Дизайн систем машинного обучения

1. Машинное обучение на практике

https://ods.ai/tracks/ml-system-design-22

О чем будем рассказывать

- Делать модели машинного обучения легко
- Трудно сделать так, чтобы они работали хорошо
- Еще труднее сделать, чтобы ими пользовались
- Будем изучать ML-системы в реальной жизни
- С точки зрения кода, оборудования и бизнеса

Чего в курсе нет

- Алгоритмы машинного обучения
- Дизайн пользовательского интерфейса
- Статистика
- Как писать код
- Как учить нейронные сети
- Как делать веб-сайты
- Как проходить собеседование по system design

План курса

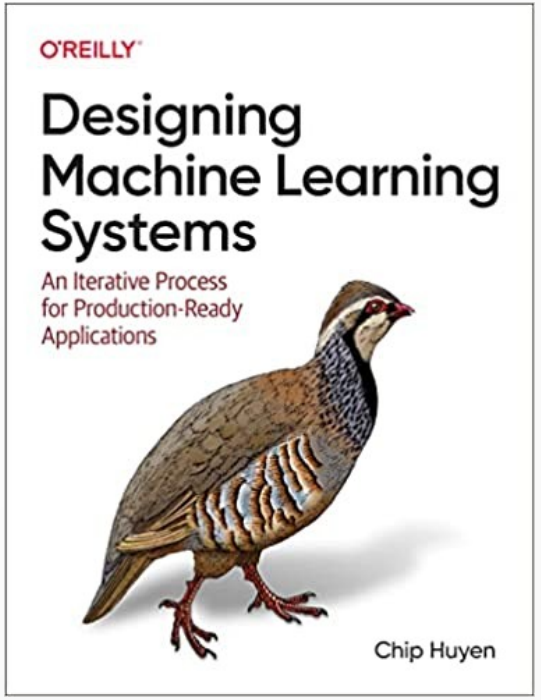
1) **Практическое применение машинного обучения — Вы находитесь здесь**
2) Основы проектирования ML-систем
3) Обучающие данные
4) Подготовка и отбор признаков
5) Выбор модели, разработка и обучение модели
6) Оценка качества модели
7) Развертывание систем
8) Диагностика ошибок и отказов ML-систем
9) Мониторинг и обучение на потоковых данных
10) Жизненный цикл модели
11) Отслеживание экспериментов и версионирование моделей
12) Сложные модели: временные ряды, модели над графами
13) Непредвзятость, безопасность, управление моделями
14) ML инфраструктура и платформы
15) Интеграция ML-систем в бизнес-процессы

Зачем вам это может пригодиться

- Запустить свой пет-проект для портфолио
- Запустить свой стартап
- Устроиться на работу
  - ML-инженером
  - MLOPS-инженером
  - Менеджером ML-проекта
  - Менеджером ML-продукта

Стоим на плечах гигантов

- Курс cs329 Machine Learning System Design
- Книга Designing Machine Learning Systems



Предварительные требования

- Любой ВУЗовский курс статистики
- Онлайн-курс https://stepik.org/course/76/info

Предварительные требования

- Любой ВУЗовский курс по программированию
- Онлайн-курс https://stepik.org/course/512/promo

Предварительные требования

- Любой ВУЗовский курс по машинному обучению
- Серия статей https://habr.com/ru/company/ods/blog/322626/

Предварительные требования

- Практический опыт программирования под Linux
- Онлайн-курс https://missing-semester-rus.github.io/

Предварительные требования

- Желательно — опыт работы с нейронными сетями и, например, pytorch. https://pytorch.org/tutorials/

Дизайн систем машинного обучения

- Процесс принятия решений про
  - Интерфейс
  - Алгоритмы, данные
  - Программную инфраструктуру
  - Оборудование
- Чтобы соответствовать требованиям и ограничениям, например, по:
  - Надежности (reliable)
  - Масштабируемости (scalable)
  - Обслуживаемости (maintainable)
  - Адаптируемости (adaptable)

Дизайн начинается с ограничений

- Требования/ограничения придется выяснять самим
- Если кто-то выдал вам требования, они неполные и противоречивые
- Требования — всегда предположения
- Предположите что-нибудь
- Придумайте, как проверить предположение
- Узнав новое, уточните предположения

Предположения ML

Машинное обучение — это автоматизированный подход к выявлению сложных шаблонов в имеющихся данных и использование этих шаблонов для предсказаний на новых данных.

- Мы можем выявить шаблоны
- Шаблоны сложные
- Данные имеются
- Можем предсказывать
- Будут новые данные

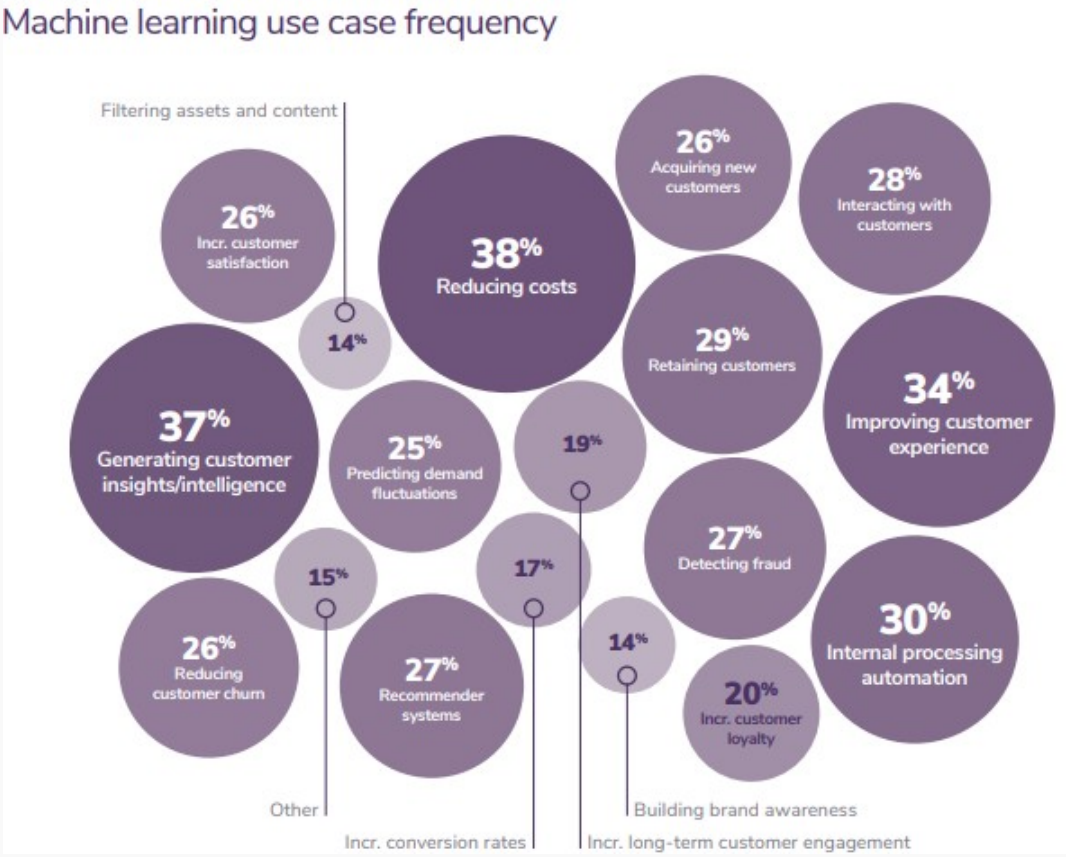
Делать ML

- Часто повторяющаяся задача
- Цена ошибки невелика
- Большой масштаб
- Шаблоны постоянно меняются

Не делать ML

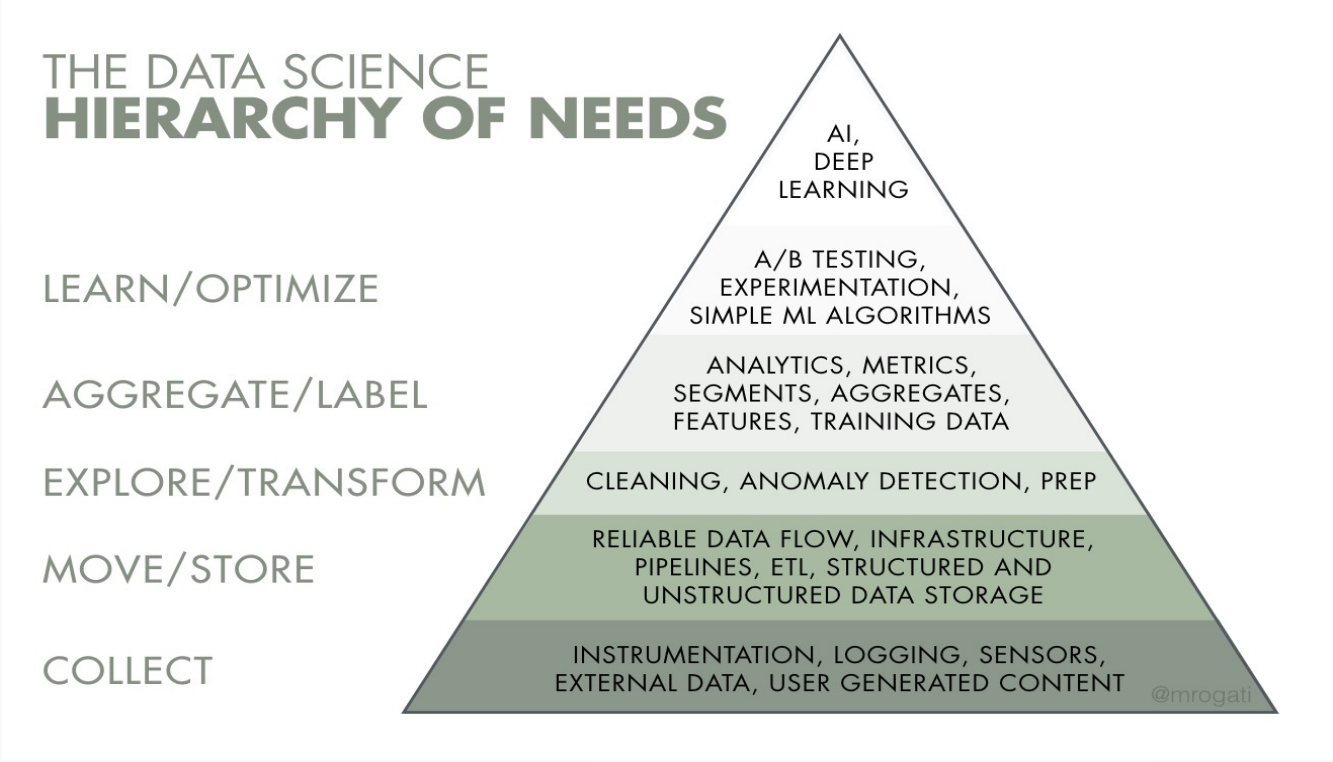
- Это неэтично
- Простое правило решает проблему
- Данные недоступны
- Цена ошибки высока
- Каждое решение должно быть объяснимо
- Дешевле нанять человека

Традиционное применение ML



2020 state of enterprise machine learning

Основание



https://hackernoon.com/the-ai-hierarchy-of-needs-18f111fcc007

Индустрия vs Исследования

Часто и небезосновательно:

- Ученые не умеют писать код
- Не воспроизводится
- Неприменимо

Часто и небезосновательно:

- Инженеры не знают основ
- Лишь бы работало
- Не проверяют базовые предположения

* Автор курса на стороне индустрии

Исследования

- Требования: Публикабельность
- Вычисления: Быстрое обучение и пропускная способность
- Данные: Обычно не меняются
- Интерпретируемость: Обычно не важна
- Поддерживаемость: Не важна
- Масштабируемость: Один и тот же масштаб

Индустрия

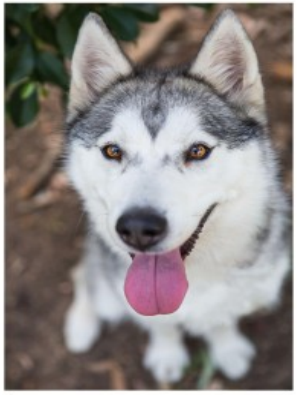
- Требования: Разные требования внутри организации
- Вычисления: Быстрый инференс и низкая задержка
- Данные: Постоянный сдвиг данных
- Интерпретируемость: Может быть очень важна
- Поддерживаемость: Может быть определяющей метрикой
- Масштабируемость: Может быть определяющей метрикой

Разные интересы

Stakeholder objectives

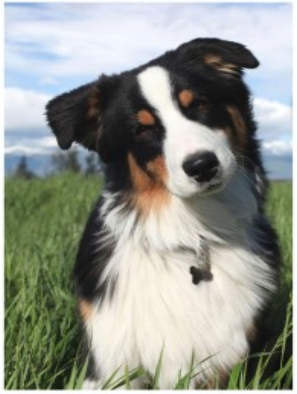
ML team
highest accuracy

Sales
sells more ads



Product
fastest inference

Manager
maximizes profit
= laying off ML teams



https://docs.google.com/presentation/d/1XKBXoLW3ED3FcvDRK3jfQeIlLCK2QTnsoZeopR6ziRQ/edit#slide=id.gb4b40ae9d8_0_16

Обучение vs Инференс

- Каждый показ страницы интернет-магазина или агрегатора может использовать результаты десятков ML-моделей.
  Или сотен.
- Лишняя доля секунды превращается в отказы от покупки
- Быстрые сайты выше в выдаче
- Конвейер или автомобиль едет быстро — нужна маленькая задержка
- Клиентов у банка много — нужна высокая пропускная способность
- А еще в индустрии модели часто переобучают и доучивают,
  и важно — как быстро это удается сделать

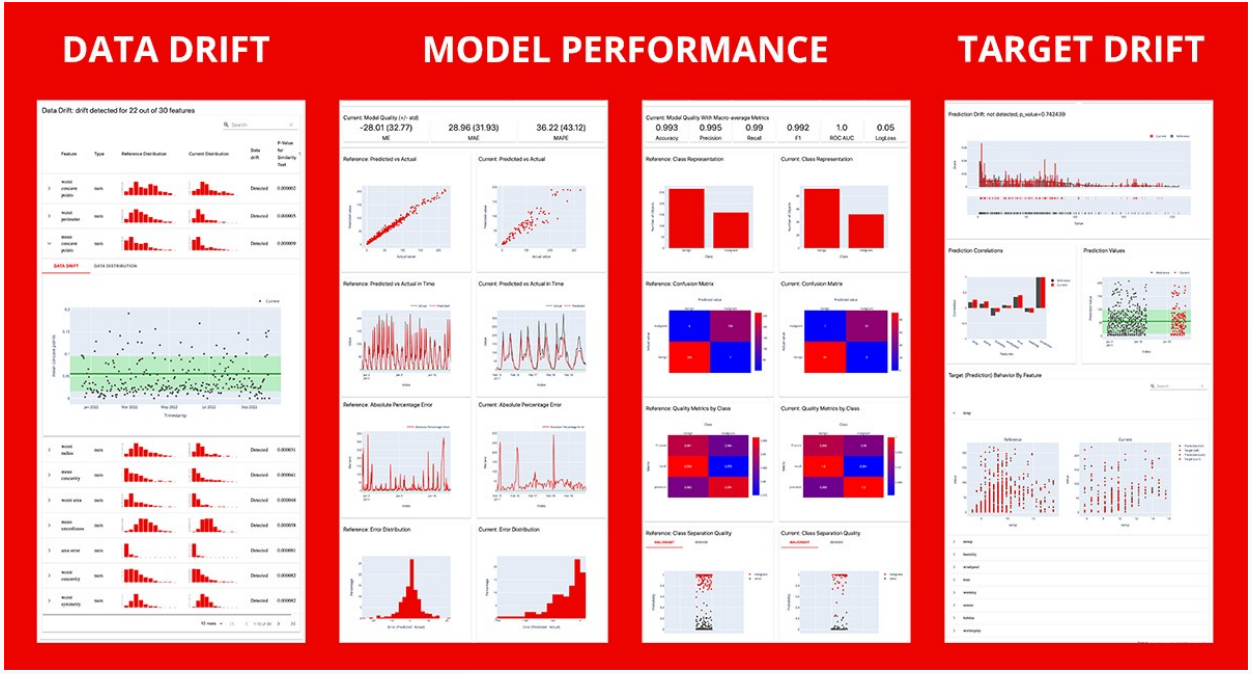
Пропускная способность

- Сколько предсказаний вы сделаете за секунду
- Важна, когда мы обсчитываем данные «пачками»
- Можно увеличить, докупив/арендовав оборудование

Задержка

- Сколько времени займет одно предсказание
- Важна, когда мы обрабатываем данные на лету
- Иногда можно увеличить, купив более быстрое железо
- Например, 300 предсказаний в секунду.
- Обрабатываем пакетами по 100 предсказаний
- Каждый занимает 0,3 сек.
- Пропускная способность — 300 RPS
- Задержка — 300 мсек
- 300 последовательных запросов — 90 секунд

Данные меняются



https://github.com/evidentlyai/evidently

Например

- Меняются входные данные — все стали носить маски
- Меняется мир — человек в маске — не хирург
- Датасетов со «сдвигом» мало. Примеры
  - Shift датасеты от Яндекса
  - WILDS 7 датасетов для детекции сдвига
  - Multimodal Single-Cell Integration
- Индустрия с этим живет

Про интерпретируемость

Geoffrey Hinton
@geoffreyhinton

Suppose you have cancer and you have to choose between a black box AI surgeon that cannot explain how it works but has a 90% cure rate and a human surgeon with an 80% cure rate. Do you want the AI surgeon to be illegal?

3:37 AM · 21 февр. 2020 г. · Twitter Web App

1 123 ретвита **615** твитов с цитатами **5 058** отметок «Нравится»

david @dwhdai · 21 февр. 2020 г.
В ответ @geoffreyhinton
Are we assuming that the human can perfectly explain the situation?

1 6

Josip Krapac @josipK · 21 февр. 2020 г.
In most cases humans can explain their actions, but post-hoc. Humans are good in constructing consistent and persuasive narratives, but these stories are not plans, they're not conscious causes of their actions.

2 4

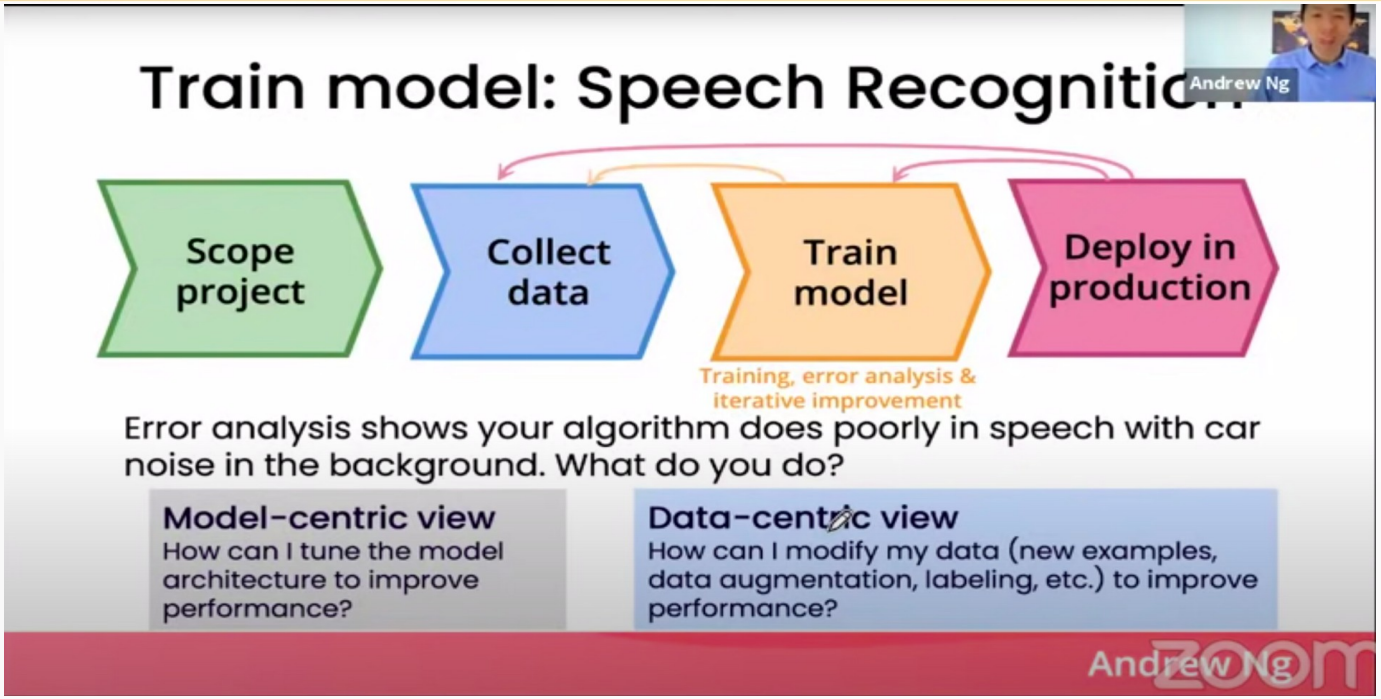
Как ML меняет разработку

- Данные
- Управление проектом
- Мониторинг

Данные

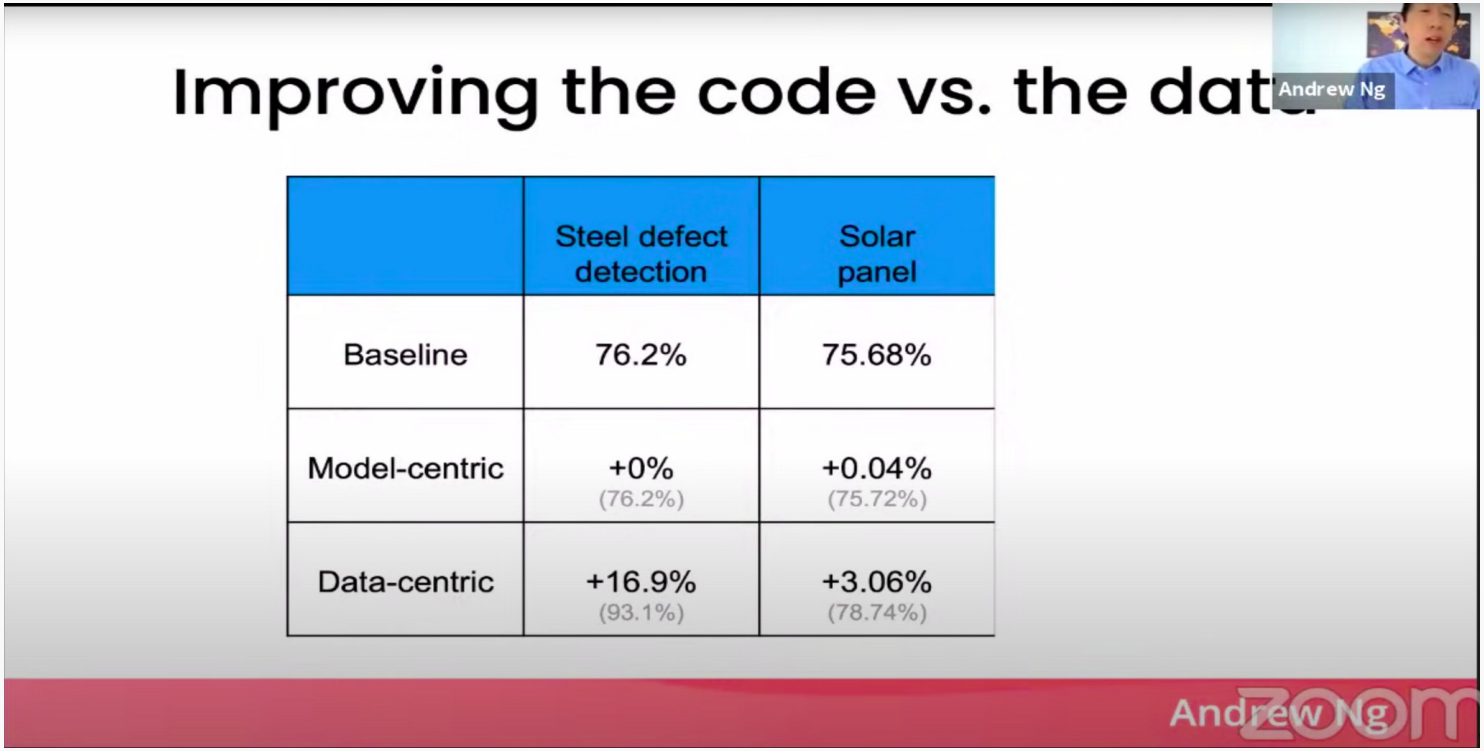
- ML — модель сцеплена с данными
- Данные **обычно** не версионируются
  - Версионируются схемы данных, но не данные
- Данные **обычно** необозримы
  - Какая точка данных портит вашу модель?
  - Данных **обычно** больше, чем кода
  - Diff **обычно** не работает
- Предположения о данных нужно проверять
  - Нужно, но трудно

Данные важны



https://www.youtube.com/watch?v=06-AZXmwHjo

Данные важнее моделей



https://www.youtube.com/watch?v=06-AZXmwHjo

Управление проектом

- Трудно управлять качеством
- Трудно управлять сроками
- Непонятны границы возможного
- Дефекты возникают не в коде
- Но править их приходится в коде
- Двухфазные проекты:
  - Discovery: фиксированное время
  - Delivery: фиксированные задачи

Мониторинг

- Как понять, что оно работает так же хорошо, как вчера?
- Как понять, что оно работает хорошо?
- Как понять, что оно вообще работает?
- Иногда отказ ML-модели долго не замечают
- Иногда «сломанные данные» долго не замечают
- Иногда эффект можно измерить только через полгода

Вроде как правила

- 50% эффекта ML-модели можно достичь без ML, правилами, регулярными выражениями и т.д.
- Если эксперт не видит в данных закономерностей, ML тоже не увидит
- Большинство проблем — на границах системы
- Основные затраты — данные
- Если метрика не в деньгах, она не важна.

https://www.youtube.com/watch?v=tlB4CL9Esl0

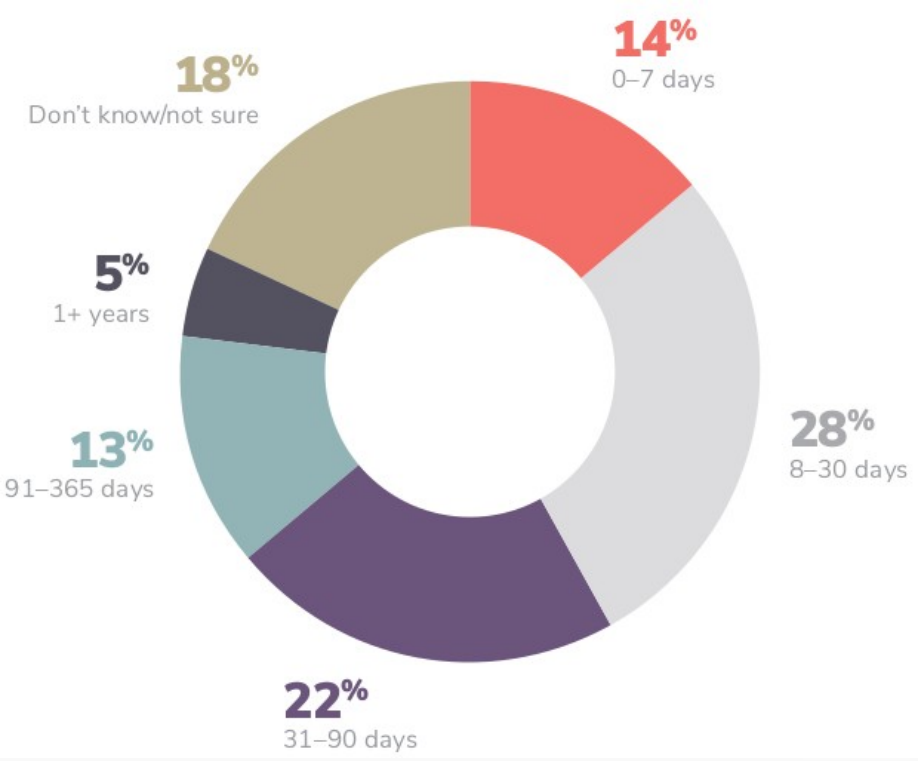
Еще проблемы

- Нужно быстро переобучить модель
- Важно — как часто вы сможете ее переобучать
- Скорее всего моделей будет много
- В данных могут быть закладки
- В данных могут быть ошибки

Горькая правда

- Хороший программист быстро научится использовать ML
- Хорошему исследователю трудно хорошо программировать
- Оба не сделают ничего полезного без DevOps инженера

Развертывание готовой модели



2020 state of enterprise machine learning

Дополнительные материалы

- Rules of ml
- Hidden Technical Debt in Machine Learning Systems
- How to avoid machine learning pitfalls

Все будет в телеграм-канале