

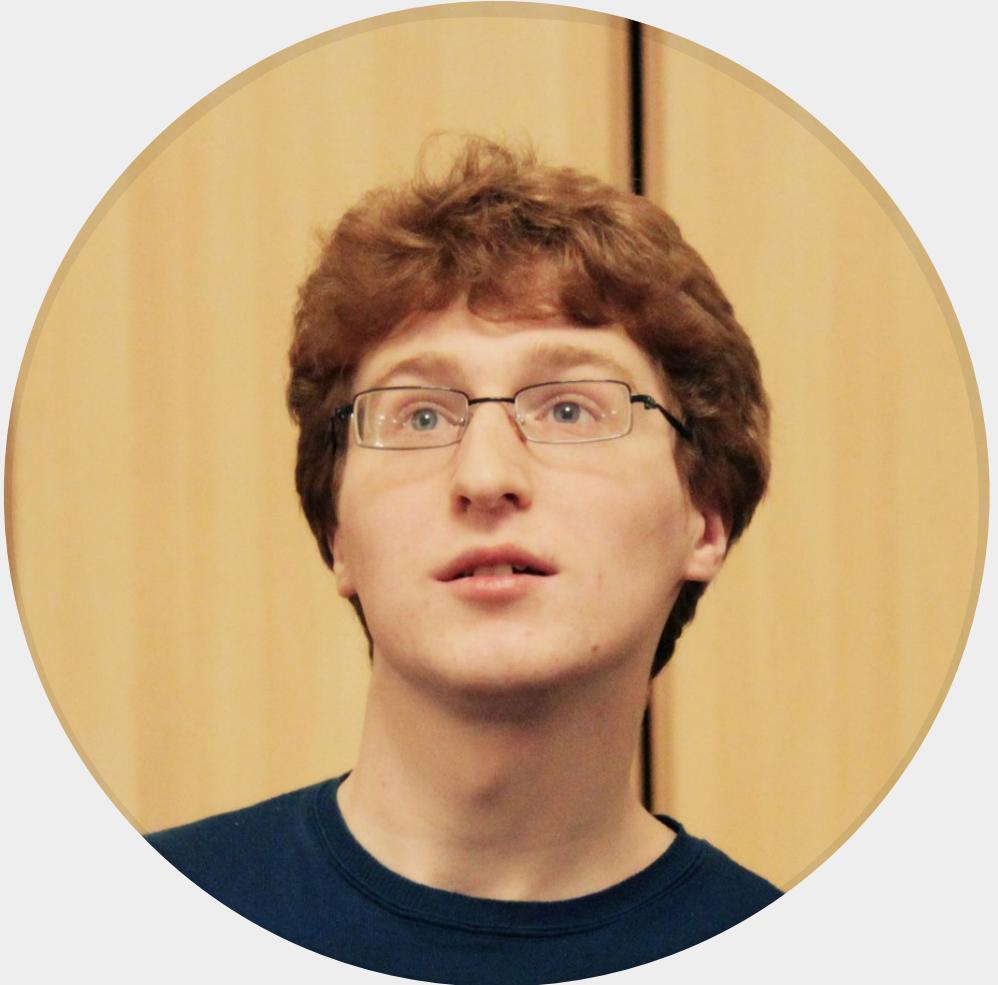


# ЗАНЯТИЕ 1.4

# БАЗОВЫЕ АЛГОРИТМЫ

# МАШИННОГО ОБУЧЕНИЯ В

# SKLEARN



# НИКИТА КУЗНЕЦОВ

Сбербанк, NLP & dev



[oychorange@gmail.com](mailto:oychorange@gmail.com)



[@NikitaKuznetsov](https://t.me/NikitaKuznetsov)

---

# ЦЕЛИ ЗАНЯТИЯ

---

# В КОНЦЕ ЗАНЯТИЯ ВЫ НАУЧИТЕСЬ:

- **решать основные задачи машинного обучения** при помощи реализованных в sklearn методах
- **оценивать качество** решения
- **предобрабатывать данные** и **подбирать параметры** моделей для улучшения качества решения

---

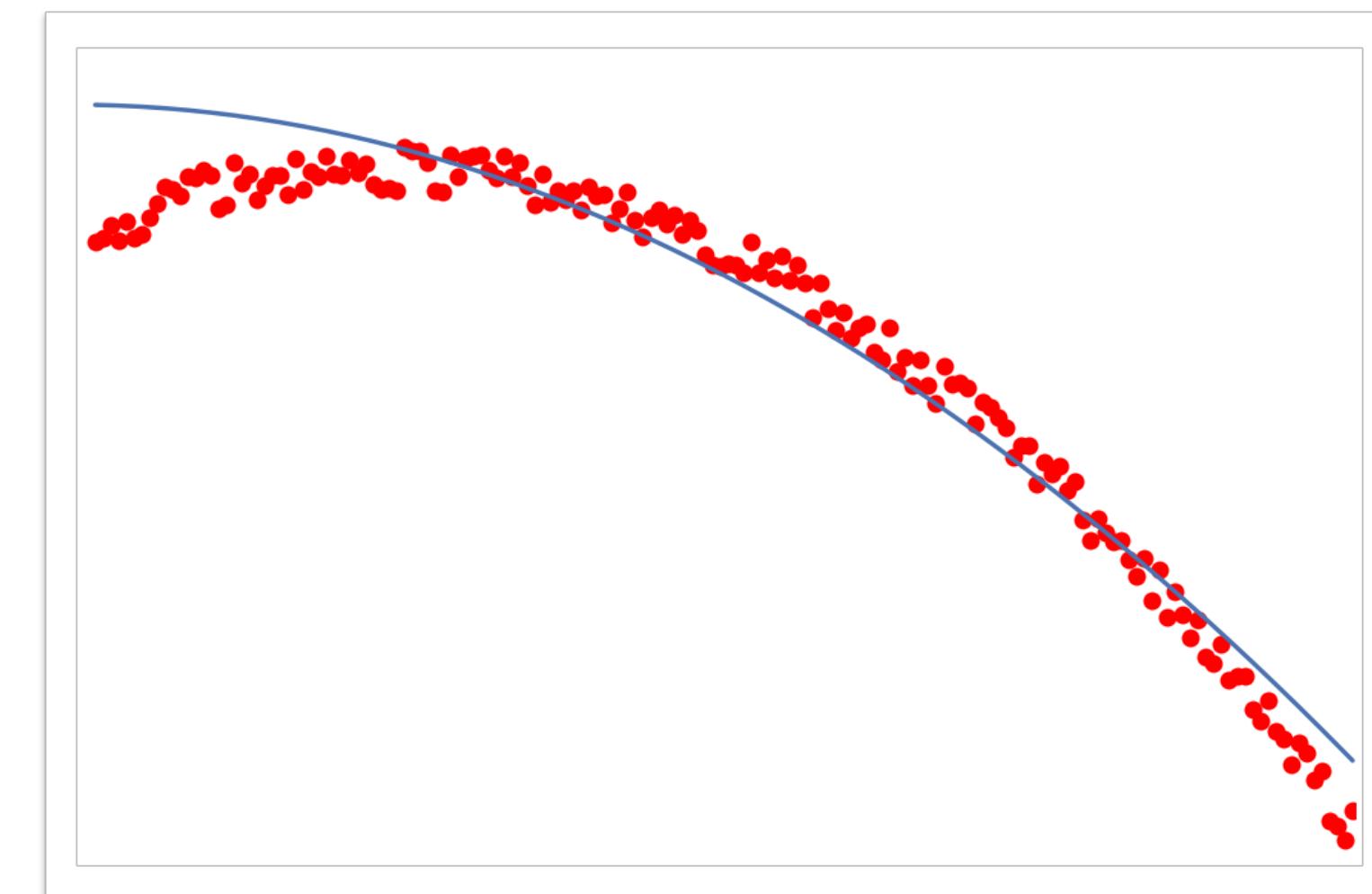
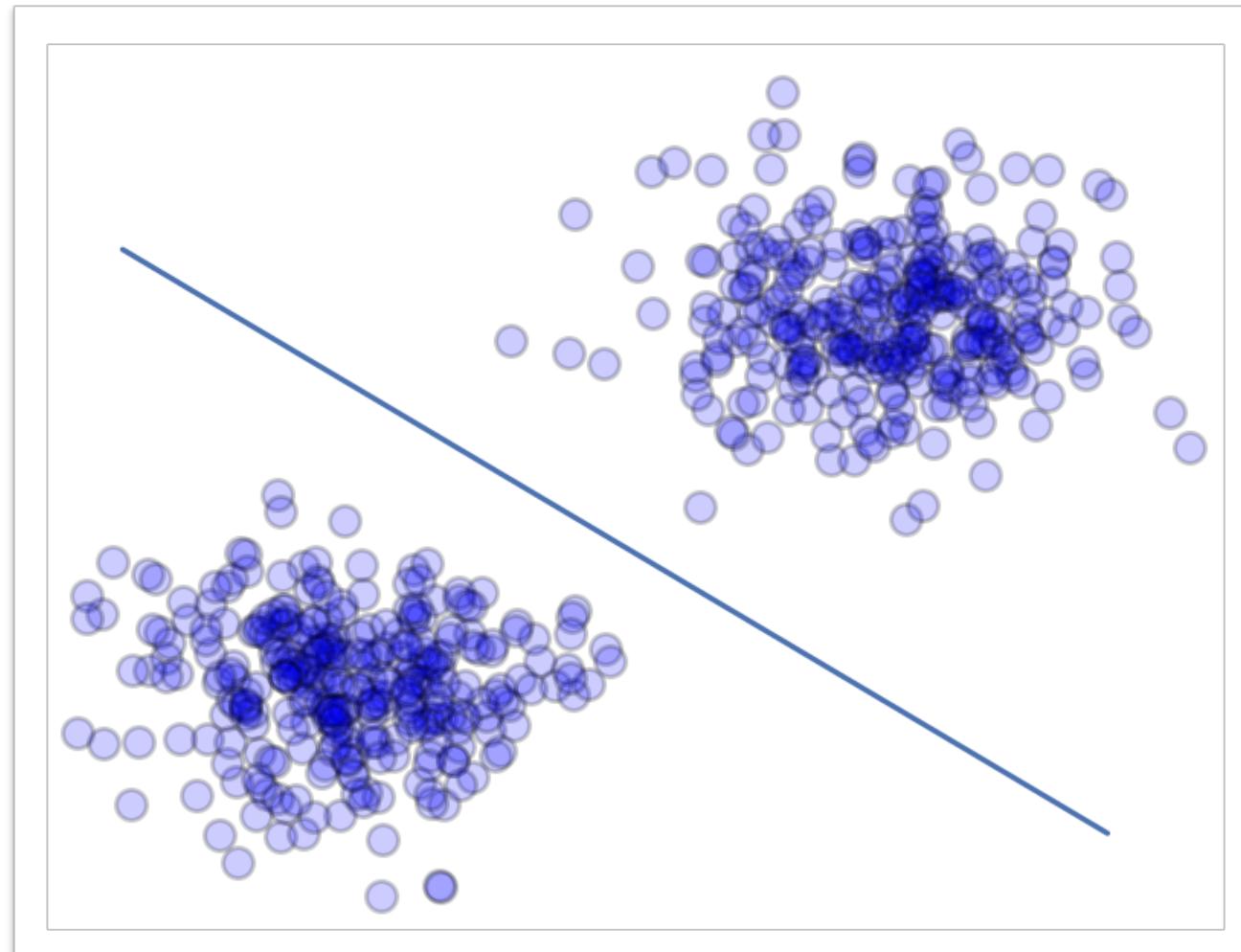
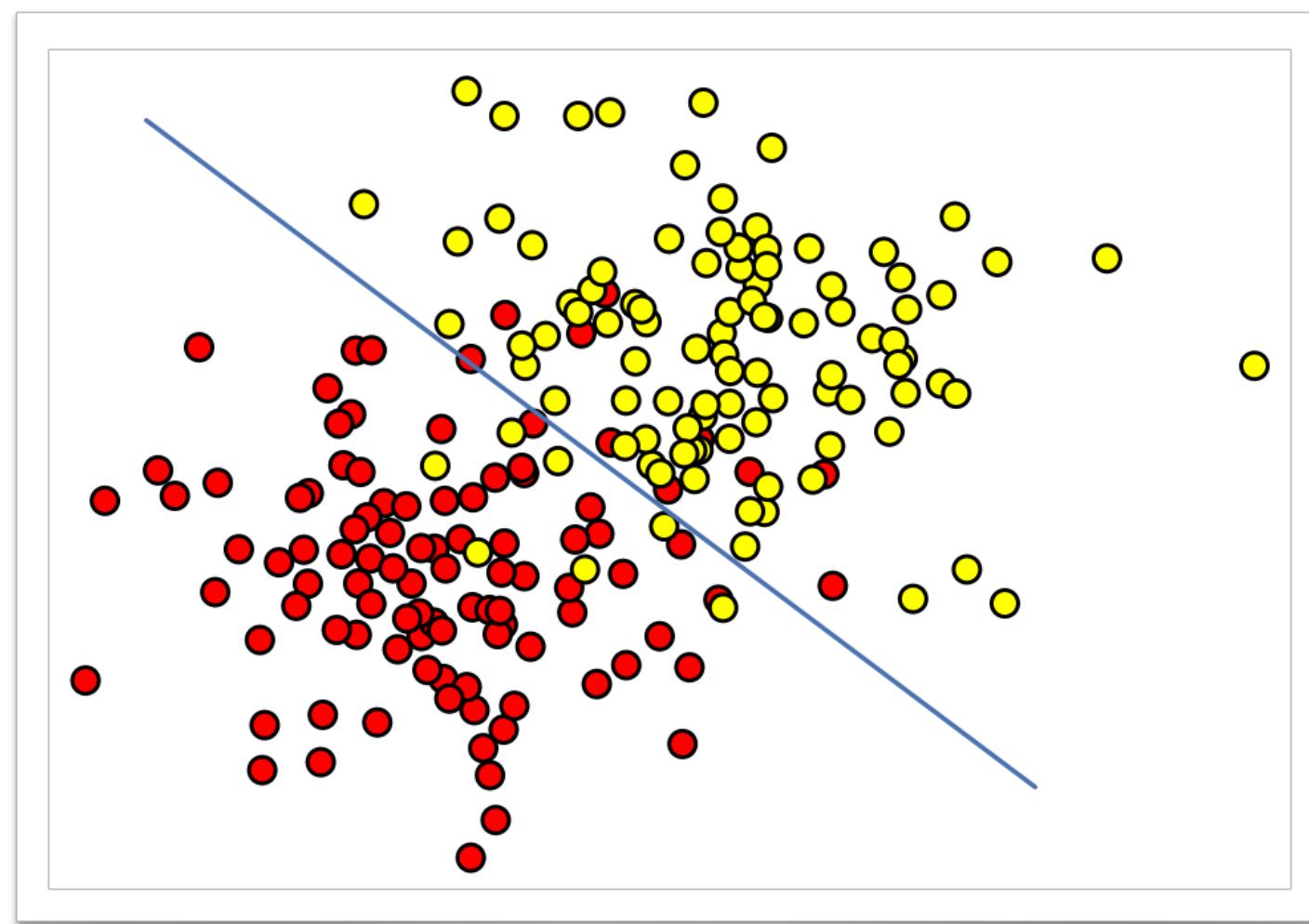
О ЧЁМ ПОГОВОРИМ И ЧТО  
СДЕЛАЕМ

- 
1. Вспомним **типы задач**, решаемые ML
  2. Обзорно познакомимся с **различными методами**, реализованными в sklearn
  3. **На практике** используем несколько из них
  4. Разберёмся, как можно **улучшить качество** решения при помощи sklearn
  5. Отработаем это **на практике**

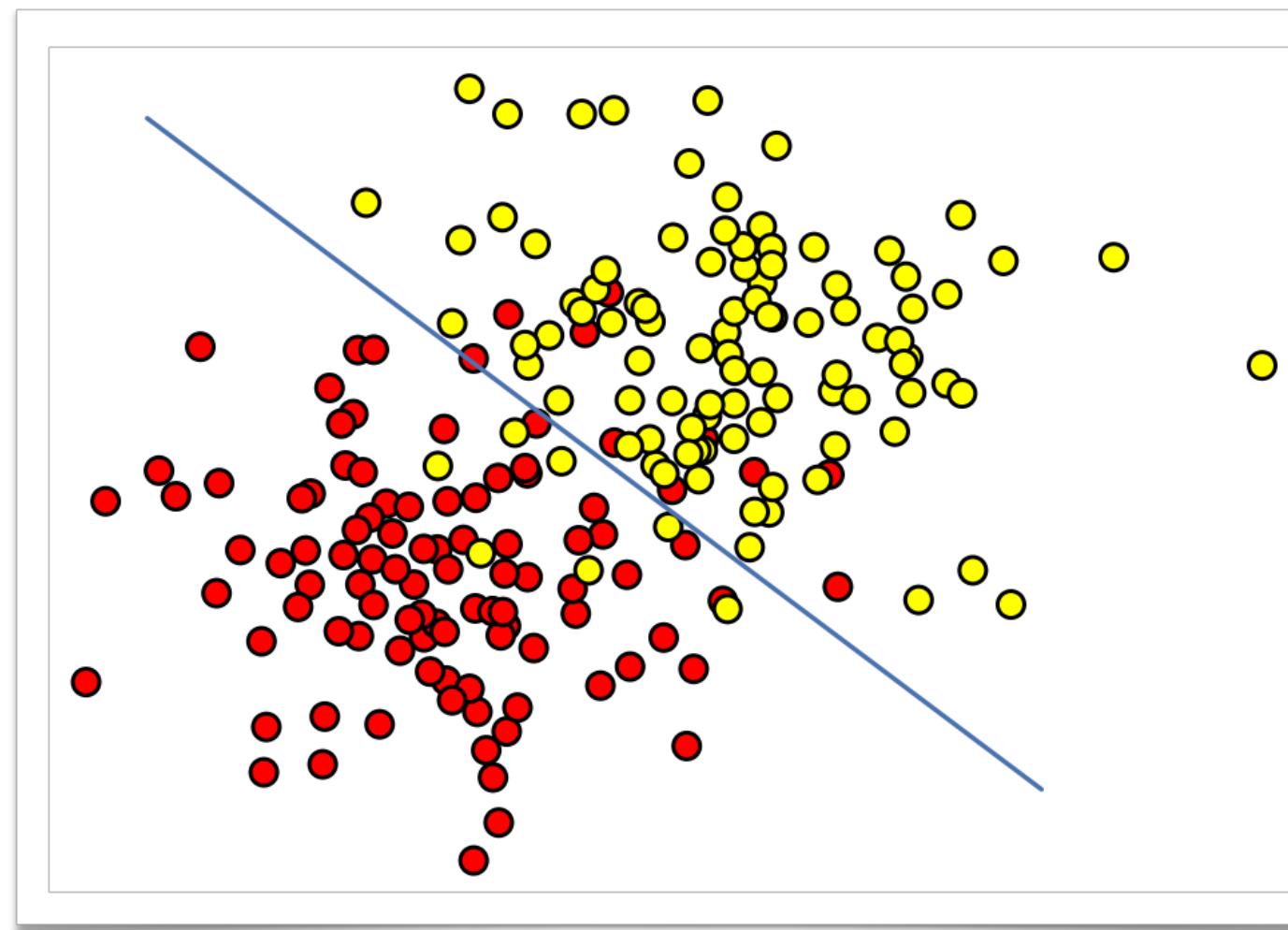
---

# 1. БИБЛИОТЕКА SCIKIT-LEARN

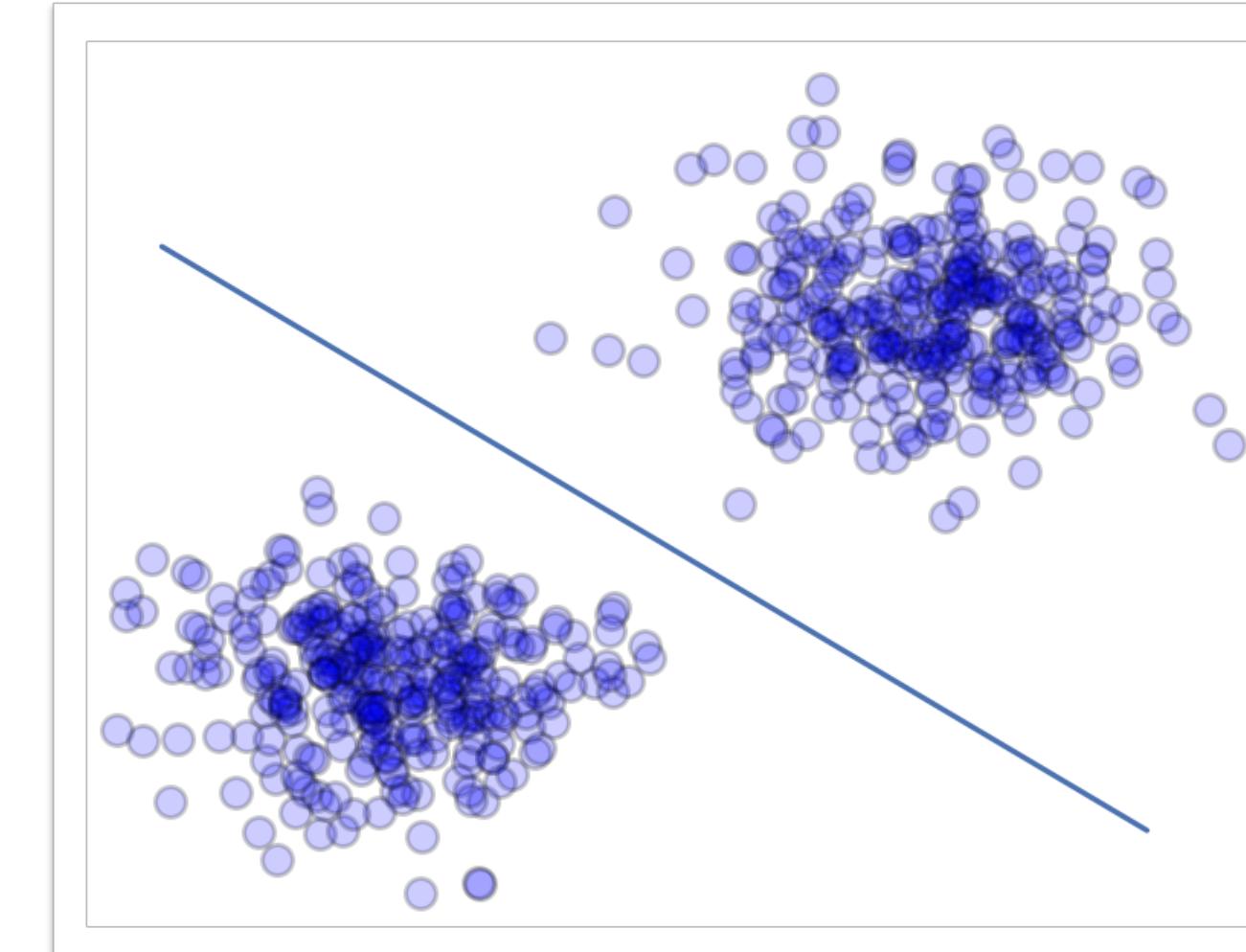
# ТИПЫ ЗАДАЧ



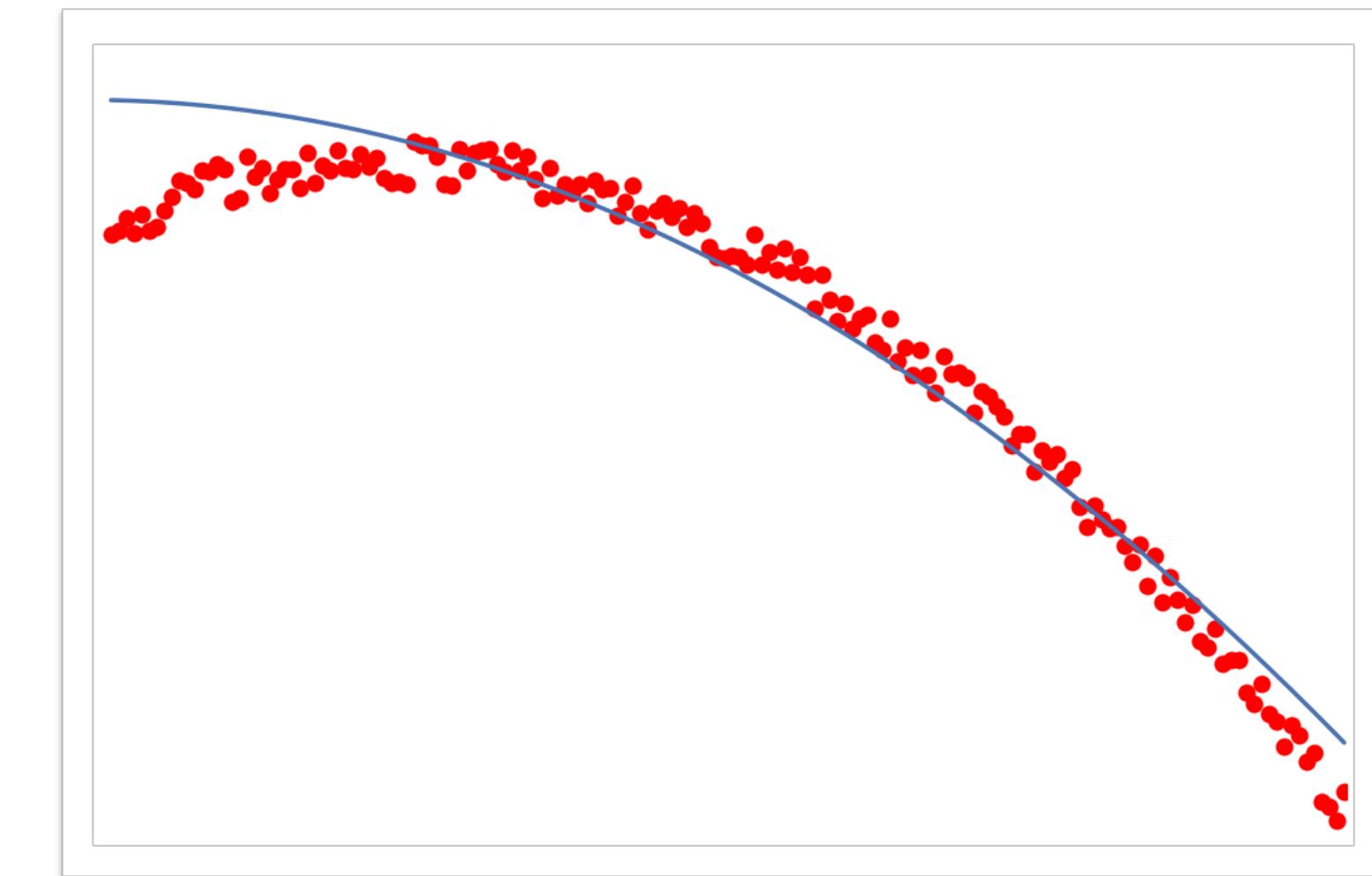
# ТИПЫ ЗАДАЧ



**классификация**  
есть разметка: X, y

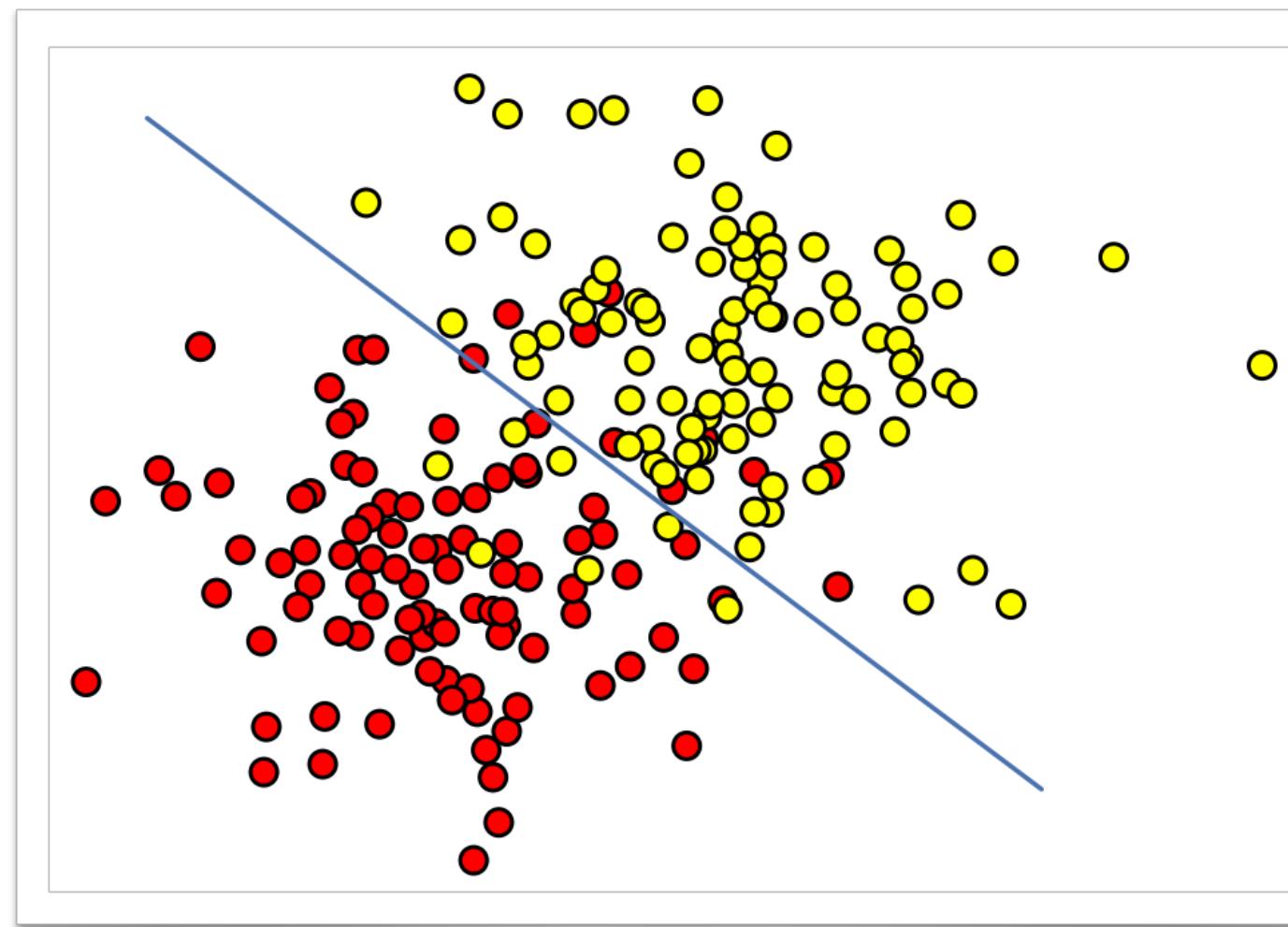


**кластеризация**  
нет разметки: X



**регрессия**  
есть разметка: X, y

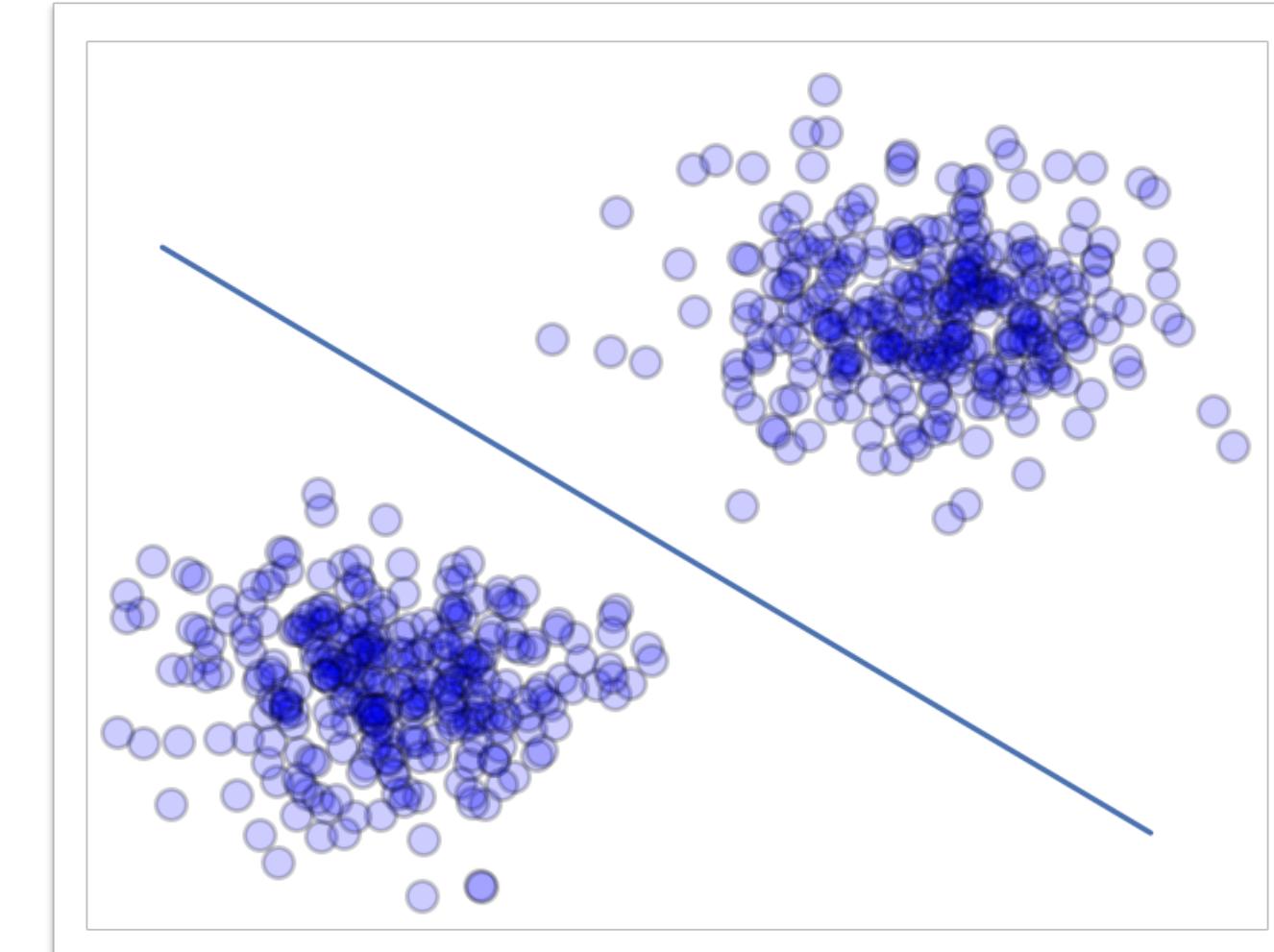
# ТИПЫ ЗАДАЧ



**классификация**

есть разметка: X, y

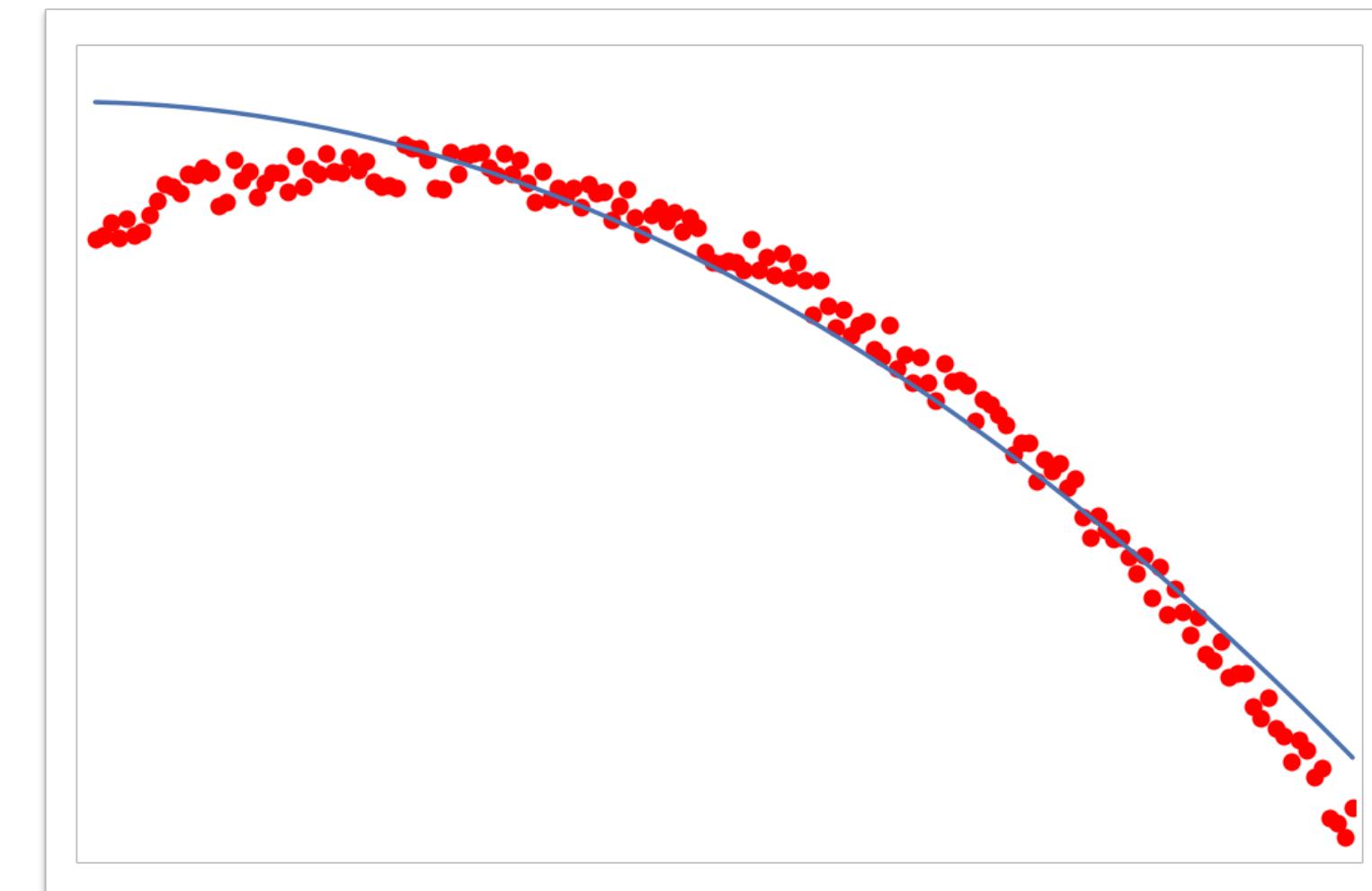
y - перечислимо



**кластеризация**

нет разметки: X

y - перечислимо

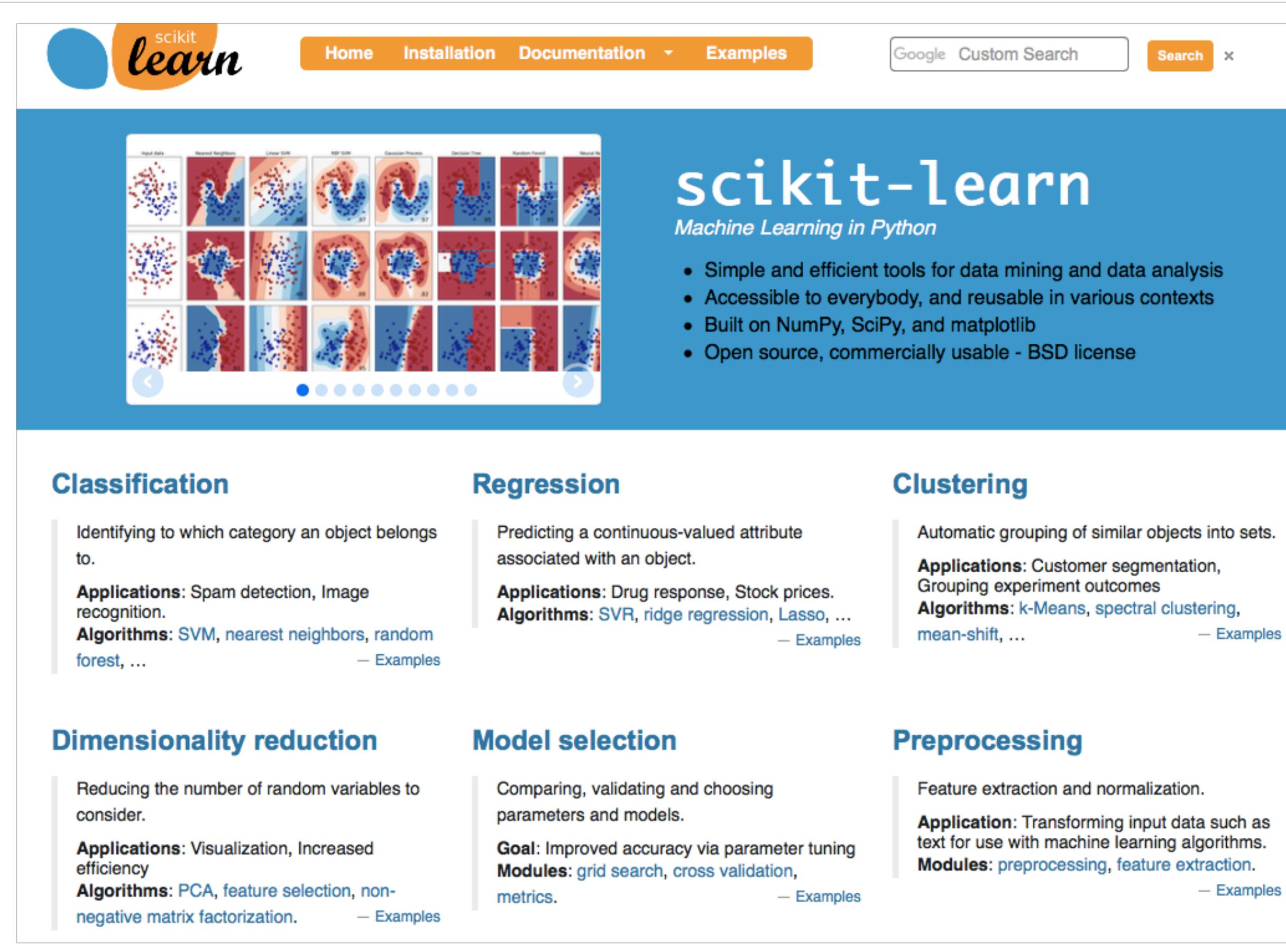


**регрессия**

есть разметка: X, y

y в непрерывном диапазоне

# SKLEARN -

The screenshot shows the official scikit-learn website. At the top, there's a navigation bar with links for Home, Installation, Documentation, Examples, and a search bar. Below the header is a banner featuring a grid of 25 small plots illustrating various machine learning models like SVM, KNN, and Random Forest. The main title "scikit-learn" is prominently displayed, followed by the subtitle "Machine Learning in Python". A bulleted list describes the library's features: simple tools for data mining and analysis, accessibility, reuseability, and a BSD license. Below this, six main categories are listed: Classification, Regression, Clustering, Dimensionality reduction, Model selection, and Preprocessing. Each category has a brief description, applications, algorithms, and examples.

Набор логически разделённых модулей

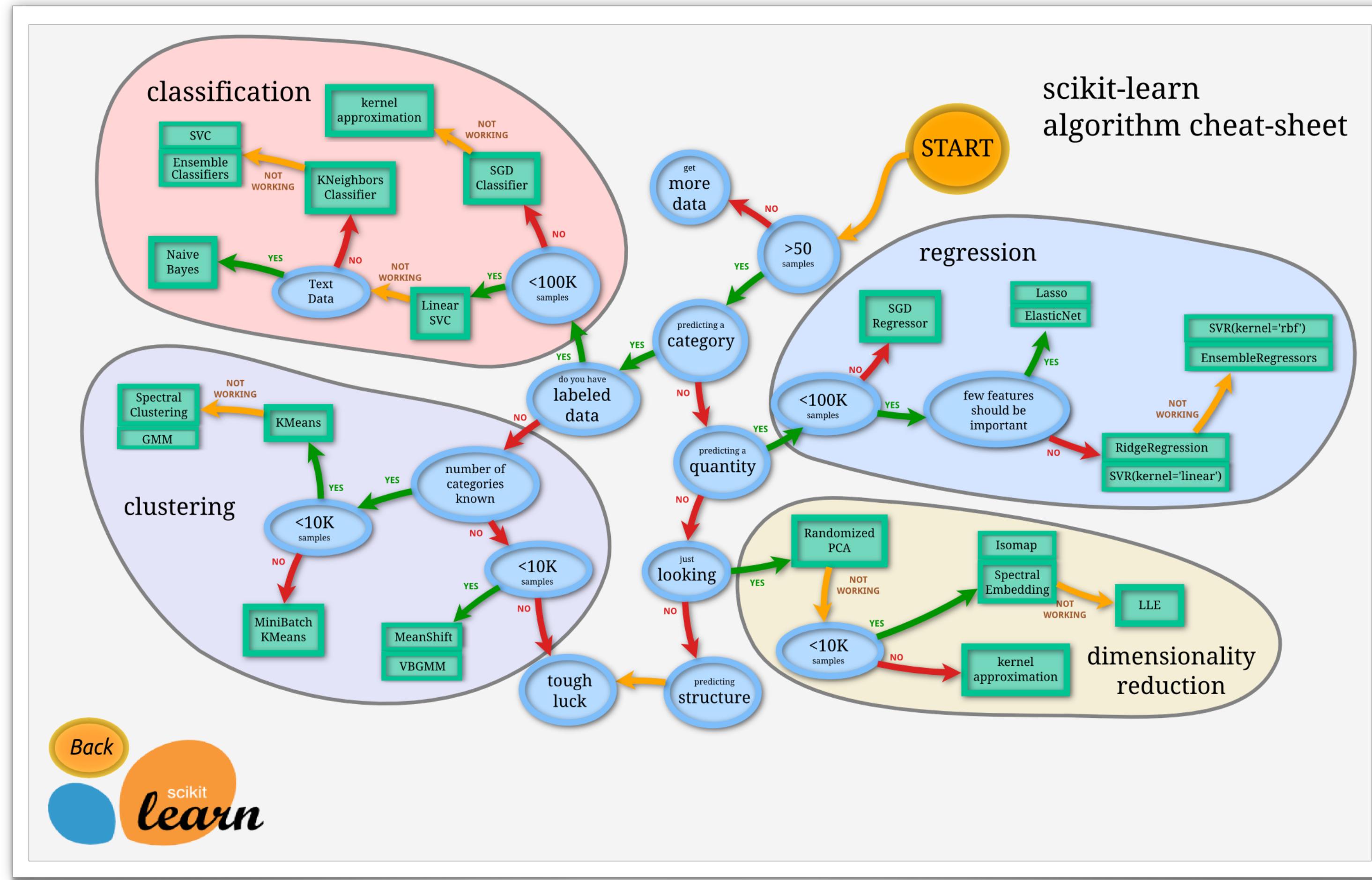
Единообразный API  
взаимодействия  
**fit + transform + predict**

Отличная документация с  
примерами и описанием  
работы алгоритмов

## ЧТО ЕЩЁ НАДО ЗНАТЬ?

- ▶ Обученные модели **можно сохранять**
- ▶ Для обучения данные должны содержаться **целиком в оперативной памяти**
- ▶ Внутри python + cython,  
через pycharm, например, можно посмотреть, что внутри :)
- ▶ Для работы необходимы **numpy / pandas**

# SKLEARN ALGO CHEATSHEET



---

## 2. МОДЕЛИ МАШИННОГО ОБУЧЕНИЯ SCIKIT-LEARN

# МОДЕЛИ МАШИННОГО ОБУЧЕНИЯ

**linear\_model** - линейные модели

- ▶ [LinearRegression](#)
- ▶ [LogisticRegression](#)

# МОДЕЛИ МАШИННОГО ОБУЧЕНИЯ

**tree** - дерево решений

- ▶ [DecisionTreeClassifier](#)
- ▶ [DecisionTreeRegressor](#)

**ensemble** - ансамбли решений: лес, бустинг

- ▶ [RandomForestClassifier](#)
- ▶ [GradientBoostingClassifier](#)

# МОДЕЛИ МАШИННОГО ОБУЧЕНИЯ

**cluster** - различные методы кластеризации

- ▶ [KMeans](#), [MiniBatchKMeans](#)
- ▶ [DBSCAN](#)
- ▶ [AffinityPropagation](#)

# ИСПОЛЬЗОВАНИЕ МЕТОДОВ ML

```
from sklearn.linear_model import LinearRegression  
X, y = ...
```

1. model = LinearRegression(**fit\_intercept=True**)
2. model.**fit**(X, y)
3. a = model.**predict**(X)  
*(если это классификация, то есть также и predict\_proba)*

оценка  $a$  должна приближать  $y$

---

### 3. ПРАКТИЧЕСКОЕ ЗАДАНИЕ - 1

# ПРАКТИЧЕСКОЕ ЗАДАНИЕ - 1

1. Загрузить данные по недвижимости Бостона
2. Разделить их на 2 части: обучающую и тестовую выборки
3. Сделать предсказание по тестовой выборке
4. Сравнить реальные значения с предсказанием

---

## 4. ДРУГИЕ ПОЛЕЗНЫЕ ФУНКЦИИ SCIKIT-LEARN

# ОЦЕНКА КАЧЕСТВА

**metrics** - различные метрики качества решений

- ▶ [classification\\_report](#)
- ▶ [mean\\_squared\\_error](#)

# ПОДБОР ПАРАМЕТРОВ МОДЕЛИ

**model\_selection** - оценка качества + подбор гиперпараметров моделей

- ▶ [GridSearchCV](#)
- ▶ [cross\\_val\\_score](#)

# ПРЕДОБРАБОТКА ДАННЫХ

**preprocessing** - нормировка

- ▶ [StandardScaler](#)

**feature\_extraction.text** - векторизация

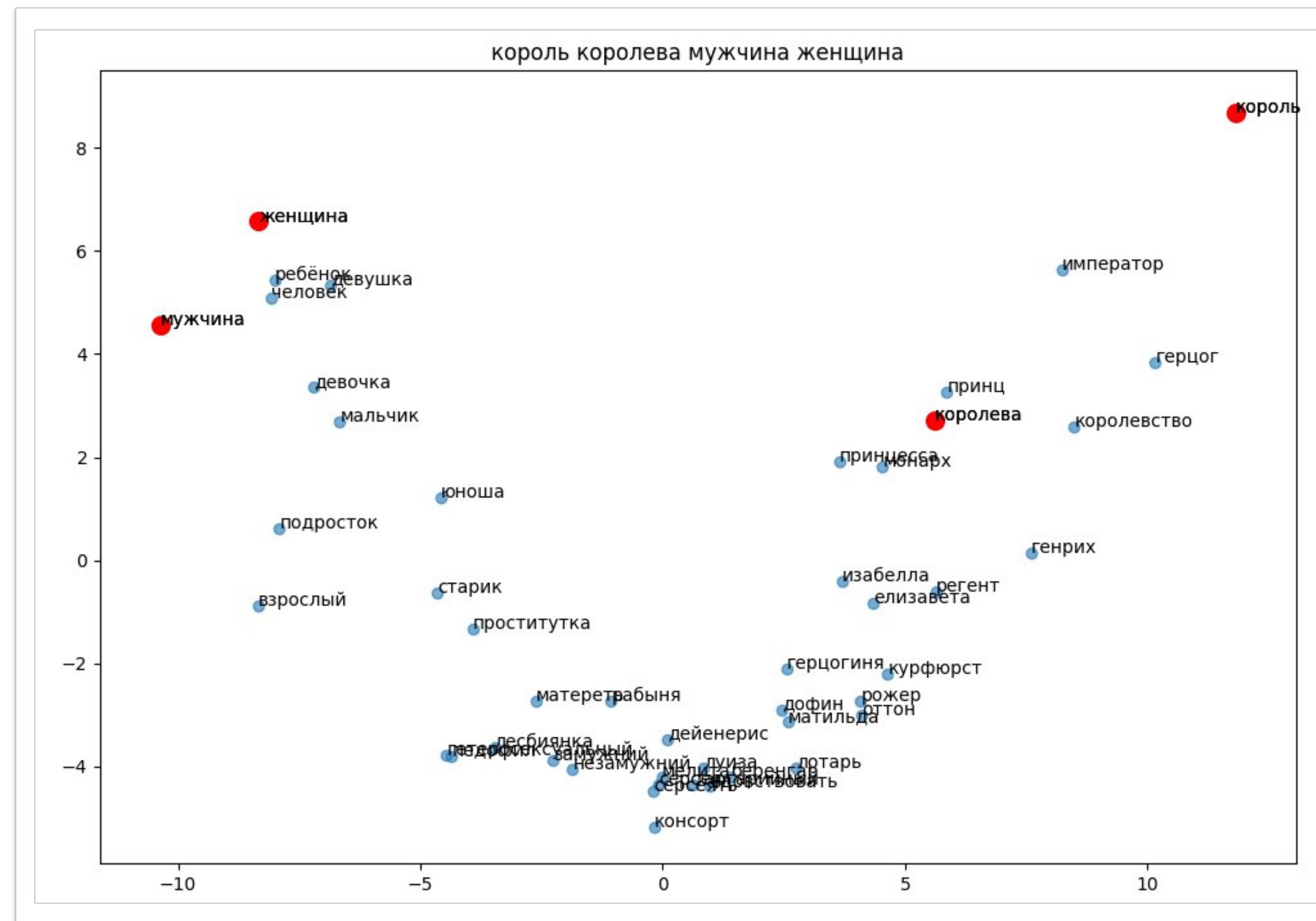
- ▶ [CountVectorizer](#)
- ▶ [TfidfVectorizer](#)
- ▶ [HashingVectorizer](#)

# СНИЖЕНИЕ РАЗМЕРНОСТИ

**decomposition** - разложение матриц и снижение размерности

- ▶ [PCA](#)
- ▶ [TruncatedSVD](#)

# СНИЖЕНИЕ РАЗМЕРНОСТИ



Пример:

PCA по векторам word2vec  
для визуализации:  
из 300-мерного пространства  
в 2-мерное

---

## 5. ПРАКТИЧЕСКОЕ ЗАДАНИЕ - 2

# ПРАКТИЧЕСКОЕ ЗАДАНИЕ - 2

1. Взять данные со соревнования [Титаник](#)
2. Перевести всё в числовой вид
3. Заполнить пропуски и отсортировать данные
4. При помощи кросс-валидации найти оптимальный параметр для логистической регрессии
5. Лучшей выбранной моделью оценить качество на отложенной выборке
6. Сделать предсказание по тестовой выборке

---

ЧТО МЫ СЕГОДНЯ УЗНАЛИ

## ЧТО МЫ СЕГОДНЯ УЗНАЛИ

---

1. Scikit-learn - open-source библиотека для решения задач машинного обучения, содержащая различные методы решения со схожим набором методов для работы
2. Также в ней содержится набор методов для предобработки выборки, подбора гиперпараметров модели и оценки качества построенного решения
3. Библиотека имеет хорошую документацию и удобна в использовании

---

# ПОЛЕЗНЫЕ МАТЕРИАЛЫ

## ПОЛЕЗНЫЕ МАТЕРИАЛЫ

---

1. [Документация sklearn](#)
2. [Sklearn cheatsheet](#)



НЕТОЛОГИЯ  
групп

# Спасибо за внимание!

## НИКИТА КУЗНЕЦОВ



[ouchorange@gmail.com](mailto:ouchorange@gmail.com)



[@NikitaKuznetsov](https://t.me/NikitaKuznetsov)