

# Введение в ML

Элен Теванян

16 мая 2023



# Инструменты для сбора, обработки и анализа данных

# Инструменты для сбора, обработки и анализа данных

Python

# Инструменты для сбора, обработки и анализа данных

Python



# Инструменты для сбора, обработки и анализа данных

Python



# Инструменты для сбора, обработки и анализа данных

Python



colab

# Инструменты для сбора, обработки и анализа данных

Python



colab

# Инструменты для сбора, обработки и анализа данных

Python



colab

+ ещё много опций

# Инструменты для сбора, обработки и анализа данных

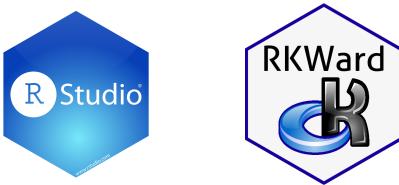
Python



colab

+ ещё много опций

R



+ что-то ещё точно есть

# Инструменты для сбора, обработки и анализа данных

Python



colab

+ ещё много опций

R



+ что-то ещё точно есть

C++



Надеюсь, вас минует

# Инструменты для работы

1. Решения для хранения данных
2. Инструменты для сбора, обработки и анализа данных
- 3. Инструменты для сбора, обработки и анализа данных**

# Визуализация данных

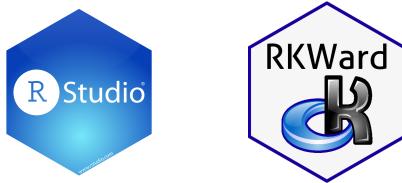
# Визуализация данных

Python



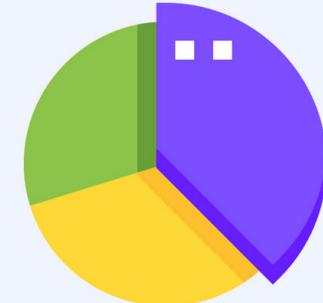
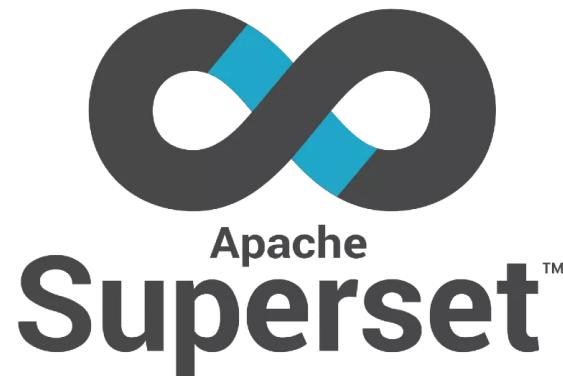
colab

R



# Визуализация данных

- Языки программирования и библиотеки под них
- BI-инструменты:



YANDEX  
DATALENS



# Описание данных

- **state**, *string*. 2-letter code of the US state of customer residence
- **account\_length**, *numerical*. Number of months the customer has been with the current telco provider
- **area\_code**, *string*="area\_code\_AAA" where AAA = 3 digit area code.
- **international\_plan**, *(yes/no)*. The customer has international plan.
- **voice\_mail\_plan**, *(yes/no)*. The customer has voice mail plan.
- **number\_vmail\_messages**, *numerical*. Number of voice-mail messages.
- **total\_day\_minutes**, *numerical*. Total minutes of day calls.
- **total\_day\_calls**, *numerical*. Total number of day calls.
- **total\_day\_charge**, *numerical*. Total charge of day calls.
- **total\_eve\_minutes**, *numerical*. Total minutes of evening calls.
- **total\_eve\_calls**, *numerical*. Total number of evening calls.
- **total\_eve\_charge**, *numerical*. Total charge of evening calls.
- **total\_night\_minutes**, *numerical*. Total minutes of night calls.
- **total\_night\_calls**, *numerical*. Total number of night calls.
- **total\_night\_charge**, *numerical*. Total charge of night calls.
- **total\_intl\_minutes**, *numerical*. Total minutes of international calls.
- **total\_intl\_calls**, *numerical*. Total number of international calls.
- **total\_intl\_charge**, *numerical*. Total charge of international calls
- **number\_customer\_service\_calls**, *numerical*. Number of calls to customer service
- **churn**, *(yes/no)*. Customer churn - target variable.

# «Под капотом» машинного обучения

## 1. Основные термины

# Пример задачи

- Для улучшения эффективности диспетчерских служб такси важно знать, когда водитель закончит один заказ и будет готов принять следующий.
- Оценка длительности текущей поездки – один из факторов эффективного распределения заказов.
- Как оценить длительность поездки?

# Терминология

- $x$  (sample) – **объект**, для которой хотим делать предсказания
- $y$  (target) – **целевая переменная**, т.е. то, что хотим предсказать
- $(x_i, y_i)_{i=1}^{\ell}$  – обучающая выборка, прецеденты, т.е. все объекты, для которых известны значения целевого признака,  
 $\ell$  – размер выборки.

# Терминология

- $x$  (sample) – объект, для которой хотим делать предсказания
  - **Поездки**
- $y$  (target) – ответ, целевая переменная, т.е. То, что хотим предсказать
  - **Длительность поездки**
- $(x_i, y_i)_{i=1}^{\ell}$  – обучающая выборка, прецеденты, т.е. все объекты, для которых известны значения целевого признака,  
 $\ell$  – размер выборки.

# Терминология

- Компьютер умеет работать с числовой информацией
- Объекты характеризуются числовой информацией – **признаками**, факторами, «фичами» (от англ. features)
- $x = (x^1, \dots, x^m)$ ,  $m$  – число признаков

# Признаки для задачи?

- Для улучшения эффективности диспетчерских служб такси важно знать, когда водитель закончит один заказ и будет готов принять следующий.
- Оценка длительности текущей поездки – один из факторов эффективного распределения заказов.
- Как оценить длительность поездки?

# Признаки для задачи

- Временные
- Географические
- Погодные
- Маршруты
- Пассажиры

# Признаки для задачи

- Временные
  - Дата и время посадки
  - Дата и время высадки
- Географические
  - Широта и долгота посадки
  - Широта и долгота высадки
- Погодные
  - Осадки
  - Сила осадков
- Маршруты
  - Наиболее быстрые маршруты
  - Скорость по маршрутам
- Пассажиры
  - Число пассажиров

# Алгоритм, модель, формула

- $a(x)$  – алгоритм/модель
- Это функция, предсказывающая ответ для любого объекта

# Два подхода к моделированию

# Два подхода к моделированию



# Два подхода к моделированию



# Два подхода к моделированию



Дискриминативное обучение



Генеративное обучение

# Виды задач в машинном обучении

## машинаное обучение

с учителем (supervised learning – SL)

Есть данные с разметкой, которые мы можем показать «машине» и обучить ее прогнозировать разметку

**Пример задачи:** знаем тип каждого яблока, надо научиться прогнозировать тип яблока

- Регрессия
- Классификация

без учителя (unsupervised learning - UL)

Есть данные, мы не знаем разметки, мы хотим, чтобы машина научилась извлекать знание из данных

**Пример:** разбить яблоки на 2 группы, потому что знаем, что там есть 2 вида яблок, но не знаем вид конкретного яблока

- Кластеризация
- Рекомендательные системы

# «Под капотом» машинного обучения

1. Основные термины
2. Параметры модели

# Модели:

- Линейные

$$y = w_0 + w_1 x_1 + \dots + w_n x_n$$

- Деревья:

- Если  $x_i > t_k$ , то прогноз – 4 минуты

# Модели:

- Линейные

$$y = w_0 + w_1x_1 + \dots + w_nx_n$$

- Деревья:

- Если  $x_i > t_k$ , то прогноз – 4 минуты
- $w_i, t_k$  - это параметры моделей, их значения определяются при обучении модели

# «Под капотом» машинного обучения

1. Основные термины
2. Параметры модели
- 3. Функции потерь**

# ФУНКЦИЯ ПОТЕРЬ

- Как перейти
  - от «наверное, время поездки зависит от километража и времени суток»
  - к «длительность поездки = 1.5\*кол-во метров» ?

# ФУНКЦИЯ ПОТЕРЬ

- Как перейти
  - от «наверное, время поездки зависит от километража и времени суток»
  - к «длительность поездки = 1.5\*кол-во метров» ?
- = Найти лучшие значения параметров модели
- = Найти алгоритм  $a(x)$ :  $a(x_i) \approx y_i$

# ФУНКЦИЯ ПОТЕРЬ

- Как перейти
  - от «наверное, время поездки зависит от километража и времени суток»
  - к «длительность поездки = 1.5\*кол-во метров» ?
- = Найти лучшие значения параметров модели
- = Найти алгоритм  $a(x)$ :  $a(x_i) \approx y_i$

# ФУНКЦИЯ ПОТЕРЬ

- Это мера корректности алгоритма
- «Потери», маскируемые  $\approx$  в нашем желании приблизить данные алгоритмом ( $a(x_i) \approx y_i$ )

# ФУНКЦИЯ ПОТЕРЬ

## Регрессия

- MSE
- MAE
- Huber

## Классификация

- Logloss
- Softmax
- Hinge

# Скромные пожелания

- Дифференцируемость
  - Ошибку минимизируем, т.е. решаем оптимационную задачу

# MSE

- Mean Squared Error
- Функция потерь для задачи **регрессии**

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

- $y_i$  - значение
- $a(x_i)$  – значение по модели (aka прогноз)
- $\ell$  - количество элементов

# MAE

- Mean Absolute Error
- Функция потерь для задачи **регрессии**

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

- $y_i$  - значение
- $a(x_i)$  – значение по модели (aka прогноз)
- $\ell$  - количество элементов

# LogLoss

- Логистическая функция потерь
- Функция потерь для задачи **классификации**

$$logloss = -\frac{1}{\ell} \sum_{i=1}^{\ell} [y_i \ln(p_i) + (1 - y_i) \ln(1 - p_i)]$$

- $y_i$  – значения (0 или 1)
- $p_i$  – модель/формула выдачи вероятностей в з-ти от признаков (сигмоида, например)
- $\ell$  количество элементов

# «Под капотом» машинного обучения

1. Основные термины
2. Параметры модели
3. Функции потерь
- 4. Как модели обучаются**

# Обучение моделей

## Аналитическое решение

- Пишем оптимизационную задачу
- Вспоминаем курс матана
- Применяем математическую магию и получаем решение «уравнения»

## Численные методы

- Пишем оптимизационную задачу
- Вспоминаем курс матана и что просто так формулу не вывести
- Вспоминаем курс численных методов
- Эвристиками «приближаемся» к решению

# На примере линейной модели

- Хотим прогнозировать длительность поездки на такси ( $y$ )
- Есть какой-то набор признаков:
  - $x_1$ : Километраж поездки
  - $x_2$ : Количество людей
  - $x_3$ : Температура в градусах
  - $x_4$ : мл осадков
- $y = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4$
- Как узнать:  $w_0, w_1, w_2, w_3, w_4$ ?

Слайд для идей и предложений

# «Под капотом» машинного обучения

1. Основные термины
2. Параметры модели
3. Функции потерь
4. Как модели обучаются
- 5. Аналитический подход**

# Аналитическое решение

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

# Аналитическое решение

- $MSE = \frac{1}{\ell} \sum_{i=1}^{\ell} (y_i - a(x_i))^2$
- $a(x) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4$

# Аналитическое решение

- $MSE = \frac{1}{\ell} \sum_{i=1}^{\ell} (y_i - a(x_i))^2$
- $a(x) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4$
- $(y_i - w_0 - w_1x_1 - w_2x_2 - w_3x_3 - w_4x_4)^2 \rightarrow \min$

# Аналитическое решение

- $MSE = \frac{1}{\ell} \sum_{i=1}^{\ell} (y_i - a(x_i))^2$
- $a(x) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4$
- $(y_i - w_0 - w_1x_1 - w_2x_2 - w_3x_3 - w_4x_4)^2 \rightarrow \min$
- Время для производных

# Аналитическое решение

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

$$\bullet \text{ } w = (X^T X)^{-1} X y$$

# «Под капотом» машинного обучения

1. Основные термины
2. Параметры модели
3. Функции потерь
4. Как модели обучаются
5. Аналитический подход
- 6. Градиентный спуск**

# Численное решение aka градиентный спуск

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

# Градиент

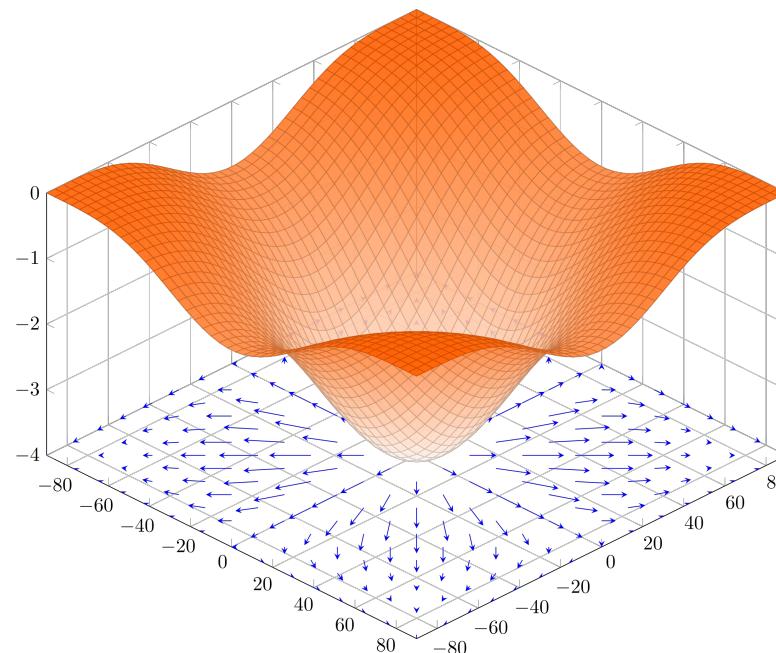
- Градиент — вектор частных производных

$$\nabla f(x) = \left( \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_d} \right)$$

- У градиента есть важное свойство!

# Важное свойство

- Зафиксируем точку  $x_0$
- В какую сторону функция быстрее всего растёт?



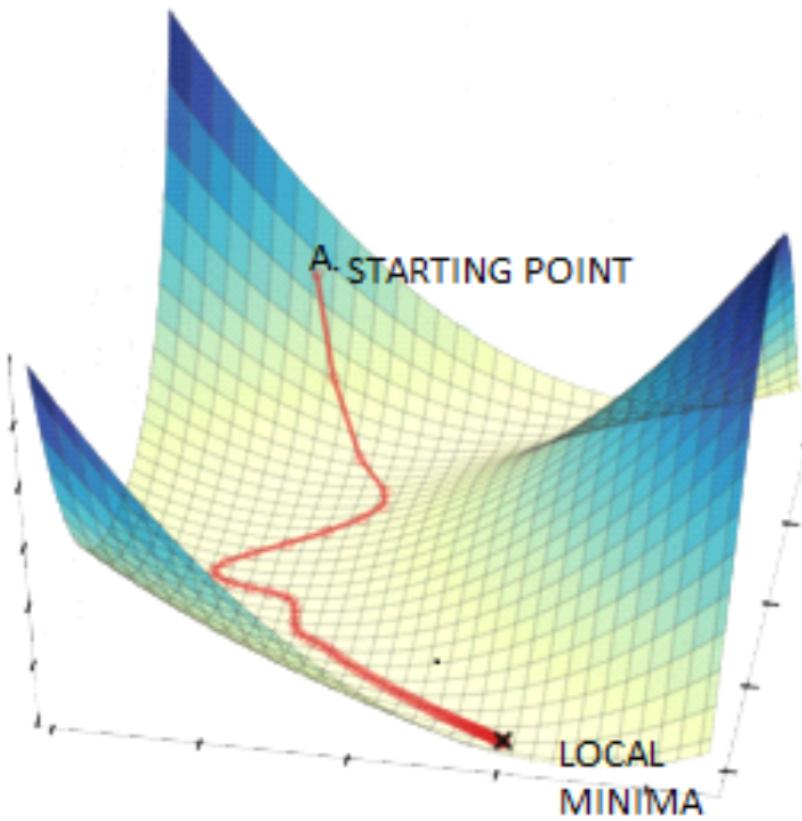
# Важное свойство

- Зафиксируем точку  $x_0$
- В какую сторону функция быстрее всего растёт?
- В направлении градиента!
- А быстрее всего убывает в сторону антиградиента

Как это пригодится?



# Как это пригодится?



# Градиентный спуск

- Стартуем из случайной точки
- Сдвигаемся по антиградиенту
- Повторяем, пока не окажемся в точке минимума

# Начальное приближение

- $w^0$  – инициализация весов
- Например, из стандартного нормального распределения

# Градиентный спуск

- Повторять до сходимости:

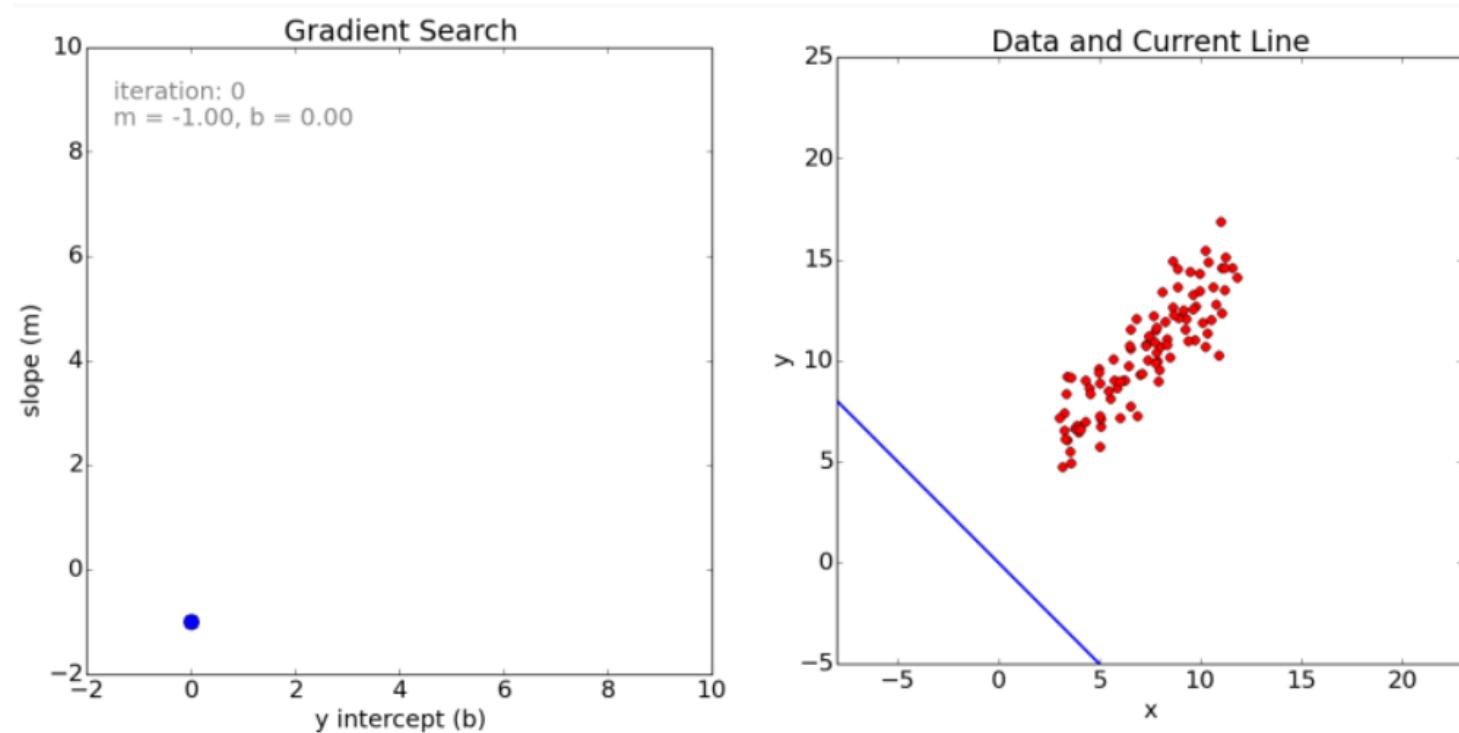
$$w^t = w^{t-1} - \eta \nabla Q(w^{t-1})$$

Новая точка

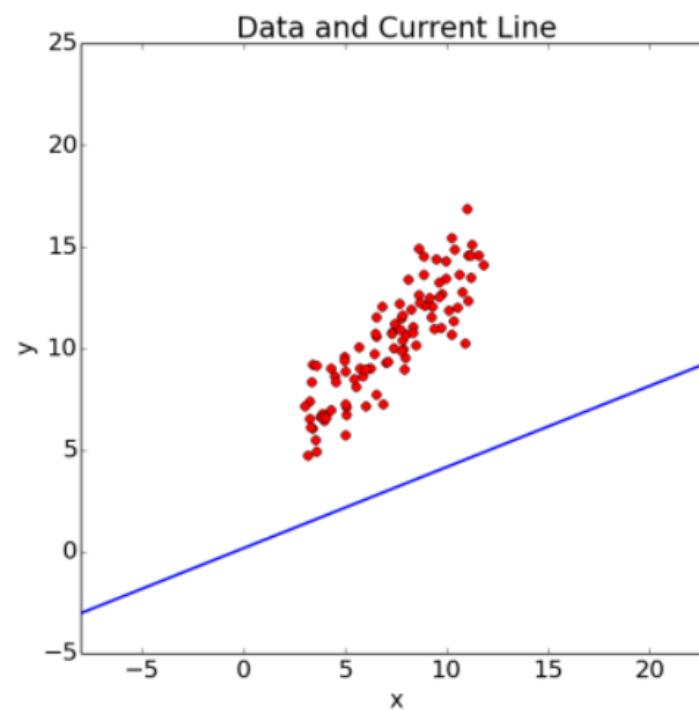
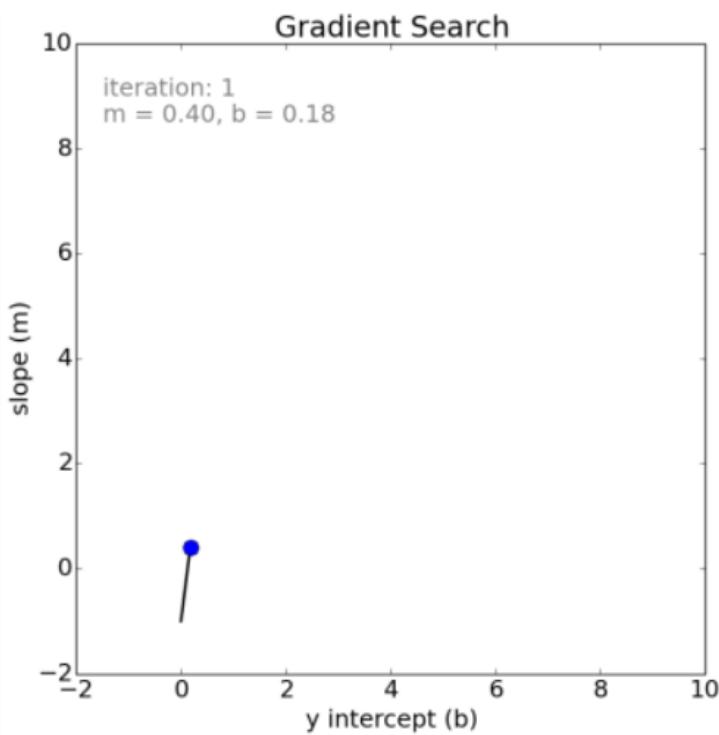
Размер шага

Градиент в  
предыдущей  
точке

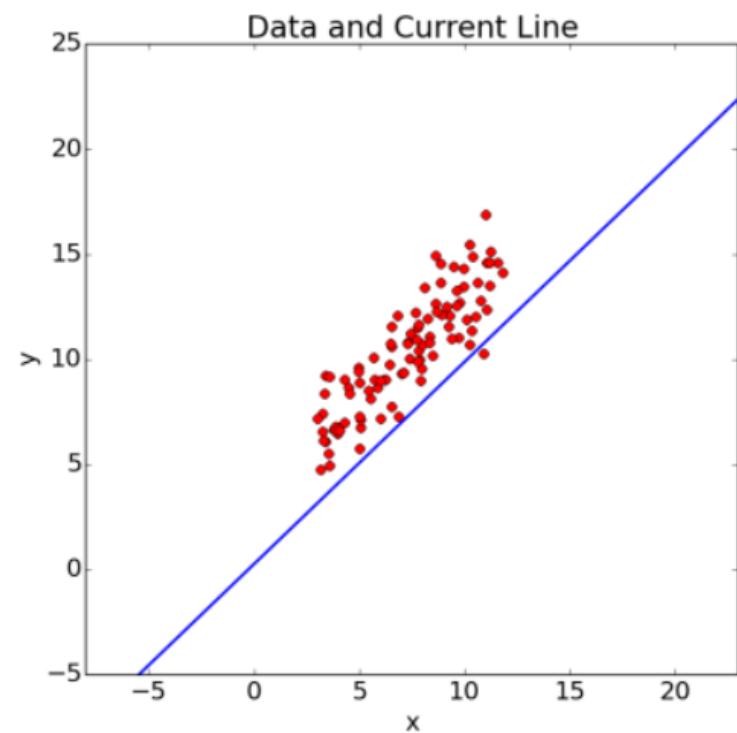
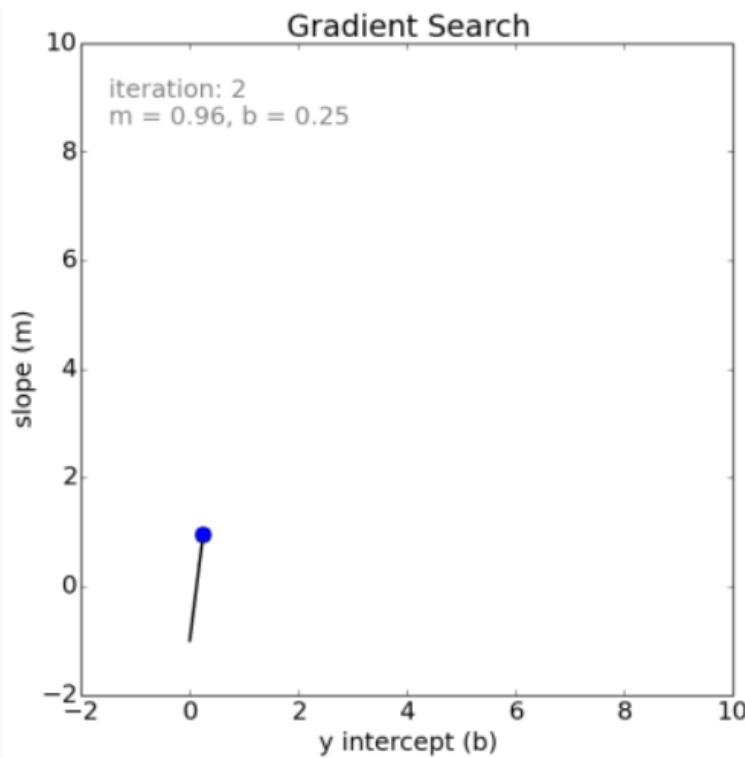
# Парная регрессия



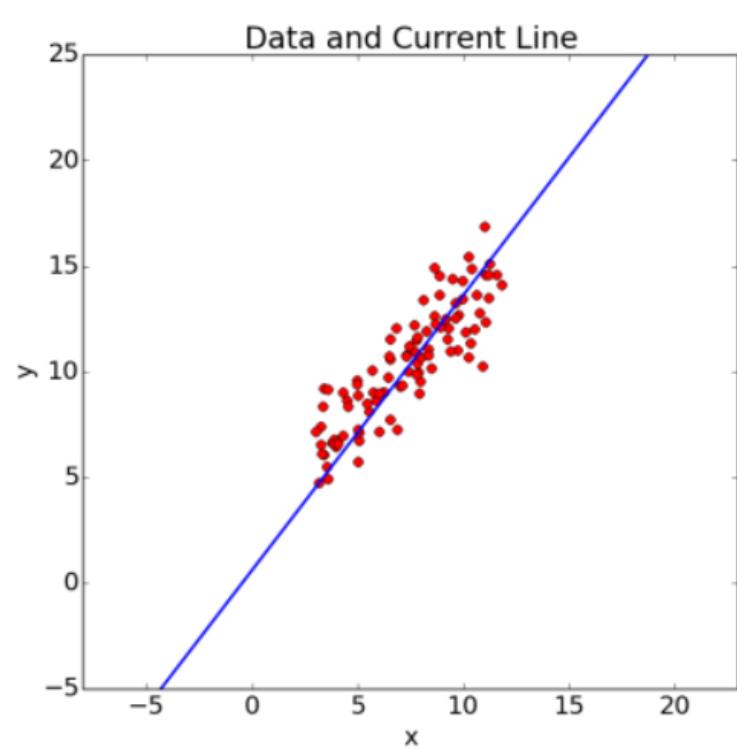
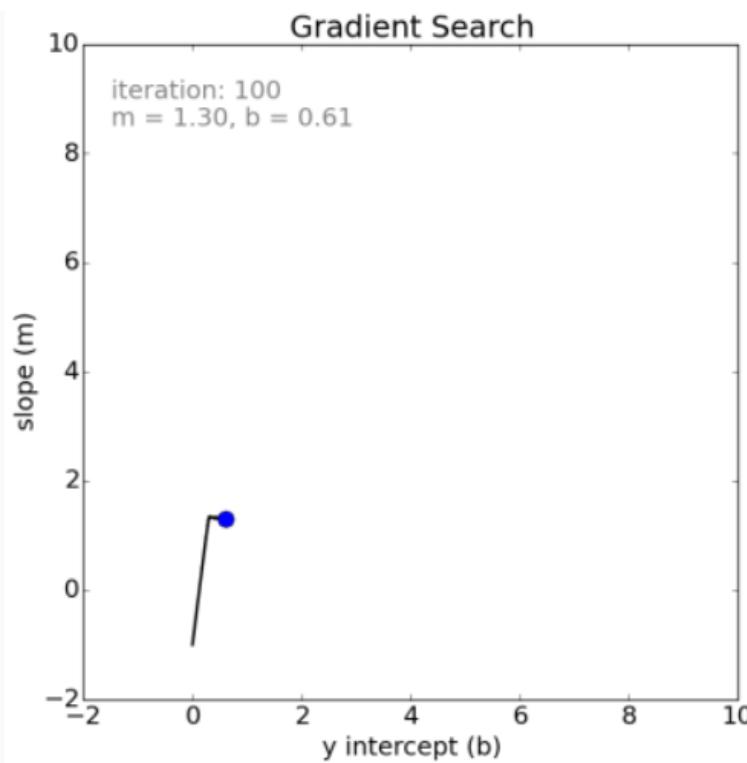
# Парная регрессия



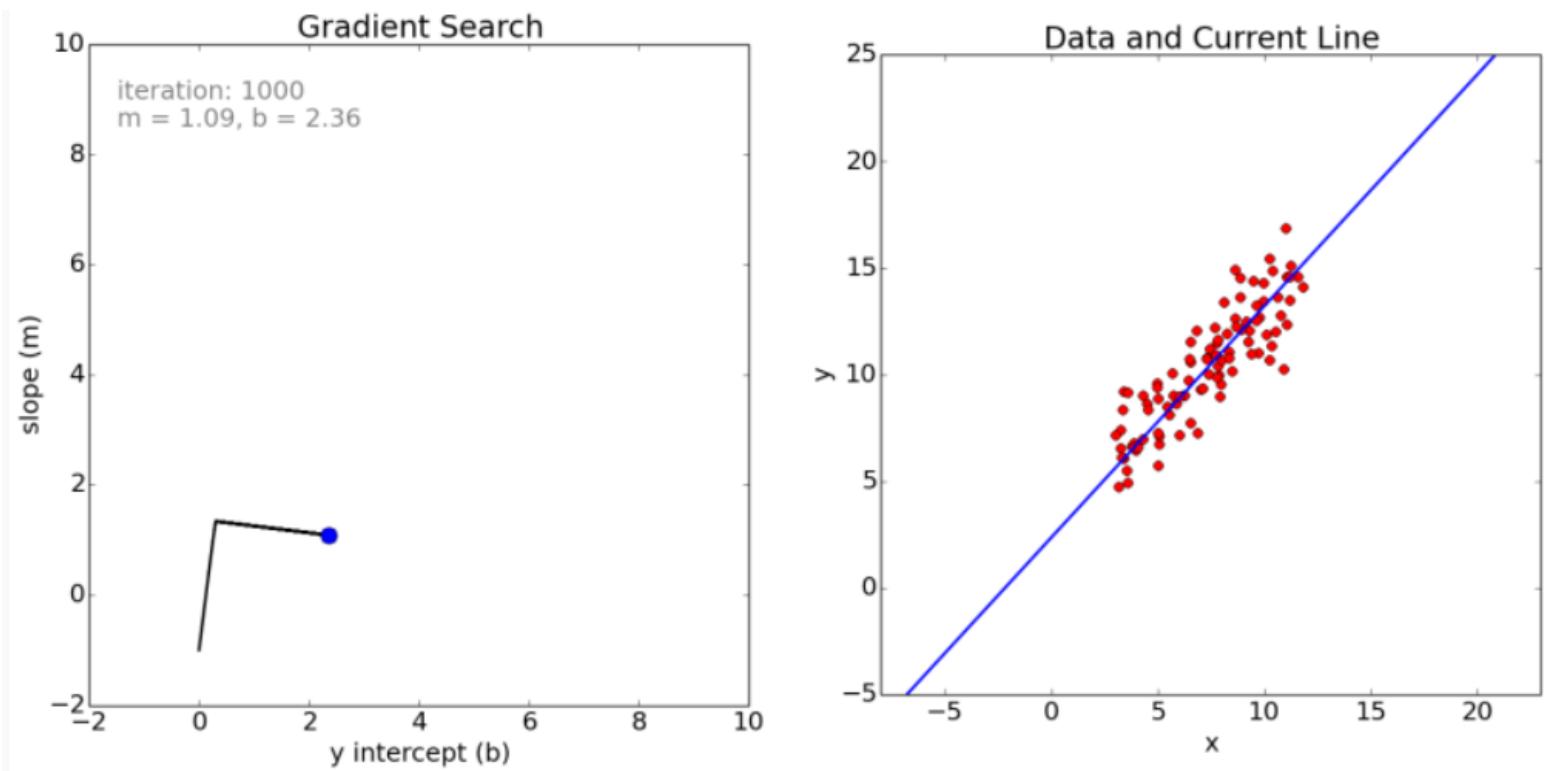
# Парная регрессия



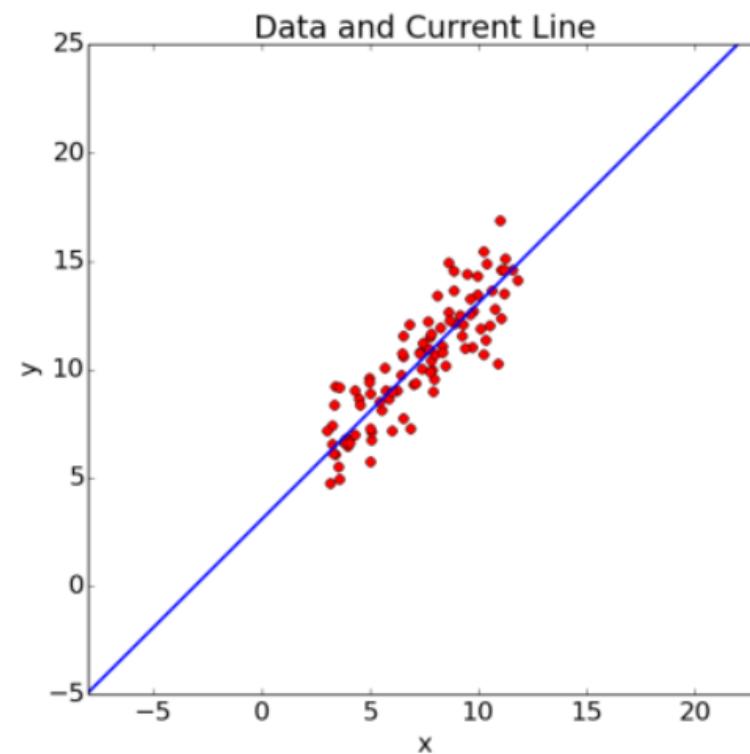
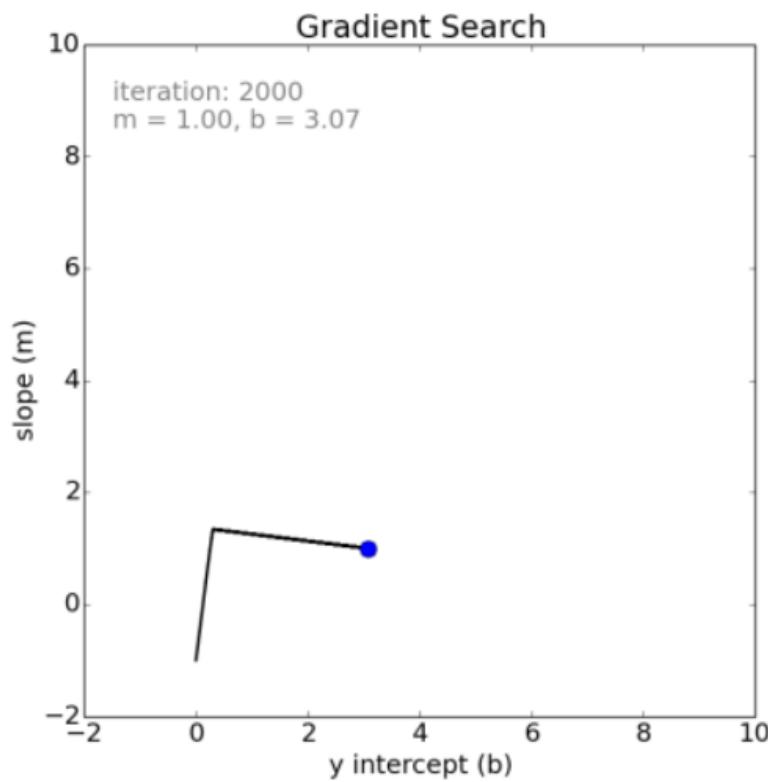
# Парная регрессия

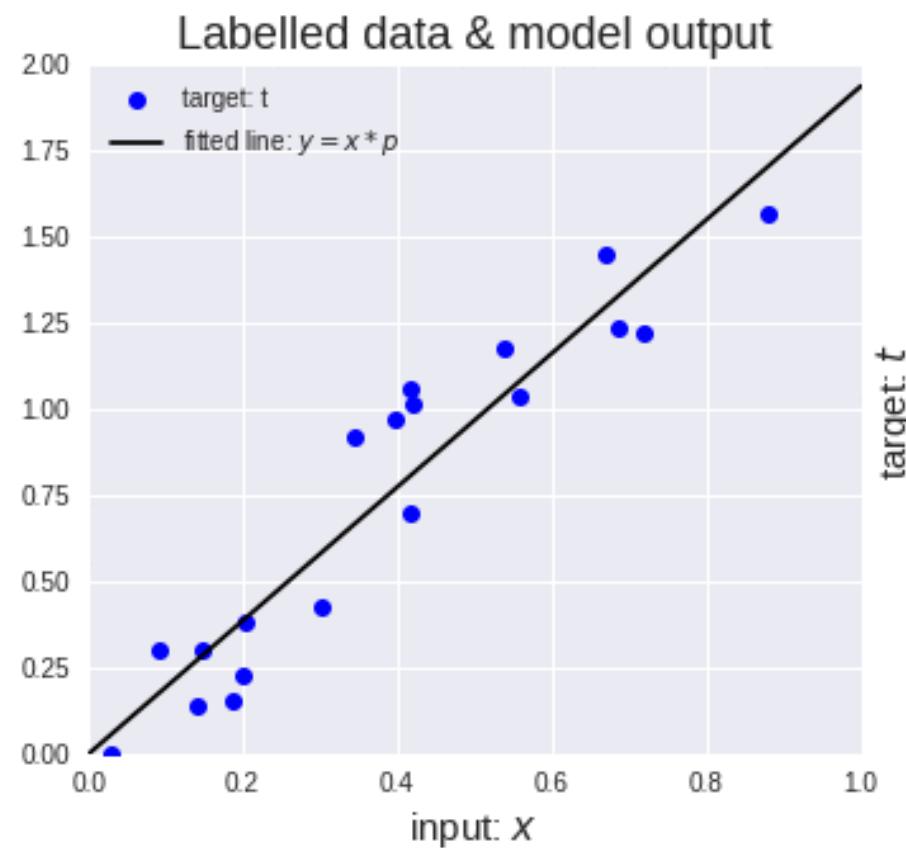
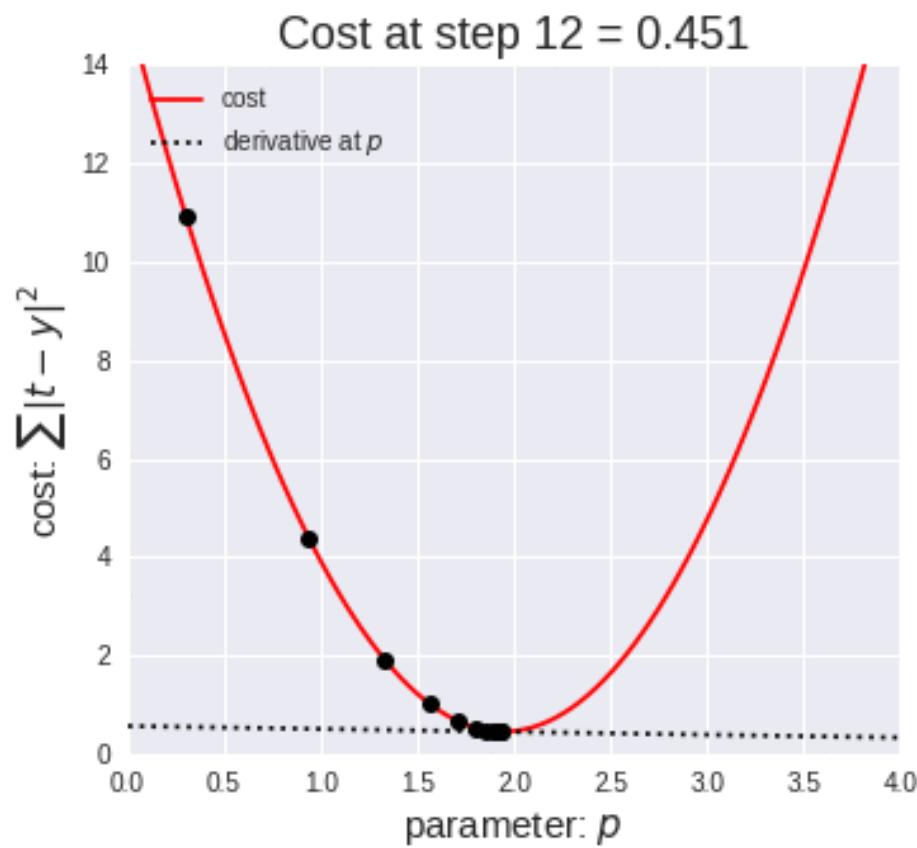


# Парная регрессия



# Парная регрессия





# Функционал ошибки

