталогов
- Определяет отображение блоков в DataNodes
- Дает команды DataNodes на создание, удаление и репликацию блоков

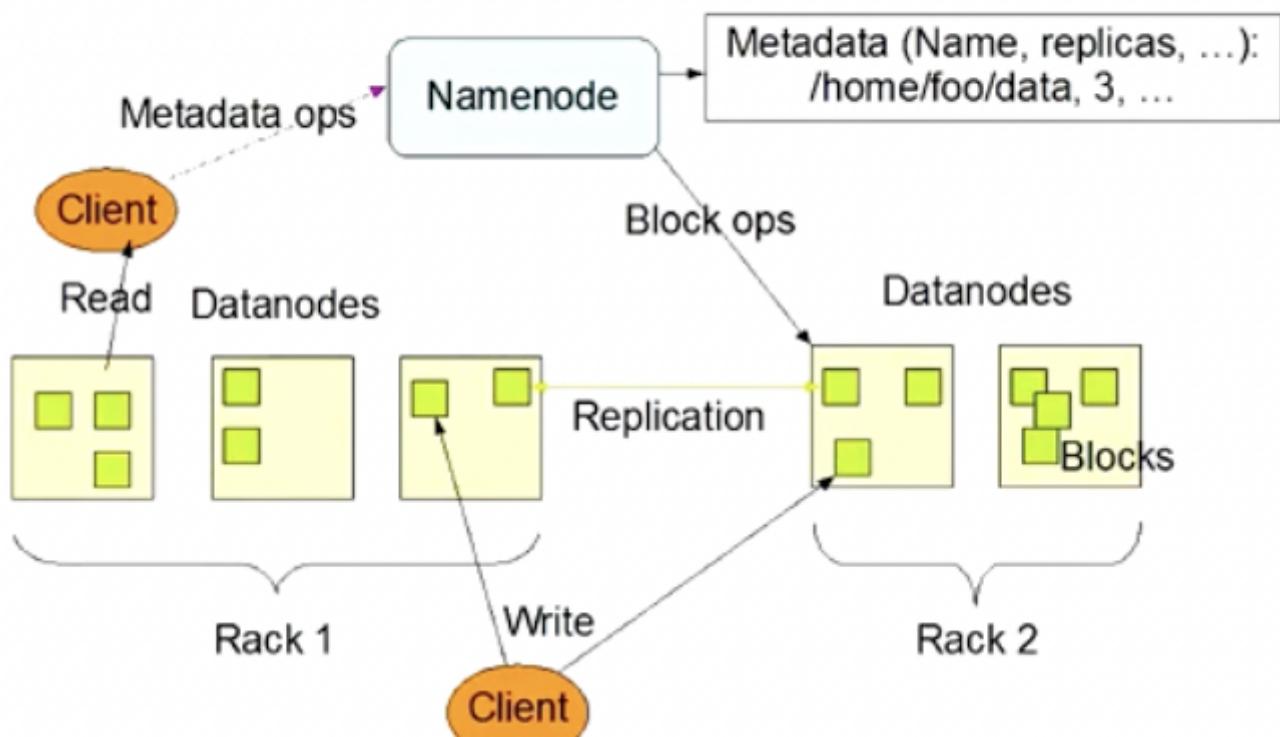
DataNodes - управляют хранилищем данных, хранят внутри блоки данных.

Другие функции:

- Обычно их несколько и по одному на каждом узле кластера
 - Отвечает за обслуживание запросов чтение/запись с клиентской файловой системы
- Client** - пользователь и приложение, взаимодействующий через API с файловой системой

Схема:

HDFS Architecture



Что такое Hadoop ?

Apache Hadoop - набор библиотек, утилит и фреймворк для работы с большим объемом данных.

Его основные модули:

- Hadoop Common - набор утилит и библиотек, используемых, в том числе, для управления распределенными файлами
- Hadoop MapReduce - платформа для программирования и выполнения MapReduce-вычислений
- YARN - система планирования заданий и управления кластером
- HDFS - распределенная файловая система

Полезные ссылки:

<https://stepik.org/course/150/promo> - Бесплатный курс по Hadoop

<http://hadoop.apache.org/> - Официальный сайт Apache Hadoop

<https://www.bigdataschool.ru/wiki/mapreduce> - Что такое MapReduce

<https://medium.com/edureka/hadoop-tutorial-24c48fbf62f6> - Туториал по экосистеме Hadoop

<https://medium.com/xnewdata/do-you-know-what-is-hadoop-mapreduce-technology-ed3b341401c7> - Серия статей по Hadoop. Очень полезно для более глубокого погружения

Инструменты обработки, анализа и визуализации данных

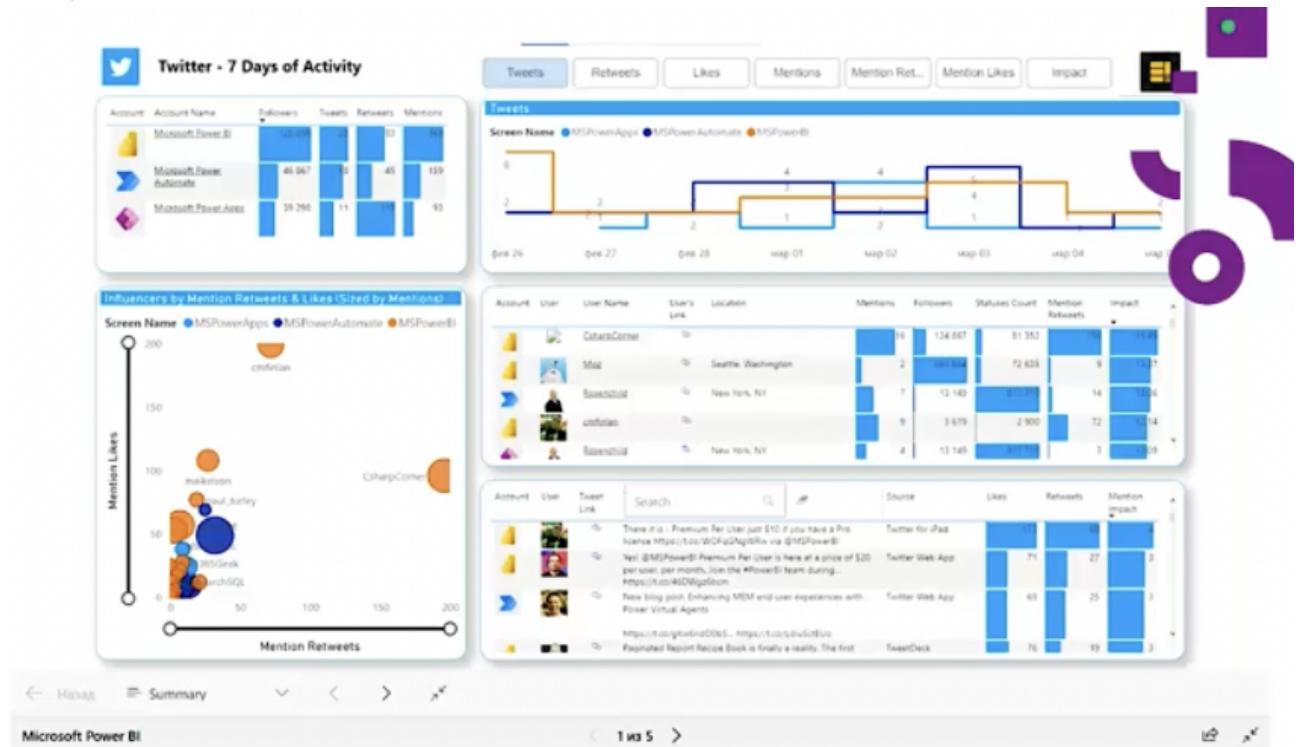
Анализ данных в Power BI

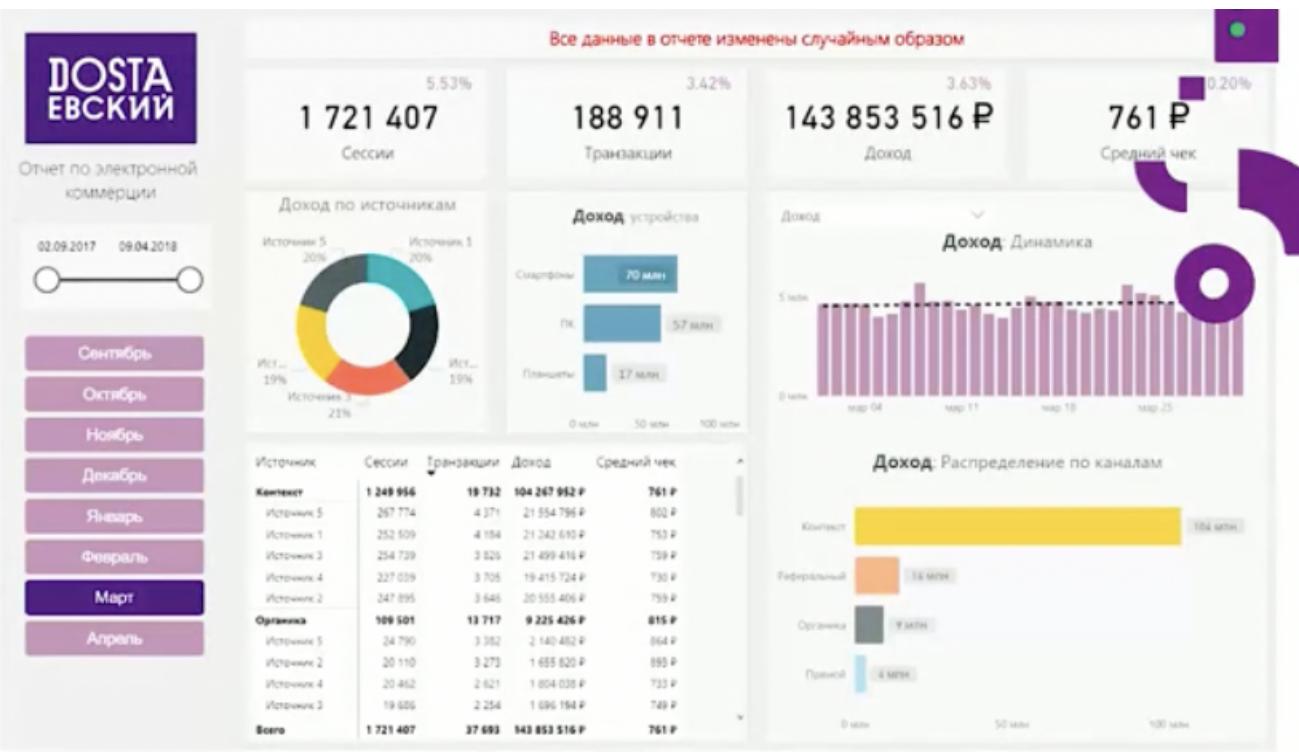
Power BI - набор продуктов от Microsoft для бизнес-аналитики - изменения в компаниях через определение потребностей и рекомендацию решений, которые обеспечивают пользу для стейкхолдеров (заинтересованных сторон).

Назначение Power BI

- Получение данных
- Визуализация
- Предоставление доступа к данным

Результат работы в Power BI - отчет. Его примеры:





Плюсы Power BI:

- Большое комьюнити
- Лидер отрасли
- Встроенные коннекторы
- Простота использования
- Сходство с Excel
- Набор визуализаций

Power BI Desktop

Возможности:

- Подключение к данным
- Преобразование и очистка данных
- Создание визуальных элементов
- Создание отчетов
- Предоставление доступа к отчетам

Компоненты:

- Power Query - извлечение, загрузка, преобразование

- Power Pivot – моделирование
- Power View – визуализация

Power Pivot использует "язык" DAX – Data Analysis eXpressions. С помощью них создаются:

- Меры – вычисляются во время использования в визуализации
- Вычисляемые столбцы – вычисляются при создании

Категории функций DAX:

- Фильтры
- Математические
- Статистические
- Логические
- Даты и времени
- Текстовые
- Информационные

Пример DAX:

Год добавления фильма на Netflix:

```
--date_add_year = year('netflix_data'[date_added])
```

Кол-во лет от создания фильма до
добавления на Netflix:

```
--date_diff_add_release =
'netflix_data'[_date_add_year_]-
'netflix_data'[release_year]
```

Примеры мер:



Среднее кол-во оценивших:

```
!avg_num_votes = AVERAGE(imdb_data[num_votes])
```

Средний рейтинг:

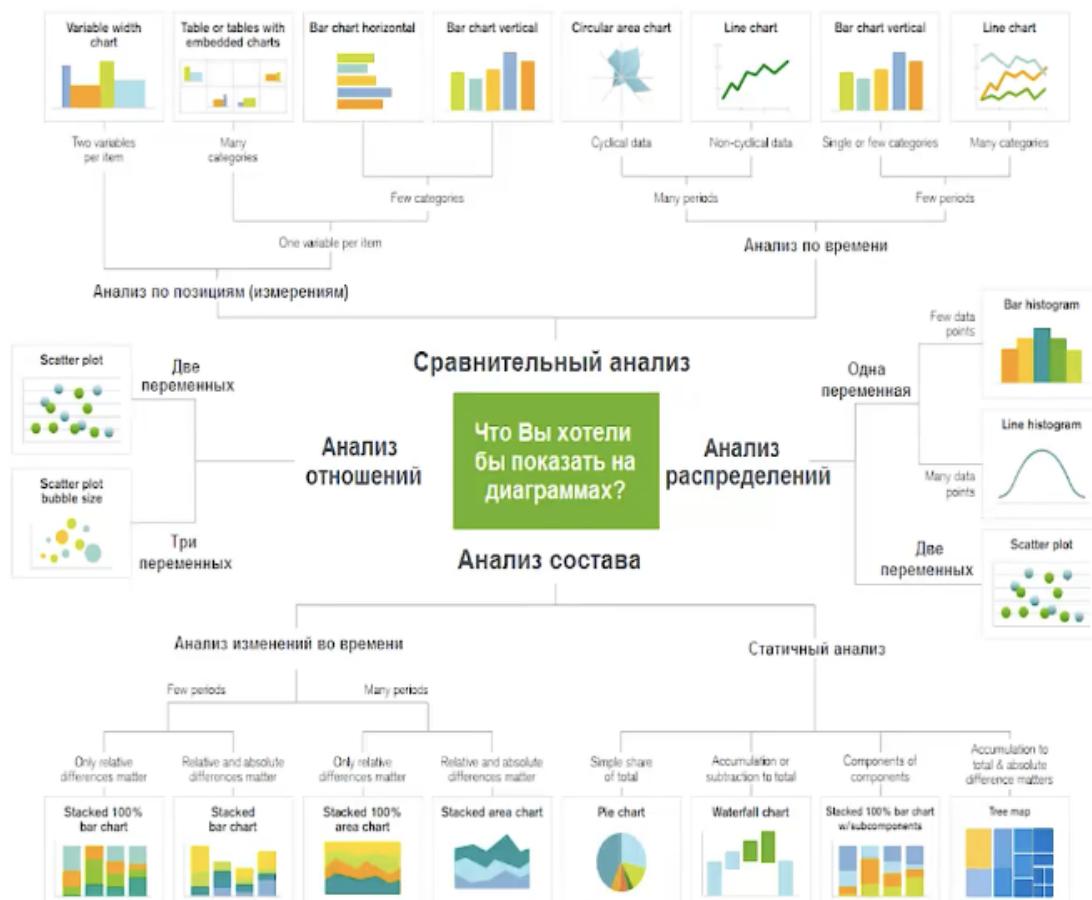
```
!avg_rating = AVERAGE(imdb_data[average_rating])
```

Средний кол-во лет до добавления на Netflix:

```
!avg_year_before_add =  
AVERAGE(netflix_data[_date_diff_add_release])
```



Подходы к выбору визуализации:



Полезные ссылки:

Данные

Netflix: https://github.com/NastyaSNK/pbi_lecture/raw/master/netflix_data.csv

Данные IMDB: https://github.com/NastyaSNK/pbi_lecture/raw/master/imdb_data.csv

Файл с итоговым

результатом: https://github.com/NastyaSNK/pbi_lecture/raw/master/imdb_data.pbix

Любопытные источники на русском:

[Как Microsoft спрятала целый сервер и как его найти / Хабр \(habr.com\)](#) - Статья на Хабре о внутренностях PowerBI

[Документация по Power BI - Power BI | Microsoft Docs](#) - Документация Microsoft

[PowerBIBook.ru — документация PowerBIBook.ru](#) - Адаптированный учебник

[Power BI в Microsoft Learn | Microsoft Docs](#) - Обучение Microsoft

[Блог NeedForData, обучающие статьи по Power BI](#) - Блог о PowerBI и Excel

[Telegram: Contact @PBI_Rus](#) - Телеграм-канал для вопросов и обсуждений по теме Power BI

<https://github.com/needforadata/PublicSharables.pdf> - Краткая методичка по применению визуализаций для Power BI

Инструменты работы с большими данными

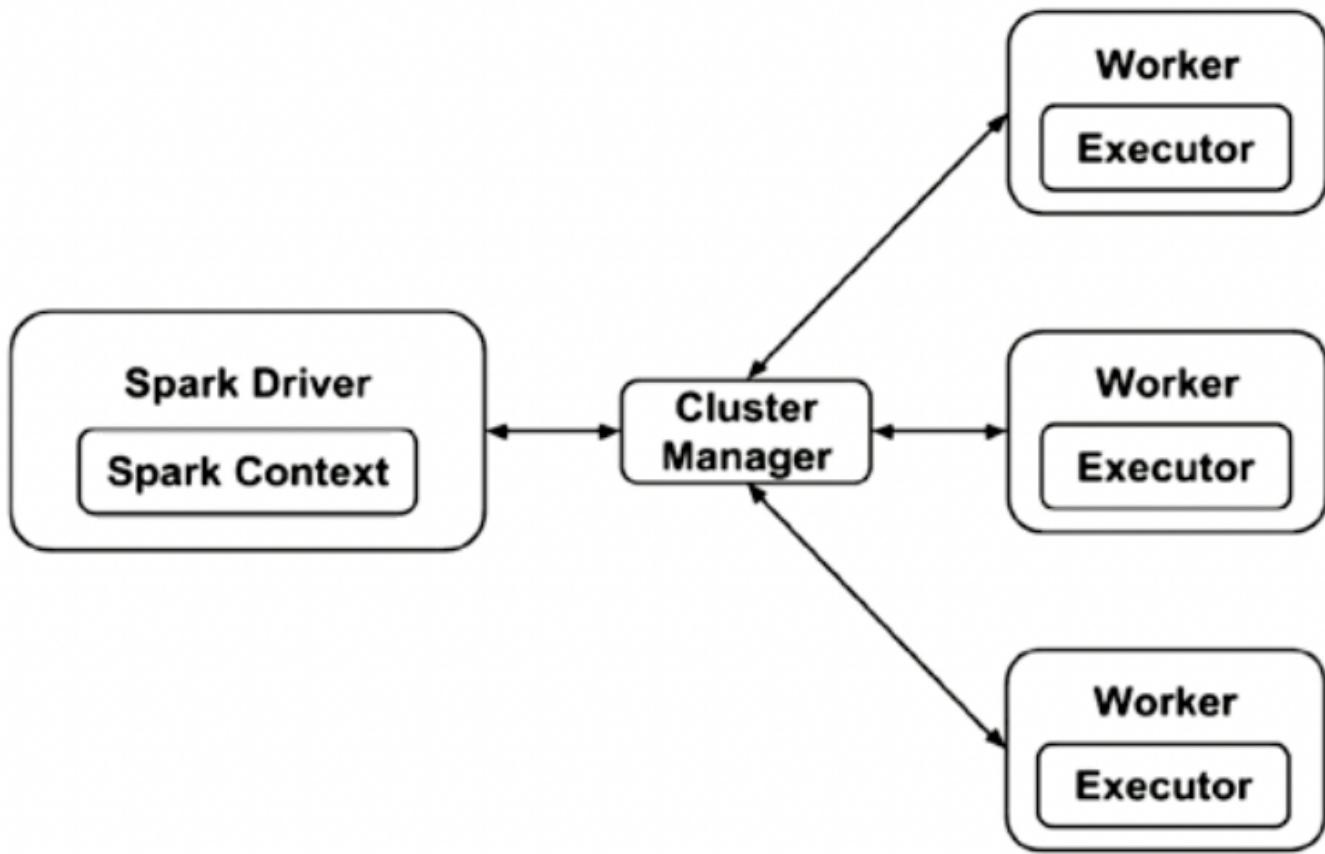
Фреймворк Apache Spark

pandas - работа с большими данным, которые помещаются в оперативную память

Spark - работа с большими данными, которые не помещаются в оперативную память

Особенность Spark - ленивые вычисления. Все манипуляции с данными подразделяются на два общих - трансформации и действия.

Схема работы:



Основой является Driver. Он запускает функцию main и разделяет приложение на задачи.

Задачи передаются исполнителям (executor), каждый из них работает над своей задачей.

Для управления кластером существует Cluster Manager, их существует несколько типов

Основные форматы хранение данных:

		
.xlsx	.csv	.json
Не открывается в текстовых редакторах и может быть зашифрован паролем	Открывается любым текстовым редактором и читаем	Открывается любым текстовым редактором и легко читаем
Потребляет много памяти	Потребляет мало памяти	Потребляет много памяти
Поддерживает вложенность	Не поддерживает вложенность	Поддерживает вложенность
Плохая масштабируемость	Плохая масштабируемость	Хорошая масштабируемость

В Spark используется особенный способ хранения данных: Parquet

- Записать один раз, считывать по потребности
- Загружаются только необходимые данные, а не весь файл
- Позволяет использовать вложенный формат данных
- Хранится по столбцам



Row Storage

Last Name	First Name	E-mail	Phone #	Street Address

Columnar Storage

Last Name	First Name	E-mail	Phone #	Street Address

Системы машинного обучения

Статистика

A/B тесты

Идея:

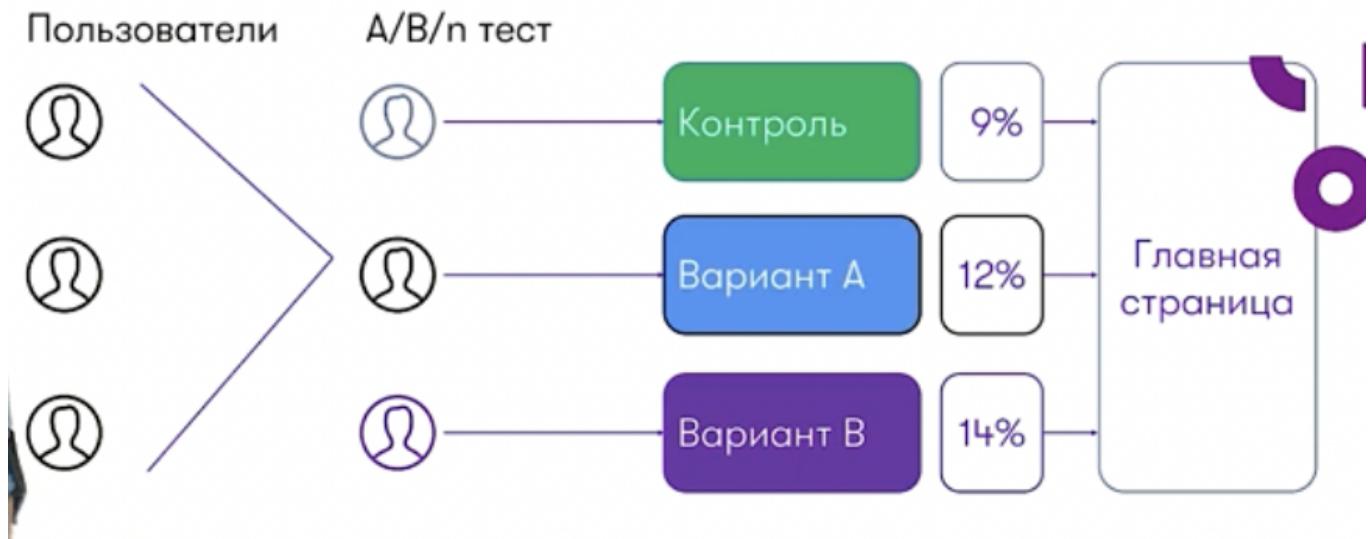


Требования к тестам:

- Понимание метрики/цели

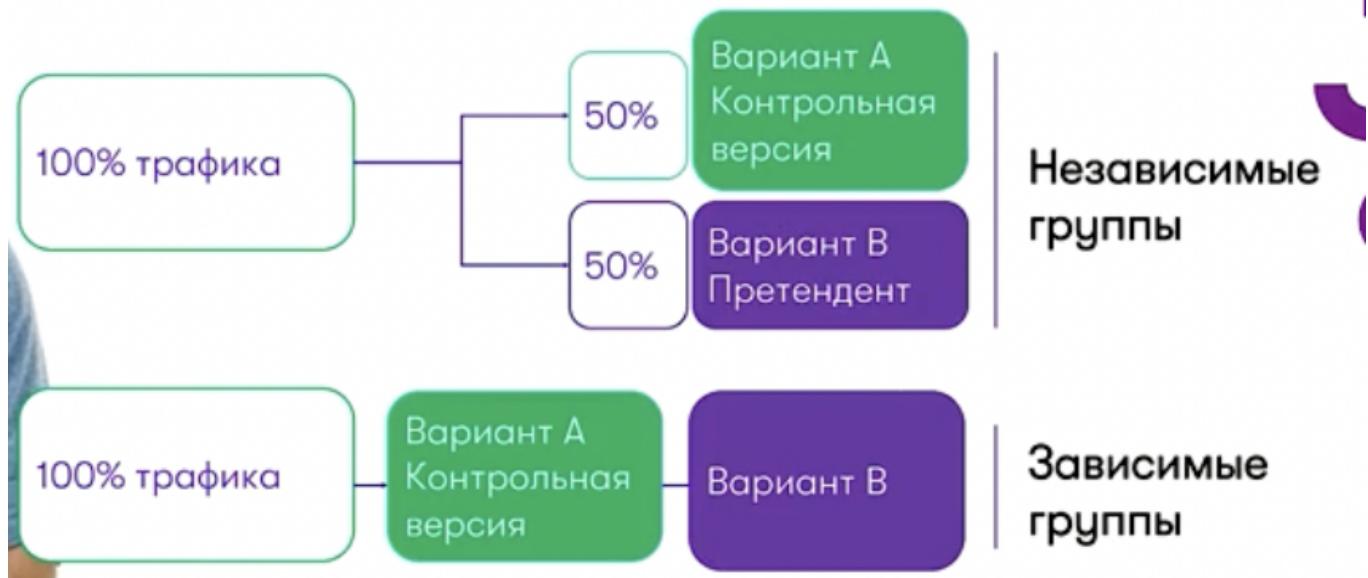
- Одновременность
- Случайность
- Достаточный объем выборки
- Независимость

A/B тестирование не ограничивается делением на 2 группы, существуют Multi A/B тесты:



Если нет деления на группы или группы зависимы, то в таком случае также можно применять тестирование, но измененное:

А/В тесты при зависимых группах

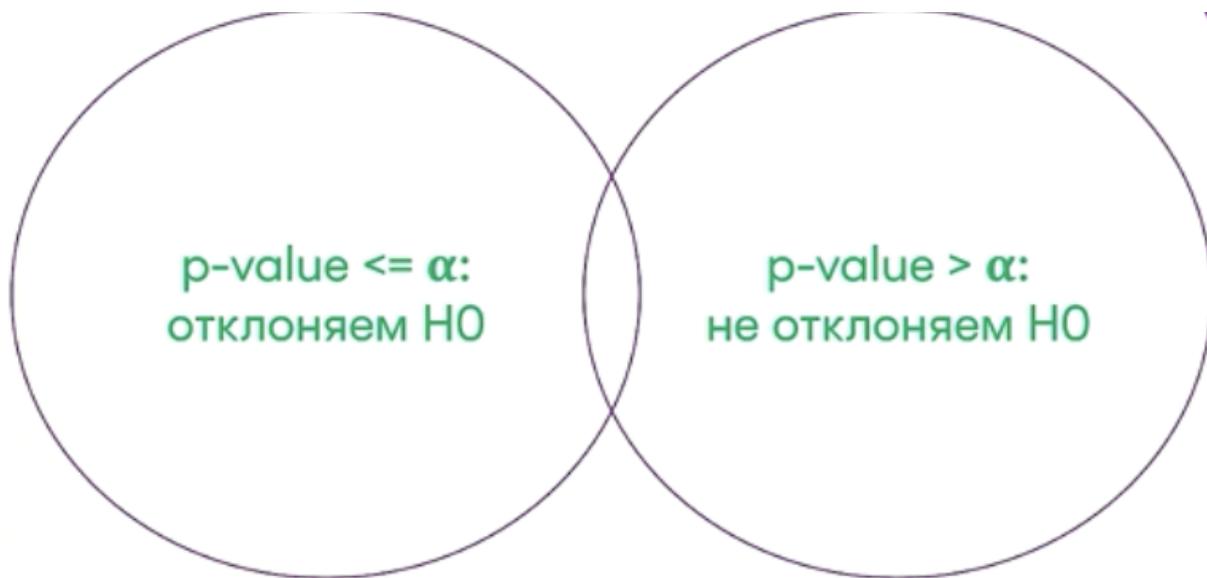


Основная задача тестов - статистическая проверка выгодных и полезных гипотез.

Основные понятия:

p-value - вероятность ошибки при условии, что эффект случайный;
вероятность ошибочно заметить различия между контрольной и тестовой группой

Гипотеза - наше предположение



p - value - это вероятность ошибки,
при условии, что эффект случайный

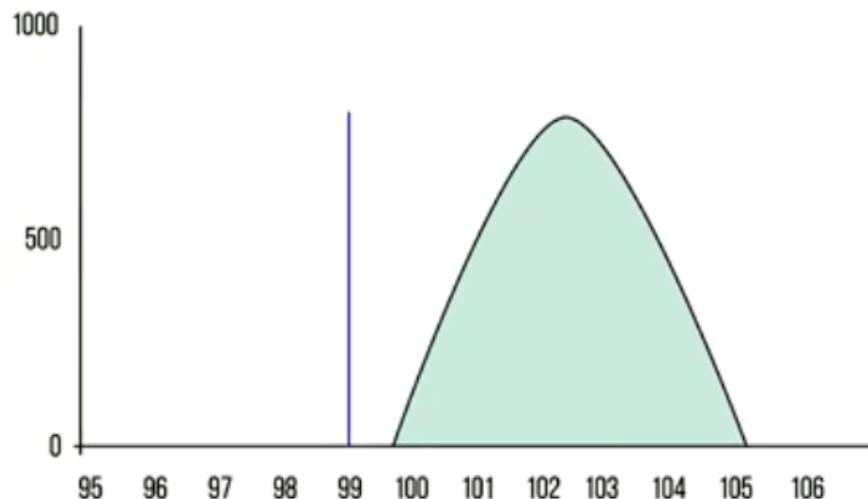
Как принять решение?

Сравнить p-value с альфа (обычно берется 5%)

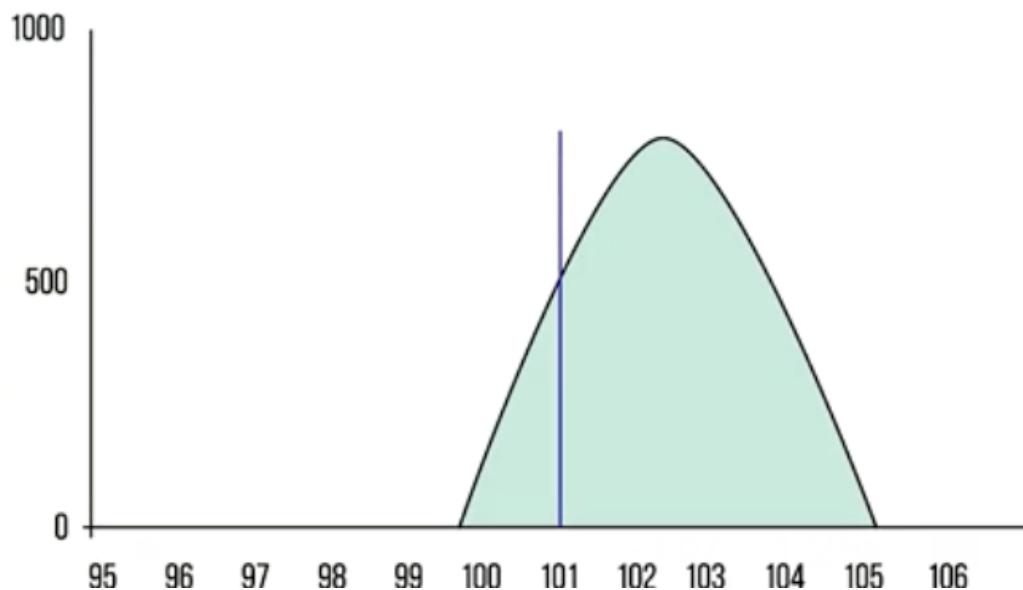
контрольной группы



p-value < 5%



p-value > 5%



p-value получаем с помощью статистических критериев. Основная цель критерия - дать величину возможной ошибки.



Сравнение двух выборок За исключением номинальных данных

Нормальное распределение
(только количественные)

Распределение ненормальное
(количественные или
категориальные)

t-тест Стьюдента

Зависимые/
Независимые

Категориальные

Критерий
Уилкоксона

U-
критерий
Манна-
Уитни

Критерий
Фишера/Пирсона

t-test/Критерий Стьюдента

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

Легко интерпретируемый
критерий.
Мы просто смотрим на
различие **средних**

Критерий Вилкоксона и Манна-Уитни

Используйте для проверки между
двумя выборками (зависимыми или независимыми)

Невозможно отклонить НО:
распределение выборок равны.
Отклонить НО: распределения
выборок не равны

Критерий Вилкоксона для независимых выборок называется критерием
Манна-Уитни

В отличие от критерия Стьюдента, критерий Манна-Уитни не требует
нормального распределения, но является трудно интерпретируемым.

Дисперсионный анализ – для сравнения большого количества групп.
Идея: сравнение дисперсии внутри группы и дисперсии между группами.
Исходя из этого вычисляются оценки того, насколько группы отличаются.
Вычисляется F-значение Фишера.

Машинное обучение в бизнесе

В лекции рассказывается о том, что на данный момент машинное обучение и
нейросети нельзя назвать полноценной научной дисциплиной. Они часто и,
что важно, очень успешно применяются в бизнесе, однако нет
фундаментальных теорий об их работе.

В этой области существует два лагеря - сторонники научного подхода, и сторонники бизнес-подхода. Первые хотят глубоко разобраться в теме и все подробно изучить, вторые же считают, что рабочие конструкции/решения, если они будут полезны, нужно брать и применять, не вдаваясь в детали. Для примера взят google-переводчик. Для перевода с точки зрения науки очень сложно реализовать какие-то модели и алгоритмы. Также сложно описать перевод формулами и логическими выражениями. Заниматься формализацией всех правил разных языков долго и дорого. Поэтому здесь используется рекуррентная нейросеть, которая обучается на десятках миллионов пар вида "слово-перевод". Да, такой вариант переводчика не будет 100% точным, однако его реализация во много раз дешевле, а польза не намного меньше.

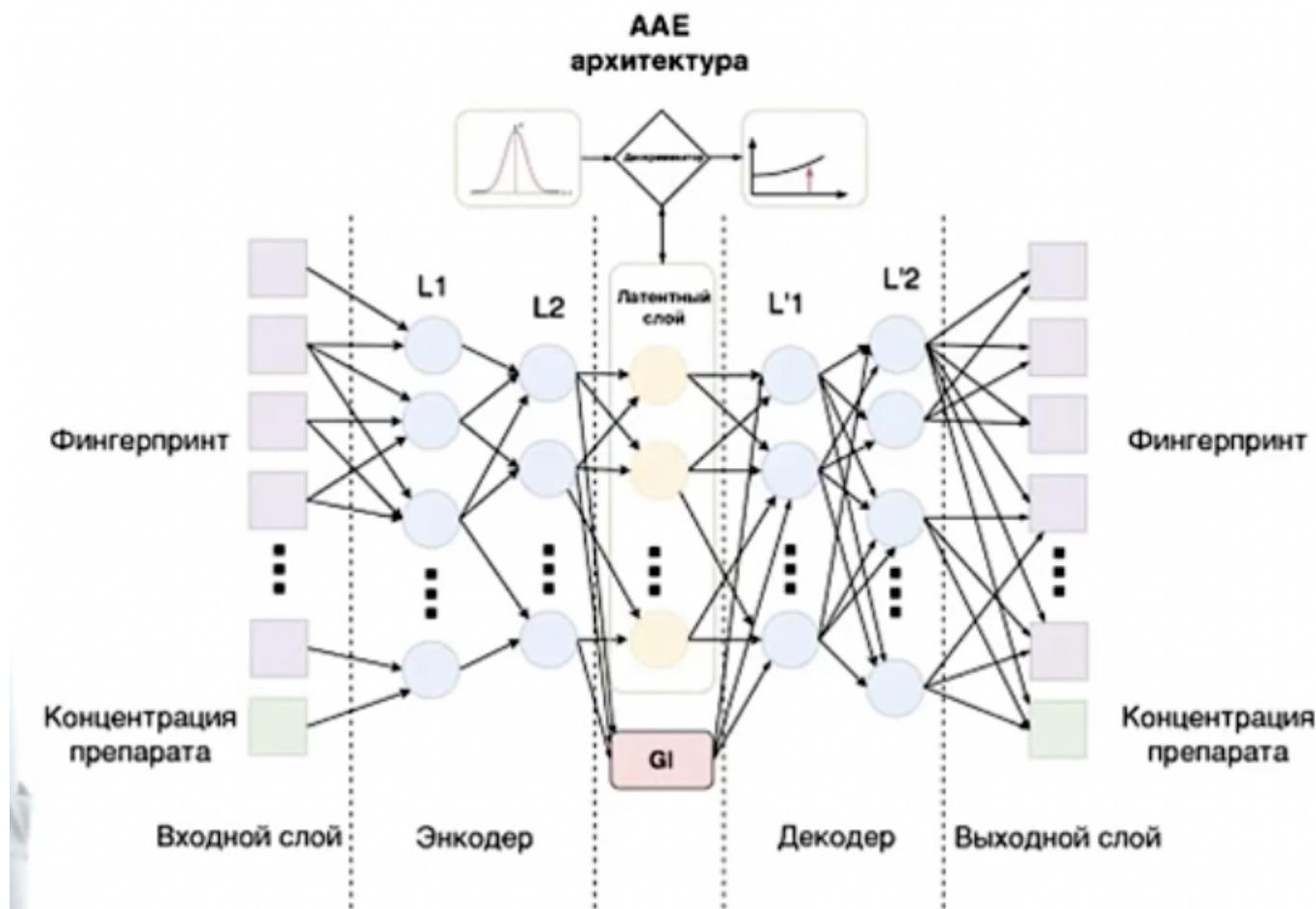
Работу таких нейросетей сложно полностью интерпретировать, а также сложно логически описать их работу после обучения.

Благодаря развитию технологий и многократному снижению стоимости комплектующих и компьютеров в целом, нейросети и машинное обучение переживают третью и самую большую волну популярности.

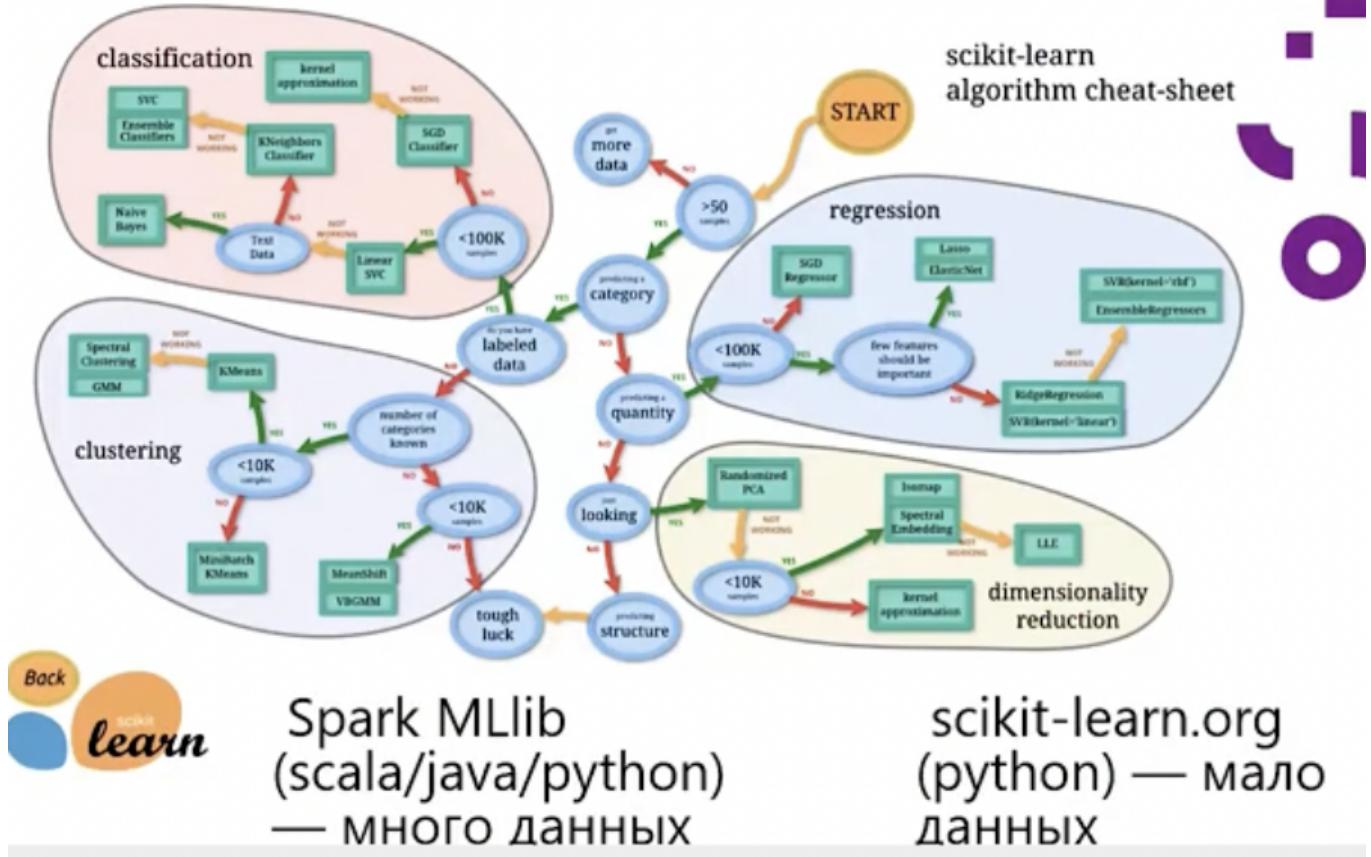
В данный момент имеются нейросети для самых различных задач. Например, для генерации интерьеров (генеративная), восстановления части изображений и т. д.

Пример нейросети для поиска способа борьбы с онкологией:

Борьба с заболеваниями



Некоторые полезные библиотеки/фреймворки:



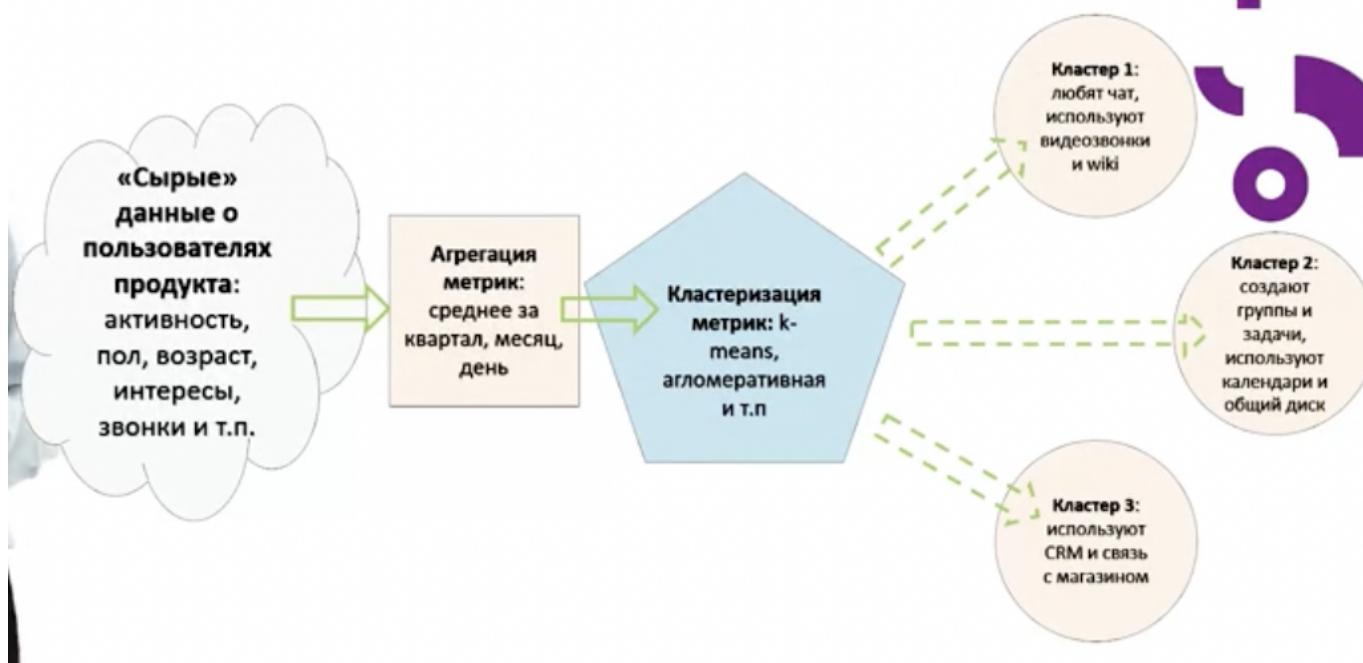
"Никогда не пытайтесь в рабочее время за деньги инвесторов повторить какой-то алгоритм с нуля"

При внедрении машинного обучения в компании начать стоит с простой аналитики – базовая статистика, крі и т.д.

Также стоит добавить визуализацию результатов, но она должна быть полезной.

После этого можно переходить к более сложным вещам, например, к кластерному анализу. Пример его использования:

Алгоритм использования кластеризации в продукте-1



Алгоритм использования кластеризации в продукте-2



Алгоритм использования кластеризации в продукте-3



Кластерный анализ: бизнес-кейсы

Сегментация клиентов, типов
использования сервиса, ...

Кластеризация «общего» товарного
каталога

Кластеризация графа связей сайтов
(пересечение аудитории)

Маркетинг работает с целевыми
группами, информация разбита на
«смысловые облака»

На следующем этапе можно внедрять персонализацию (похожа на
кластеризацию, но работает на размеченных данных) и классификацию. О ней

подробнее.

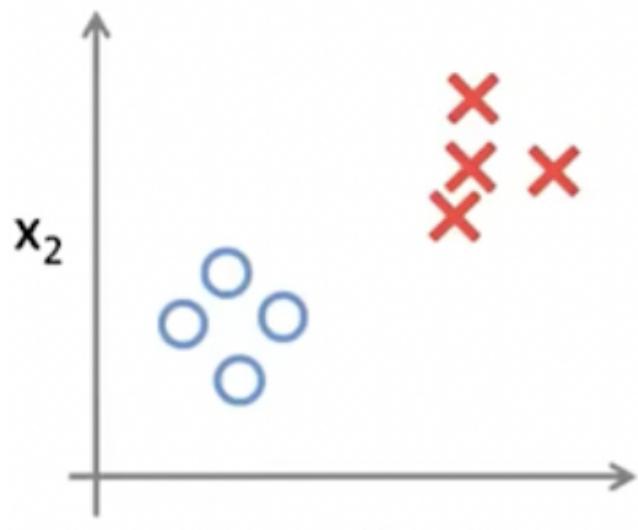
Классификация

Разбиваем по группам. Обучение

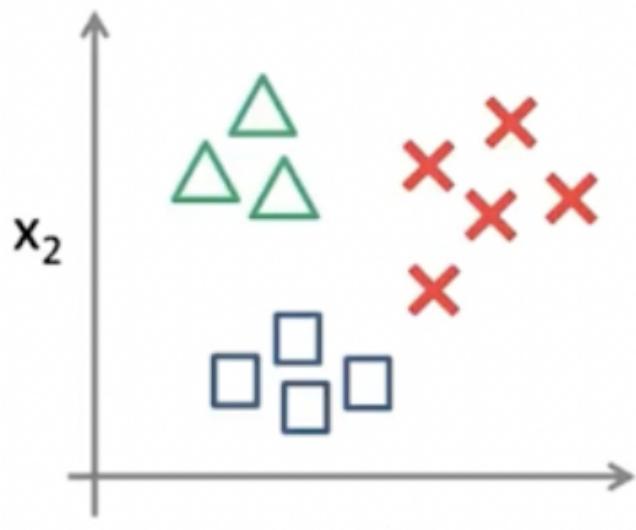
Бинарная

Мультиклассовая

Binary classification:



Multi-class classification:



Классификация: бизнес-кейсы

Удержание: найти клиентов, которые скоро уйдут (churn-rate)

Найти клиентов, готовых стать платными

Найти клиентов, которые готовы купить новую услугу

Найти готовых уволиться

Определить у клиента пол

Совет - попробовать работать с градиентным бустингом вместо более привычных/старых алгоритмов.

Как запустить проект ML-проект в облаке?

Главный совет - воспользоваться готовым решением. Например, Amazon Machine Learning

Сервис "Amazon Machine Learning"

- Только логистическая регрессия
- Работа через API
- Простой язык для feature engineering
- Простая визуализация и контроль качества классификаторов (binary, multiclass) и регрессии
- Масштабирование

Ссылка на сертификат:

<https://stepik.org/cert/1767781>



РУССКАЯ
ШКОЛА
ПРОГРАММИРОВАНИЯ

Настоящий сертификат подтверждает, что

Sergey Loginov

успешно завершил/а курс

Big Data и Data Science: перейди на новый уровень

<https://stepik.org/course/101689> | <https://stepik.org/cert/1767781>

13.11.2022