

Блок

Построение модели

Цели блока

- Понимать задачи ML
- Знание классических алгоритмов
- Оценка качества обучения
- Улучшение разработанных моделей

- 10 занятий
- 1 лабораторная работа
- Домашние задания
- Самостоятельное изучение

Структура блока

ROADMAP блока

1. Библиотека `sklearn`
2. Алгоритмы классификации: линейные методы, логистическая регрессия и SVM
3. Алгоритмы классификации: деревья решений
4. Алгоритмы регрессии: линейная и полиноминальная
5. Алгоритмы кластеризации
6. Ансамблирование
7. Функции потерь и оптимизация
8. Оценка точности модели, переобучение, регуляризация
9. Улучшение качества модели

Кто я?



Артур Сапрыкин

- ML/DL, NLP
- Backend
- Программный архитектор



asaprykin92@gmail.com



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

Занятие № 1

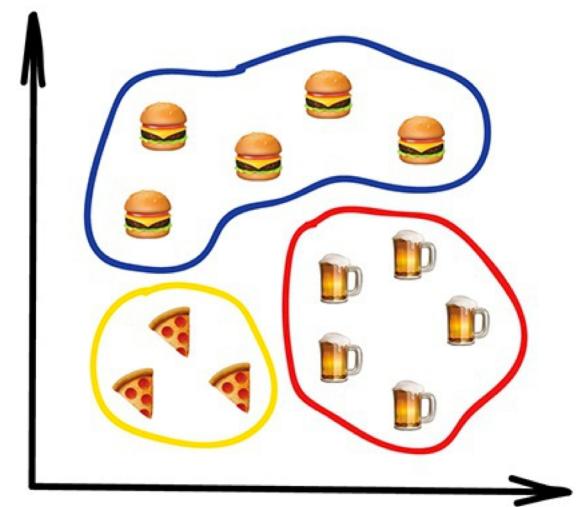
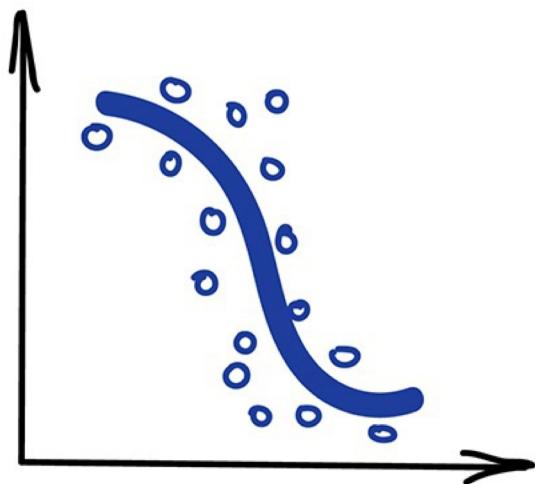
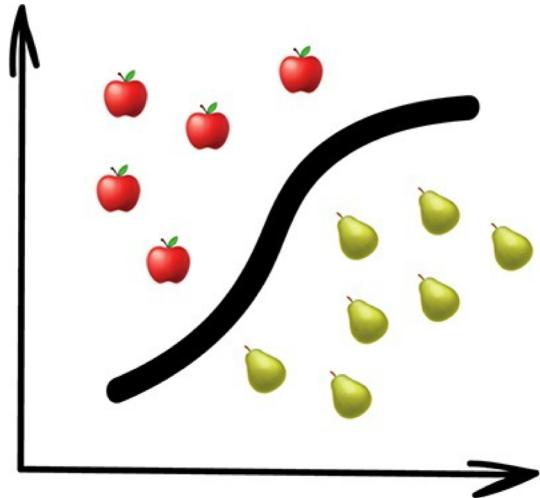
Библиотека Sklearn

- Узнать, для чего нужна библиотека sklearn
- Уточнить основные задачи ML
- Вспомнить основные функции библиотеки на практическом примере

Цели занятия

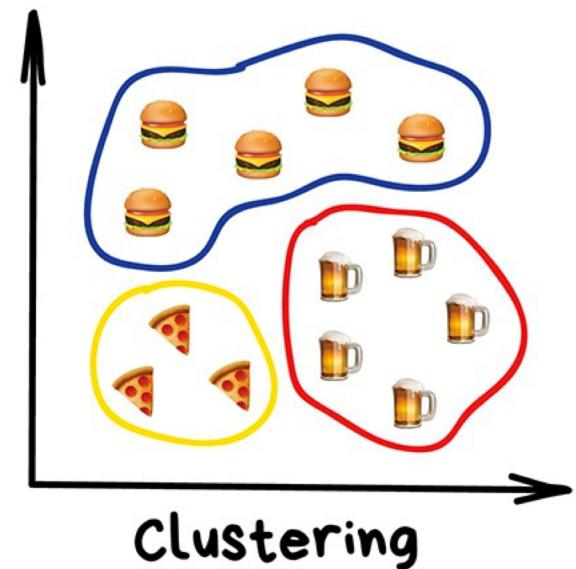
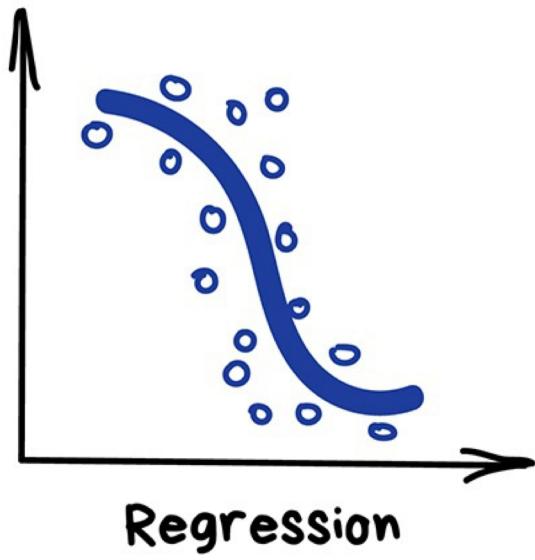
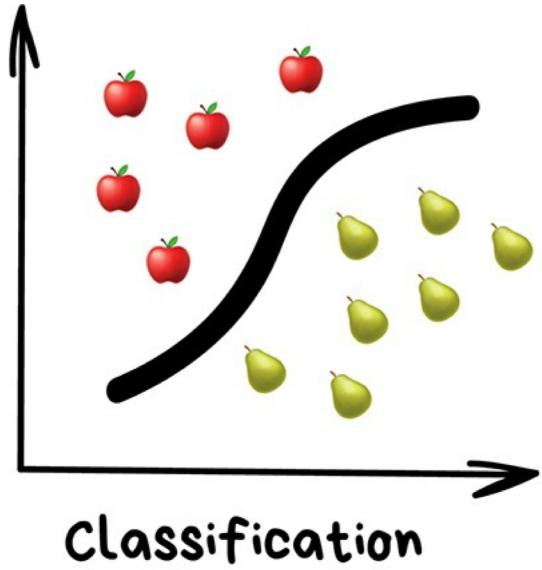
Введение

Введение



Источник: https://vas3k.ru/blog/machine_learning/

Введение



[Источник: https://vas3k.ru/blog/machine_learning/](https://vas3k.ru/blog/machine_learning/)

Почему мы любим Sklearn?

1. Огромный набор инструментов для создания моделей на основе машинного обучения
2. Качественная документация
3. Высокая скорость работы
4. Единообразный API взаимодействия

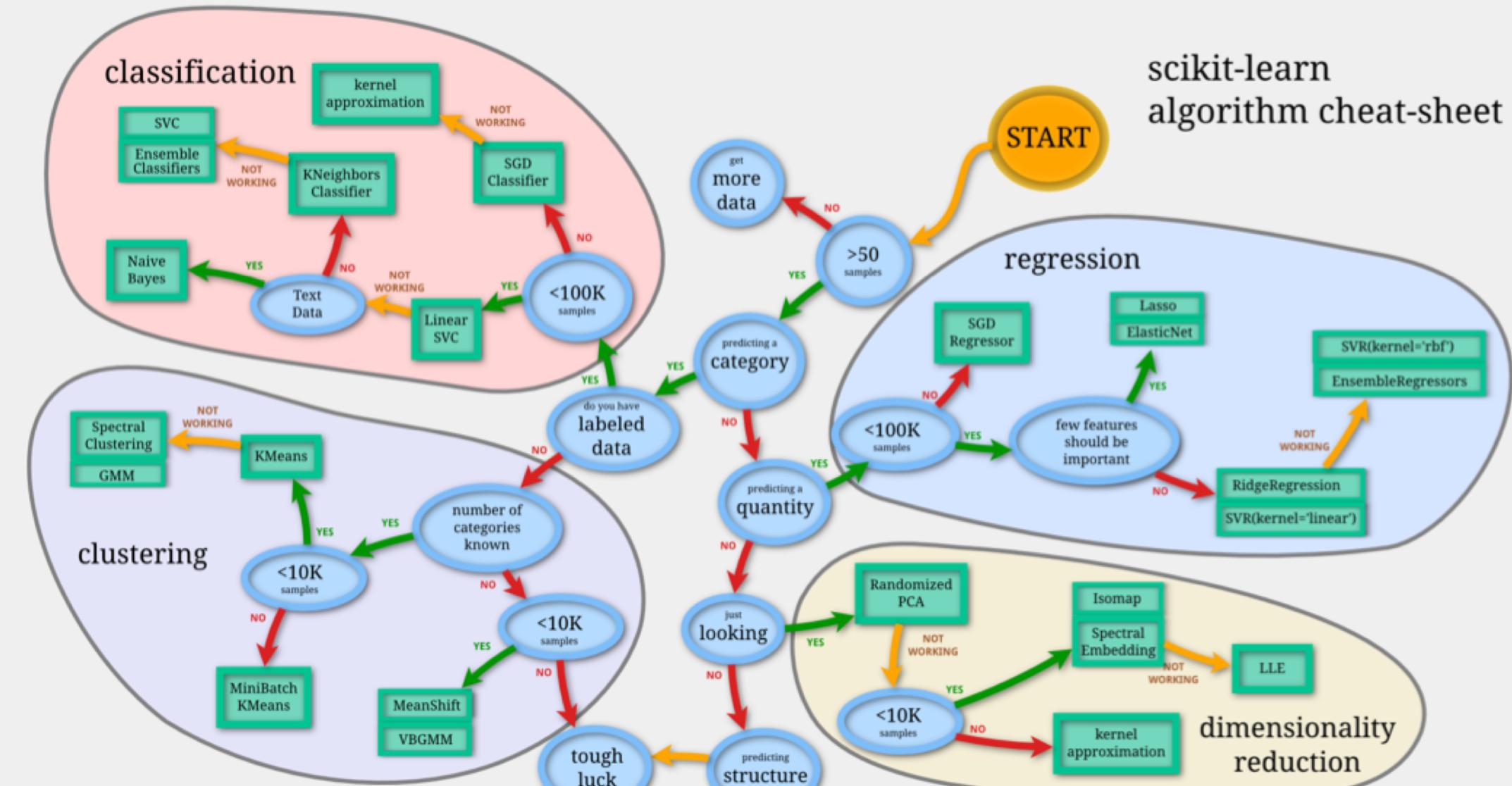


<https://scikit-learn.org/stable/modules/classes.html>

Важно знать

1. Обученные модели **можно НУЖНО** сохранять
2. Чтобы обучать модели у исследователя **должно быть достаточно оперативной памяти**. Все данные для обучения модели должны храниться там.
3. Разработан на **Python + Cython**
4. Для работы необходимо иметь **numpy и pandas**

scikit-learn algorithm cheat-sheet



Back

scikit
learn

https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

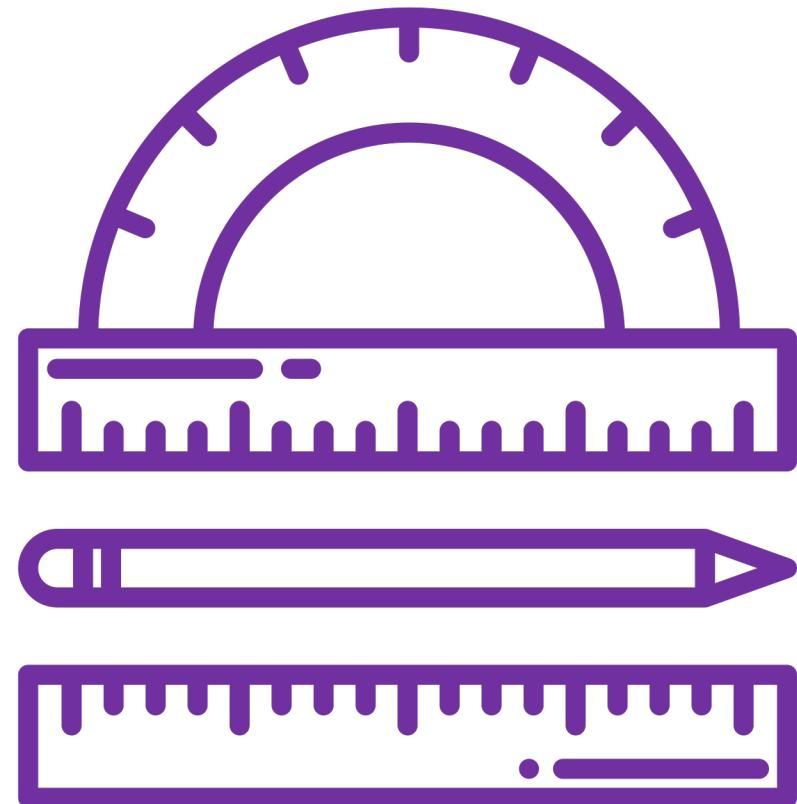
Что внутри коробки?

Алгоритмы ML

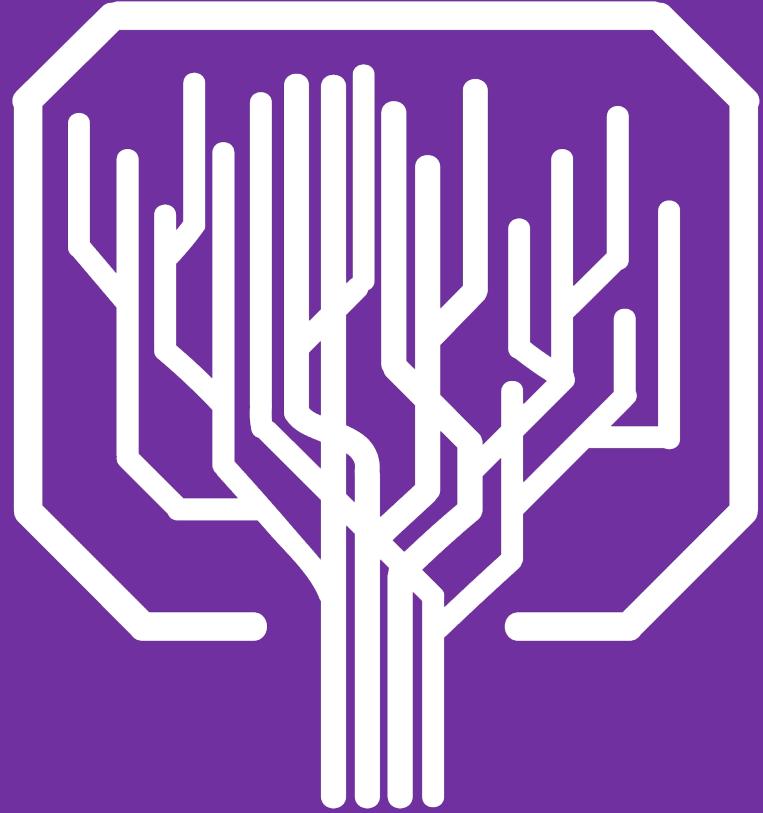
Модуль: [linear_model](#)

Что внутри? Линейные модели

- LinearRegression
- LogisticRegression



Алгоритмы ML



Модуль: tree

Что внутри? Деревья решений

- DecisionTreeClassifier
 - DecisionTreeRegressor
-

Модуль: ensemble

Что внутри? Ансамбли деревьев

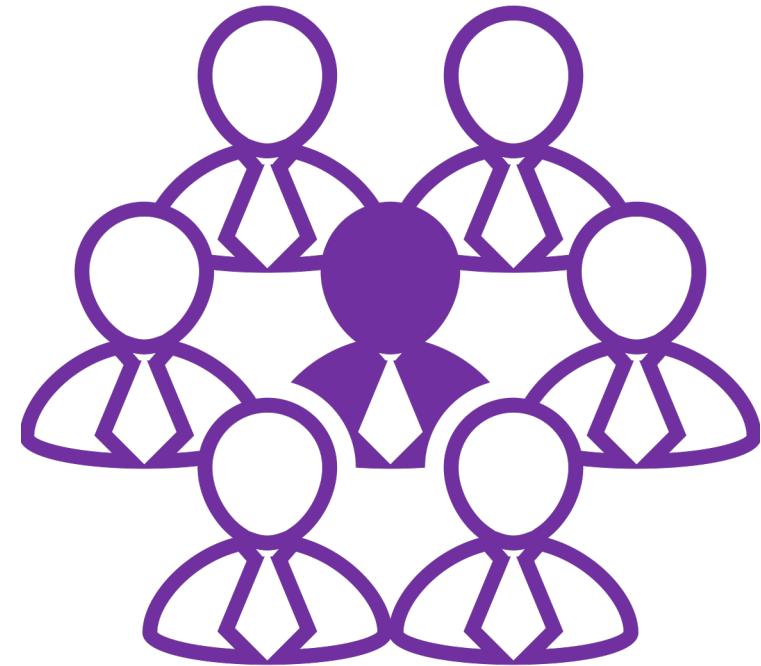
- RandomForestClassifier
- GradientBoostingClassifier

Алгоритмы ML

Модуль: cluster

Что внутри? Алгоритмы кластеризации

- KMeans, MiniBatchKMeans
- DBSCAN
- AffinityPropagation



Как использовать?

```
from sklearn.linear_model import LinearRegression
```

X, y = КАКИЕ-ТО ДАННЫЕ

```
model = LinearRegression(fit_intercept=True)
```

```
model.fit(X, y)
```

```
a = model.predict(X)
```

(если это классификация, то есть также и predict_proba)

оценка a должна приближать y

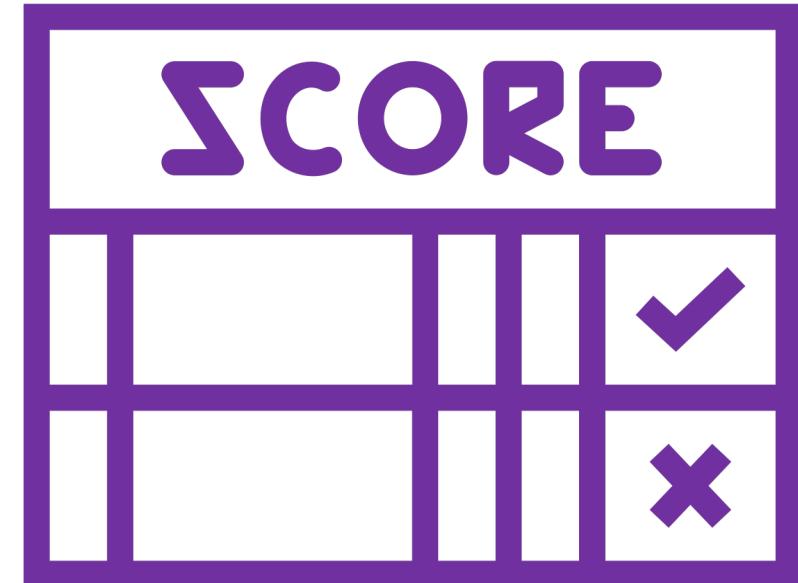
ПРАКТИЧЕСКОЕ ЗАДАНИЕ

Модули Sklearn

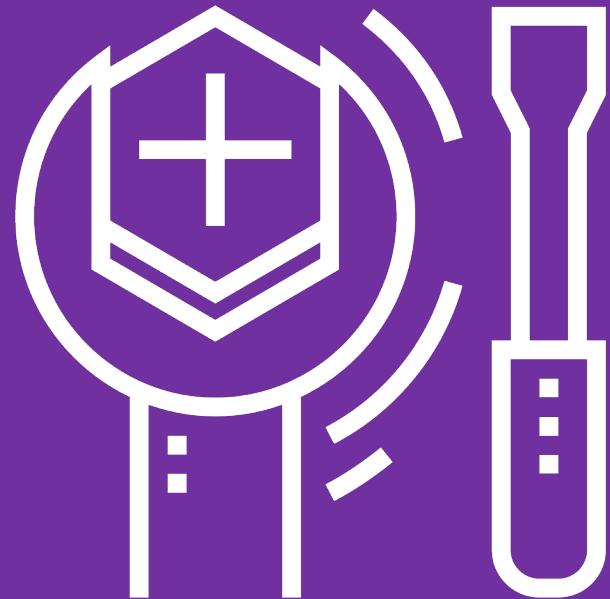
Модуль: metrics

Что внутри? Оценка качества алгоритмов

- classification_report
- mean_squared_error



Модули Sklearn



Модуль: model_selection

Что внутри? Подбор параметров

- GridSearchCV
- cross_val_score

Модули Sklearn

Модуль: preprocessing

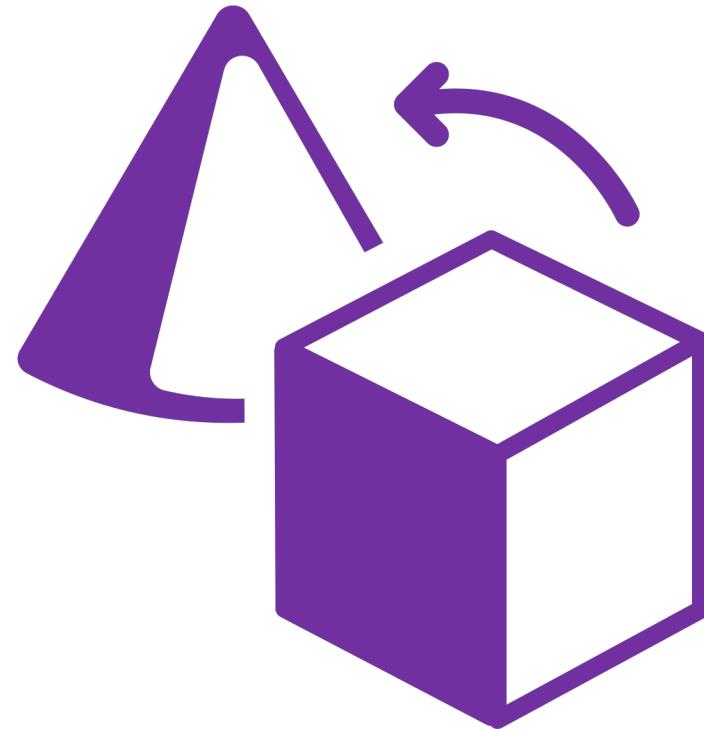
Что внутри? Предобработка данных

- StandardScaler
-

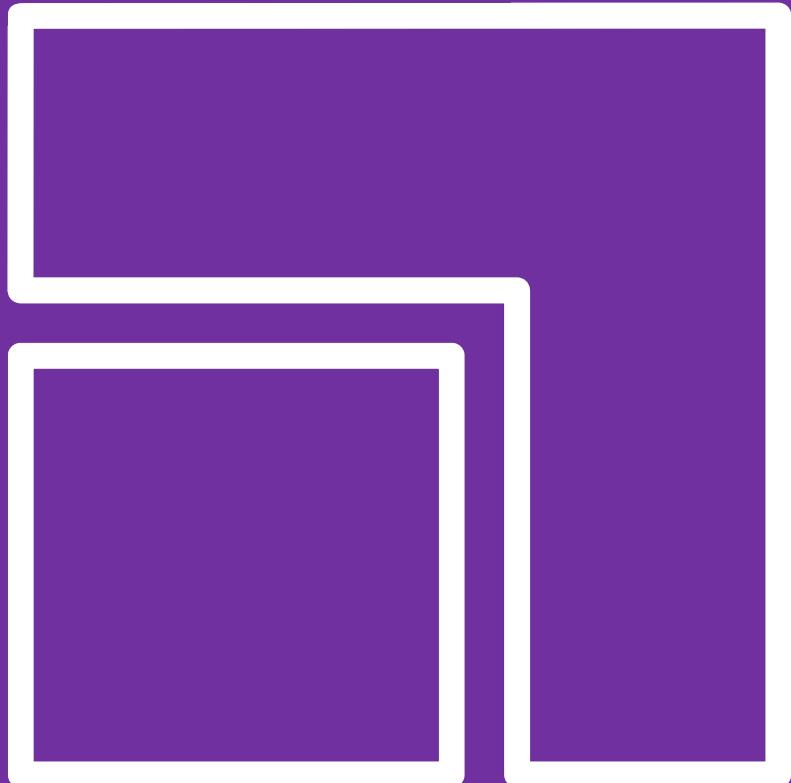
Модуль: **feature_extraction.text**

Что внутри? Векторизация текста

- CountVectorizer
- TfidfVectorizer



Модули Sklearn



Модуль: decomposition

Что внутри? Снижение размерности

- PCA
- TruncatedSVD

ПРАКТИЧЕСКОЕ ЗАДАНИЕ

ВОПРОСЫ