



Машинное обучение

[Виктор Владимирович Китов](#)

Навигация по темам учебника доступна слева вверху

(три палочки на мобильных устройствах).

Почта для обратной связи: deepmachinelearning@yandex.ru.

[Условные обозначения](#) учебника.

[Лицензия](#) на использование материалов.

Вы можете помочь:

- Напишите, если учебник помог вам разобраться в какой-то теме, живой отклик всегда ценен!
- Расскажите об учебнике своим друзьям и коллегам по работе.
- Напишите обратную связь по материалам учебника.
- Напишите, если заметите опечатки и ошибки (даже незначительные) в тексте или работе сайта.

[Следующая страница](#)

[Введение »](#)



Машинное обучение

[Виктор Владимирович Китов](#)

Навигация по темам учебника доступна слева вверху

(три палочки на мобильных устройствах).

Почта для обратной связи: deepmachinelearning@yandex.ru.

[Условные обозначения](#) учебника.

[Лицензия](#) на использование материалов.

Вы можете помочь:

- Напишите, если учебник помог вам разобраться в какой-то теме, живой отклик всегда ценен!
- Расскажите об учебнике своим друзьям и коллегам по работе.
- Напишите обратную связь по материалам учебника.
- Напишите, если заметите опечатки и ошибки (даже незначительные) в тексте или работе сайта.

[Следующая страница](#)

[Введение »](#)



Посвящается моим родителям Владимиру и Ольге.

Введение в машинное обучение

Неформальное определение

В повседневной жизни все мы ежедневно сталкиваемся с принятием решений:

- при текущей дорожной ситуации надо ли ехать на метро или на машине?
- удалить ли письмо в электронной почте как спам или сохранить?
- ожидается ли дождь, и стоит ли брать с собой зонт?
- стоит ли звонить человеку с определённым предложением или он от него, скорее всего, откажется?
- стоит ли докупить хлеба и молока или их хватит до конца недели?
- какое фокусное расстояние на фотоаппарате установить, чтобы лицо фотографируемого человека получилось чётким?

Аналогичные проблемы принятия решений решаются организациями в массовом порядке:

- сколько хлеба и молока закупить магазину, чтобы удовлетворить спрос до конца недели?
- как почтовому сервису автоматически разделять письма на полезные и спам?
- какими способами и по каким маршрутам отправлять грузы?
- какую погоду предсказать сервису прогноза погоды на оставшийся конец дня?
- как автоматически устанавливать фокусное расстояние на производимых фотоаппаратах?

При массовом и повторяющемся принятии решений целесообразно процесс принятия этих решений автоматизировать. Можно разработать явную систему правил этого процесса. Например, при определении важности письма мы можем смотреть на то, переписывались ли мы ранее с отправителем, принадлежит ли отправитель надёжной и известной компании, включает ли текст письма определённые ключевые слова, которые нам заранее не интересны? В этом случае мы как бы явно программируем алгоритм принятия решений. Но проблема заключается в том, что

- Сложно разработать универсальный алгоритм, который бы подходил всем пользователям. Одних может не интересовать получение кредита или психологическая консультация, а для других это может оказаться актуальным.
- Сложно учесть всё многообразие ситуаций. Например, "бесплатная психологическая консультация" может быть сформулирована как "консультация психолога без оплаты", и изначальное правило уже перестанет действовать.

В подобных случаях полезно использовать **машинное обучение** (machine learning).

Машинное обучение - это процесс, в результате которого компьютер по наблюдаемым данным обучается лучше решать заданную задачу. Метод решения задачи при этом ищется в широком классе функций, параметризованном вектором параметров, который и подбирается по наблюдаемым данным.

Вместо явного прописывания четкой системы правил принятия решений в идеологии машинного обучения эти правила подбираются автоматически по данным. Под компьютером при этом может пониматься любое вычислительное устройство, например смартфон или процессор робота. Рассмотрим более детальное определение:

Машина учится на заданном **опыте** решать некоторую **задачу**, относительно некоторого **показателя качества**, если

показатель качества растёт на задаче после получения опыта.

В нашем примере задача - это классификация писем на спам/не спам, показатель качества - доля верно классифицированных писем, а опыт - коллекция прошлых писем, которые до этого были вручную размечены по классам.

В другом примере задачей выступает предсказание времени в пути, отталкиваясь от текущего времени суток, дня недели, погоды и загрузки дорог, показатель качества - модуль отклонения предсказанного времени от фактического, а опытом - история предыдущих передвижений в известных условиях и с известным временем в пути.

Примеры задач

Приведём примеры популярных задач, решаемых с помощью машинного обучения:

- Предсказать, уйдёт ли клиент к конкурентам? (churn prediction)
- Является ли последовательность финансовых транзакций мошеннической? (fraud detection)
- Предсказание пробок и времени в пути при планировании маршрута (traffic prediction).
- Стоит ли показывать заданный товар покупателю в качестве рекомендации? (recommender systems)
- Рекомендовать ли человека в качестве друга в социальной сети?
- Является ли аккаунт в социальной сети ботом?
- Голосовой ассистент: распознавание речи, автоматический ответ на вопросы, генерация речевого ответа.
- Идентификация человека по лицу. Распознавание номера машины на камерах.
- Подсчёт и отслеживание людей по камерам видеонаблюдения (object tracking). Обнаружение неправомерных действий (activity recognition).
- Автоматическое управление машинами (self-driving cars): распознавание ситуации, планирование маршрута.
- Автоматическая торговля на бирже (algorithmic trading).
- Перевод с одного языка на другой (machine translation).
- Постановка медицинских диагнозов по жалобам пациента и результатам обследований.
- Рекомендация веб-страниц по поисковому запросу (information retrieval).
- Автоматическая оценка ожидаемой зарплаты кандидата по резюме.
- Игра компьютера в шахматы, управление игровыми персонажами.
- Автоматическая оценка квартиры по её характеристикам.
- Хвалит или ругает пользователь товар в своём отзыве? (sentiment analysis)
- Генерация иллюстраций к тексту. Текстовое описание, что показано на изображении.
- Прогноз погоды. Рекомендации фермерам, когда сажать/поливать/удобрять посевы.
- Автоматическое написание программного кода (no code AI).
- Автоматический выбор, каким пользователям какую онлайн-рекламу показать (targeted ads).
- Генерация химических соединений, обладающих требуемыми свойствами:
 - крепкий, но легкий и термостойкий материал с повышенной проводимостью (material design)
 - препарат, обеспечивающий лечение и обладающий минимальными побочными эффектами (drug discovery)

Типы обучения

Машинное обучение (machine learning) описывает в целом подходы про подготовку данных, настройку и оценку прогнозирующих алгоритмов. Этому посвящена первая книга сайта, которую вы сейчас читаете.

Глубокое обучение (deep learning) - подраздел машинного обучения про сложные многоуровневые модели (нейросети), способные

решать более сложные задачи прогнозирования. С ростом вычислительных мощностей и объёма данных существует устойчивый тренд на замену классических алгоритмов машинного обучения на нейросетевые, обеспечивающие большую точность и возможность генерировать не только численные ответы, но и ответы в виде сложно структурированных данных, таких как текст, речь, изображение и видео. Глубокому обучению посвящена [вторая часть книги](#).

Обучение с подкреплением (reinforcement learning) - также подраздел машинного обучения, в котором строится не однократный прогноз независимо для каждого объекта, а вырабатывается интерактивная стратегия поведения в изменяемой среде.

Примером обучения с подкреплением может служить автоматическая игра в шахматы, в которой необходимо последовательно генерировать каждый следующий ход. Успех генерации определяется не только текущим ходом, но и всей последовательностью решений в течение партии. Обучение с подкреплением также применяется в управлении игровыми персонажами в играх, машинами-роботами на дорогах, дронами, продвинутыми чат-ботами и роботизированными ассистентами.

Структура книги

Учебник посвящён классическому машинному обучению. В [первой части](#) изучаются основные постановки задач машинного обучения и фундаментальные понятия, необходимые для настройки моделей. Во [второй части](#) рассказывается про подготовку данных перед их использованием прогнозирующими моделями. В [третьей части](#) представлен общий вид классификаторов, необходимый для понимания их работы. [Четвёртая часть](#) посвящена метрическим методам регрессии и классификации, которые строят прогнозы, отталкиваясь от расстояний между изучаемыми объектами. [Пятая часть](#) знакомит читателя с линейной регрессией вместе с её всевозможными обобщениями и усложнениями. Оценивание качества прогнозов в задаче регрессии представлено в [шестой части](#). В [седьмой части](#) даётся определение линейных классификаторов в общем виде, а также рассказывается про популярные методы этого класса - метод опорных векторов и логистическую регрессию. Построению многоклассовых классификаторов из набора бинарных посвящена [восьмая часть](#). В [девятой части](#) описываются основные методы градиентной оптимизации, применяемые в классическом машинном обучении. В [десятой части](#) описаны методы оценки точности работы классификаторов. [Одиннадцатая часть](#) посвящена решающим деревьям. В [двенадцатой части](#) читатель познакомится с понятием переобученных и недообученных моделей на примере разложения на смещение и разброс. В [тринадцатой части](#) описывается принцип построения прогнозов, используя не одну модель, а сразу несколько, и приводятся описания популярных методов построения композиций моделей, включая усреднение, голосование, бэггинг, стэкинг и другие. [Четырнадцатая часть](#) описывает алгоритм бустинга - самого популярного и успешного метода построения композиций моделей. [Пятнадцатая](#) и [шестнадцатая](#) части посвящены различным подходам к интерпретации простых и более сложных моделей машинного обучения.

Учебник не затрагивает тему использования многослойных нейросетей (глубокого обучения). Этой теме посвящён [второй учебник](#) сайта.

Примеры кода для запуска методов

Многие методы, описанные в учебнике, сопровождаются примерами их запуска на языке python с использованием библиотек sklearn, numpy и matplotlib. Для этих библиотек использовались версии 1.3.0, 1.26.0 и 3.8.4 соответственно. Для анализа данных и тестирования различных методов машинного обучения удобно использовать бесплатную среду разработки [jupyterlab](#). Для удобства установки рекомендуется использовать менеджер пакетов [anaconda](#). В приводимых примерах для генерации данных используются следующие функции:

```
import numpy as np
from sklearn.datasets import make_moons

def get_demo_classification_data():
    X, Y = make_moons(n_samples=3000, noise=0.3, random_state=0) # генерируем данные для классификации
    X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.4, stratify=Y, random_state=0) # раз
    return X_train, X_test, Y_train, Y_test

def get_demo_regression_data():
    np.random.seed(0)
    X = np.random.normal(size=[3000, 5])
    NOISE = 0.3 * np.random.normal(size=[3000])
    Y = X.mean(axis=1) + (X**2).mean(axis=1) + NOISE # генерируем данные для регрессии
    X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.4, random_state=0) # разбиваем выбор
    return X_train, X_test, Y_train, Y_test
```

Примеры запуска методов будут идти после описания самих методов, но вы также можете посмотреть код сразу всех примеров [по ссылке](#) с результатами его работы.

[Предыдущая страница](#)
[« Машинное обучение](#)

[Следующая страница](#)
[Основы машинного обучения »](#)



[🏠 Основы машинного обучения](#)

Основы машинного обучения

В этом разделе вы узнаете основные понятия и этапы машинного обучения.

[📄 Обучение с учителем](#)

[Определение и примеры задач обучения с учителем. Объекты, признаки, обучающая выборка, типы задач машинного обучения с учителем](#)

[📄 Настройка параметров модели](#)

[Настройка моделей машинного обучения. Функция потерь, функция выигрыша, оптимизация параметров методов машинного обучения.](#)

[📄 Выпуклость потерь](#)

[Выпуклые функции потерь - определение и мотивация использования. Свойства выпуклых функций.](#)

[📄 Регуляризация модели](#)

[Регуляризация моделей машинного обучения. Виды регуляризации моделей-L1 регуляризация, L2 регуляризация, ElasticNet и другие методы.](#)

[📄 Взвешенный учёт наблюдений](#)

[Учёт объектов обучающей выборки с разными весами - формула, мотивация примеры использования.](#)

[📄 Связь с принципом максимального правдоподобия](#)

[Связь вероятностных моделей и моделей машинного обучения. Оценка вероятностных моделей методом максимального правдоподобия и связь с минимизац...](#)

[📄 Обобщающая способность](#)

[Недообучение и переобучение моделей машинного обучения. Понятие гиперпараметров моделей, их отличие от параметров моделей.](#)

[□ Оценка качества прогнозов](#)

[Оценка качества прогнозов и подбор гиперпараметров моделей. Стратифицированные выборки в машинном обучении.](#)

[□ Этапы решения задачи машинного обучения](#)

[Последовательность действий при разработке и внедрении алгоритмов машинного обучения. Методология CRISP-DM.](#)

[□ Обучение без учителя](#)

[Обучения без учителя в машинном обучении. Определение и основные виды задач.](#)

[□ Частичное обучение](#)

[Частичное и трансдуктивное обучение в машинном обучении.](#)

[□ Вопросы](#)

[Вопросы по машинному обучению.](#)

[Предыдущая страница](#)

[« Введение](#)

[Следующая страница](#)

[Обучение с учителем »](#)



🏠 Подготовка данных

Подготовка данных

В этом разделе вы изучите основные методы подготовки данных к работе с ними методов машинного обучения.

[□ Фильтрация выбросов](#)

[Задача фильтрация выбросов \(детекции аномалий\) в машинном обучении.](#)

[□ Заполнение пропусков](#)

[Методы заполнения пропусков в данных для вещественных, бинарных и категориальных признаков перед обработкой.](#)

[□ Обработка временного признака](#)

[Обработка временного признака при подготовке данных для моделей машинного обучения. Учёт тренда и сезонности.](#)

[□ Обработка категориальных признаков](#)

[Методы кодирования категориальных признаков при подготовке данных для обучения моделей машинного обучения. Ordinal encoding, frequency encoding, on...](#)

[□ Нормализация признаков](#)

[Нормализация признаков для моделей машинного обучения. Понятие медианы, квантили и персентили.](#)

[□ Генерация признаков](#)

[Подходы к генерации новых признаков \(feature engineering\) на этапе подготовки данных для обучения моделей машинного обучения.](#)

[□ Сокращение числа признаков](#)

[Отбор признаков \(feature selection\) и снижение размерности признакового пространства \(dimensionality reduction\) при подготовке данных для обучения модел...](#)

[□ Преобразование целевой переменной](#)

[Преобразование целевой переменной для повышения качества прогнозов.](#)

[□ Вопросы](#)

[Вопросы для самопроверки по подготовке данных перед применением машинного обучения.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)
[Фильтрация выбросов »](#)



[🏠 Классификаторы в общем виде](#)

Классификаторы в общем виде

В этом разделе вы изучите общий вид классификатора и способ оценки качества классификации через отступ.

[□ Общий вид прогнозирующих функций](#)

[Задача регрессии и классификации в общем виде. Дискриминантные функции \(рейтинги классов\).](#)

[□ Отступ классификации](#)

[Оценка качества прогнозов в задаче классификации. Функция потерь, отступ в классификации \(margin\).](#)

[□ Предсказание вероятностей и преобразование SoftMax](#)

[SoftMax преобразование - определение, применения и свойства. Гиперпараметр температуры в SoftMax.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Общий вид прогнозирующих функций »](#)



🏠 Метрические методы прогнозирования

Метрические методы прогнозирования

В этом разделе вы изучите метрические методы построения прогнозов.

[□ Метрические методы](#)

[Метрические методы прогнозирования для задачи регрессии и классификации.](#)

[□ Метод ближайших центроидов](#)

[Метод ближайших центроидов \(nearest centroids\) для классификации объектов в машинном обучении.](#)

[□ Метод К ближайших соседей](#)

[Метод К ближайших соседей \(K nearest neighbors\) для задачи регрессии и классификации объектов в машинном обучении.](#)

[□ Анализ метода К ближайших соседей](#)

[Преимущества и недостатки метода К ближайших соседей. Понятие проклятия размерности \(curse of dimensionality\).](#)

[□ Обобщение метода К ближайших соседей с весами](#)

[Обобщение метода К ближайших соседей за счёт взвешенного учёта объектов.](#)

[□ Веса в метрических методах](#)

[Выбор весов, с которыми учитываются объекты обучающей выборки при прогнозировании с помощью метрических методов машинного обучения.](#)

[□ Локально-постоянная регрессия](#)

[Регрессия Надарая-Ватсона, локально-постоянная регрессия - описание метода и основные параметры.](#)

[□ **Функции расстояния**](#)

[Популярные функции расстояния и меры близости в машинном обучении. Косинусная мера близости, расстояние Махаланобиса, мера близости Жаккара, ре..](#)

[□ **Вопросы**](#)

[Предыдущая страница](#)

[« Предсказание вероятностей и преобразование SoftMax](#)

[Следующая страница](#)
[Метрические методы »](#)

© 2023-25 [Виктор Китов](#). [Новости проекта](#).



🏠 [Линейная регрессия и её обобщения](#)

Линейная регрессия и её обобщения

В этом разделе вы изучите алгоритм линейной регрессии и способы его усложнения и обобщения.

[□ Линейная регрессия](#)

[Модель линейной регрессии - определение, предположения, достоинства и недостатки.](#)

[□ Аналитическое решение для линейной регрессии](#)

[Метод наименьших квадратов \(МНК-оценка\)-аналитический вывод оптимальных значений для коэффициентов линейной регрессии.](#)

[□ Регуляризация в линейной регрессии](#)

[Регуляризация в линейной регрессии. Методы регуляризации и их свойства.](#)

[□ Аналитическое решение для гребневой регрессии](#)

[Аналитический вывод решения для коэффициентов гребневой регрессии.](#)

[□ Линейный ансамбль моделей](#)

[Построение ансамбля моделей через их линейную комбинацию. Способы регуляризации ансамбля.](#)

[□ Регрессия опорных векторов](#)

[Метод регрессии опорных векторов - определение, интуиция, обобщение через ядра Мерсера \(kernel trick\). Опорные и неинформативные объекты.](#)

[□ Orthogonal matching pursuit](#)

[Метод orthogonal matching pursuit с последовательным включением признаков в модель линейной регрессии.](#)

[□ Локально-линейная регрессия](#)

[Локально-линейная регрессия \(LOWESS\) - определение и свойства.](#)

[□ Дополнительная литература](#)

[□ Вопросы](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)
[Линейная регрессия »](#)

© 2023-25 [Виктор Китов. Новости проекта.](#)



[🏠 Оценка качества регрессии](#)

Оценка качества регрессии

В этом разделе вы узнаете, как можно оценивать качество прогнозов регрессионных моделей.

[☐ Меры оценки качества регрессионных прогнозов](#)

[Способы оценки качества регрессионных прогнозов и их сравнительный анализ.](#)

[☐ Конечные меры эффективности](#)

[Ключевые показатели эффективности \(key performance indicators, KPI\) и их связь со стандартными мерами качества моделей машинного обучения.](#)

[☐ Поточечный график](#)

[Визуализация и анализ ошибок прогнозирования в задаче регрессии.](#)

[☐ Вопросы](#)

[Вопросы для самопроверки по задаче регрессии.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Меры оценки качества регрессионных прогнозов »](#)



[🏠 Линейная классификация](#)

Линейная классификация

В этом разделе вы изучите алгоритм линейной классификации и способы его настройки.

[□ Линейная классификация](#)

[Линейный бинарный и многоклассовый классификатор в общем виде.](#)

[□ Оценка весов линейного классификатора](#)

[Методы линейной классификации - основные функции потерь, виды регуляризации моделей и влияние масштаба признаков на прогнозы.](#)

[□ Бинарная логистическая регрессия](#)

[Логистическая регрессия \(logistic regression\) для решения задачи бинарной классификации. Расчёт вероятностей классов.](#)

[□ Многоклассовая логистическая регрессия](#)

[Логистическая регрессия для решения задачи многоклассовой классификации.](#)

[□ Метод опорных векторов](#)

[Метод опорных векторов \(support vector machine, SVM\) для задачи классификации. Его геометрическое обоснование и обобщение через ядра Мерсера \(kernel...](#)

[□ Дополнительная литература](#)

[□ Вопросы](#)

[Вопросы по метрическим методам и линейной классификации.](#)

[Предыдущая страница](#)

[« Вопросы](#)



[🏠 Многоклассовая классификация набором бинарных классификаторов](#)

Многоклассовая классификация набором бинарных классификаторов

В этом разделе вы узнаете, как можно с помощью набора бинарных классификаторов решать задачу многоклассовой классификации.

[☐ Метод один-против-всех](#)

[Многоклассовая классификация с помощью набора бинарных классификаторов методом один-против-всех \(one-vs-all, one-vs-rest\).](#)

[☐ Метод один-против-одного](#)

[Многоклассовая классификация с помощью набора бинарных классификаторов методом один-против-одного \(one-vs-one\).](#)

[☐ Кодирование с исправлением ошибок](#)

[Метод кодирования с исправлением ошибок \(error-correcting output codes\) для решения задачи многоклассовой классификации с помощью набора бинарных...](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Метод один-против-всех »](#)



[🏠 Численная оптимизация](#)

Численная оптимизация

В этом разделе вы узнаете, как можно настраивать веса моделей в случае, когда минимум эмпирического риска нельзя найти аналитически.

[□ Численные методы оптимизации](#)

[Численные методы оптимизации для настройки параметров моделей машинного обучения. Понятие градиента и антиградиента. Безградиентные методы опти...](#)

[□ Метод градиентного спуска](#)

[Метод градиентного спуска \(gradient descent\) для настройки параметров моделей машинного обучения. Условия остановки, особенности решений, выбор ша...](#)

[□ Метод стохастического градиентного спуска](#)

[Метод стохастического градиентного спуска \(stochastic gradient descent, SGD\) для настройки параметров моделей машинного обучения. Минибатч объектов,...](#)

[□ Мониторинг сходимости](#)

[Мониторинг сходимости методов оптимизации. Экспоненциальное сглаживание и скользящее среднее.](#)

[□ Стохастический градиентный спуск с инерцией](#)

[Использование инерции \(momentum\) для ускорения метода стохастического градиентного спуска. Инерция Нестерова \(Nesterov momentum\).](#)

[□ Метод Ньютона](#)

[Метод Ньютона для настройки параметров моделей машинного обучения. Обоснование метода, его достоинства и недостатки. Методы оптимизации второго...](#)

[□ Вопросы](#)

[Вопросы по градиентным методам оптимизации в машинном обучении.](#)

[Предыдущая страница](#)

[« Кодирование с исправлением ошибок](#)

[Следующая страница](#)

[Численные методы оптимизации »](#)

© 2023-25 [Виктор Китов](#). [Новости проекта](#).



🏠 [Оценка качества классификации](#)

Оценка качества классификации

В этом разделе вы изучите основные меры оценки качества классификации.

[□ Базовые меры качества многоклассовой классификации](#)

[Основные меры для оценки качества классификации. Точность, частота ошибок, матрица ошибок. Взвешенный учёт ошибок разных типов.](#)

[□ Специальные меры качества для бинарной классификации](#)

[Специальные меры оценки качества прогнозов в задаче бинарной классификации, precision и recall, F-мера. Задача ранжирования. Меры precision@K, recall@...](#)

[□ Обобщение бинарных мер качества на многоклассовый случай](#)

[Расширение точности и полноты для оценки многоклассовых прогнозов. Микроусреднение и макроусреднение. Мера mean average precision \(mAP\).](#)

[□ ROC-кривая](#)

[Определение ROC-кривой \(ROC-curve\). Площадь под ROC-кривой AUC \(area under curve\). Меры true positive rate \(TPR\), false positive rate \(FPR\) в бинарной кла...](#)

[□ Лучший классификатор на ROC кривой](#)

[Нахождение наилучшего классификатора на ROC-кривой.](#)

[□ Эквивалентное определение AUC](#)

[Эквивалентное определение AUC \(area under curve\) как доли верно упорядоченных пар объектов. Доказательство эквивалентности. Способ оптимизации AUC...](#)

[□ Контроль качества предсказания вероятностей](#)

[Калибровка вероятностей и методы оценки качества вероятностных прогнозов. Мера Бриера \(Brier score\).](#)

[□ Дополнительная литература](#)

[Вопросы по методам оценки качества классификации.](#)

[□ Вопросы](#)

[Вопросы по методам оценки качества классификации.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Базовые меры качества многоклассовой классификации »](#)



🏠 Решающие деревья

Решающие деревья

В этом разделе вы изучите алгоритм решающих деревьев для решения задачи регрессии и классификации.

[□ Решающие деревья](#)

[Алгоритм решающих деревьев \(decision trees\) в задачах прогнозирования. CART, C4.5, ID3. Пример решающего дерева. Связь с правилами алгоритмами \(ru...](#)

[□ Особенности прогнозов решающего дерева](#)

[Особенности прогнозов решающими деревьями в задачах классификации и регрессии. Взаимное дополнение решающих деревьев и линейных моделей.](#)

[□ Настройка решающего дерева](#)

[Алгоритм настройки решающего дерева \(decision tree fitting\). Выбор решающих правил во внутренних узлах и назначение прогноза в листовых вершинах дере...](#)

[□ Функции неопределённости](#)

[Функции неопределённости \(impurity functions\) для настройки решающих деревьев. Их мотивация и свойства. Критерий Джини, энтропийный критерий и крит...](#)

[□ Учёт пользовательской функции потерь](#)

[Обоснование функций неопределённости \(impurity functions\) при настройке решающего дерева как минимизаторов потерь определённого вида. Обосновани...](#)

[□ Обрезка решающих деревьев](#)

[Методы обрезки решающих деревьев \(tree pruning\) для повышения качества прогнозов решающих деревьев. Метод обрезки по минимальной цене \(minimal co...](#)

[□ Обработка пропущенных значений](#)

[Методы заполнения пропущенных значений признаков в алгоритме решающего дерева.](#)

[□ Важность признаков](#)

[Оценка важности признаков \(feature importance\), используя решающее дерево - метод mean decrease in impurity.](#)

[□ Анализ решающих деревьев](#)

[Преимущества и недостатки решающих деревьев \(decision trees\) в машинном обучении.](#)

[□ Обобщения решающих деревьев](#)

[Обобщения и усложнения алгоритма решающих деревьев \(decision trees\).](#)

[□ Дополнительная литература](#)

[Вопросы по методам оценки качества классификации.](#)

[□ Вопросы](#)

[Вопросы по решающим деревьям.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Решающие деревья »](#)



[🏠 Переобучение и недообучение](#)

Переобучение и недообучение

В этом разделе мы детальнее разберём понятия переобученных и недообученных моделей.

[☐ Сложность прогнозирующих моделей](#)

[Переобучение \(overfitting\) и недообучение \(underfitting\) в моделях машинного обучения. Важность подбора сложности \(выразительной способности\) модели.](#)

[☐ Разложение на смещение и разброс](#)

[Разложение на смещение и разброс \(bias-variance decomposition\) - интерпретация, компоненты разложения для переобученных \(overfitted\) и недообученных \(...\)](#)

[☐ Доказательство разложения](#)

[Доказательство разложения на смещение и разброс \(bias-variance decomposition\).](#)

[☐ Дополнительная литература](#)

[Вопросы по методам оценки качества классификации.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Сложность прогнозирующих моделей »](#)



[🏠 Ансамбли моделей](#)

Ансамбли моделей

В этом разделе вы изучите как строить прогнозы с помощью не одной, а сразу нескольких моделей.

[□ Ансамбли моделей](#)

[Понятие ансамблей моделей \(model ensemble\) в машинном обучении. Мотивация использования ансамблей для борьбы с переобучением, недообучением и...](#)

[□ Математическое обоснование ансамблей](#)

[Преимущества использования ансамблей моделей. Доказательства, что можно из набора неточных моделей можно построить точный ансамбль для задач рег...](#)

[□ Простая агрегация в ансамблях](#)

[Простые способы объединения прогнозов в ансамблях моделей. Голосование по большинству, усреднение прогнозов, усреднение рейтингов Бриера.](#)

[□ Методы построения базовых моделей](#)

[Способы построения различных базовых моделей для формирования ансамбля моделей \(model ensemble\).](#)

[□ Настройка на разных фрагментах обучающей выборки](#)

[Методы построения ансамблей моделей машинного обучения по подвыборкам из исходной обучающей выборки. Методы bagging, random subspaces, rando...](#)

[□ Ансамбли рандомизированных деревьев](#)

[Алгоритм случайного леса \(random forest\) и особо случайных деревьев \(extra trees\). Описание методов и основные гиперпараметры. Примеры запуска в skleam...](#)

[□ Стэкинг](#)

[Алгоритм стэкинга \(stacking\) и блендинга \(blending\) для объединения прогнозов разных моделей в ансамбль. Пример использования стэкинга в skleam.](#)

[□ Дополнительная литература](#)

[Вопросы по методам оценки качества классификации.](#)

[□ Вопросы](#)

[Вопросы по построению ансамбля моделей.](#)

[Предыдущая страница](#)

[« Дополнительная литература](#)

[Следующая страница](#)

[Ансамбли моделей »](#)



🏠 Бустинг

Бустинг

В этом разделе вы изучите алгоритм бустинга.

[☐ Бустинг](#)

[Алгоритм бустинга - основанная идея и особенности реализации метода.](#)

[☐ Сравнение бустинга с другими ансамблями моделей](#)

[Сравнение алгоритма бустинга с бэггингом, случайным лесом, стэкингом и другими методами построения ансамблей моделей.](#)

[☐ Алгоритм AdaBoost](#)

[Алгоритм AdaBoost, его основные предположения, интуиция и математический вывод. Идея обобщения метода.](#)

[☐ Градиентный бустинг](#)

[Идея метода градиентного бустинга \(gradient boosting\), интуиция метода, алгоритм настройки и пример его работы в случае регрессии и бинарной классифи...](#)

[☐ Алгоритм градиентного бустинга](#)

[Пошаговый алгоритм настройки градиентного бустинга. Особенность реализации градиентного бустинга для решающих деревьев.](#)

[☐ Улучшения градиентного бустинга](#)

[Методы повышения точности градиентного бустинга за счёт сжатия \(shrinkage\) и обучения базовых моделях на подмножествах объектов и признаков \(subsam...](#)

[☐ Иллюстрация работы](#)

[Визуализация работы градиентного бустинга по шагам. Анализ изменения ошибок в зависимости от шага.](#)

[□ Градиентный бустинг второго порядка](#)

[Градиентный бустинг второго порядка - алгоритм настройки следующей базовой модели градиентного бустинга при квадратичном приближении функции по...](#)

[□ Популярные реализации](#)

[Продвинутые реализации градиентного бустинга - xgBoost, LightGBM, CatBoost. Основные идеи улучшений.](#)

[□ Точность градиентного бустинга](#)

[Анализ точности градиентного бустинга. Случаи, когда точнее работают метод К ближайших соседей и линейные модели. Важность использования ансамбля...](#)

[□ Дополнительная литература](#)

[Вопросы по методам оценки качества классификации.](#)

[□ Вопросы](#)

[Вопросы по градиентному бустингу.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Бустинг »](#)



🏠 [Интерпретация простых моделей](#)

Интерпретация простых моделей

В этом разделе вы изучите методы интерпретации простых моделей машинного обучения.

[□ Интерпретируемое машинное обучение](#)

[Понятие интерпретируемости моделей машинного обучения. Глобальная и локальная интерпретируемость. Интерпретируемые модели \(white-box models\) и н.](#)

[□ Интерпретация метрических методов](#)

[Интерпретация прогнозов метода К ближайших соседей и метода ближайших центроидов.](#)

[□ Метод наивного Байеса](#)

[Определение и интерпретация прогнозов метода наивного Байеса. Предположение наивного Байеса \(naive Bayes assumption\).](#)

[□ Интерпретация линейной регрессии](#)

[Интерпретация весов и прогнозов линейной регрессии.](#)

[□ Интерпретация логистической регрессии](#)

[Интерпретация весов логистической регрессии.](#)

[□ Интерпретация решающего дерева](#)

[Интерпретация модели и прогнозов решающего дерева.](#)

[□ Вопросы](#)

[Вопросы по интерпретируемости простых моделей машинного обучения.](#)

[Предыдущая страница](#)
[« Вопросы](#)

[Следующая страница](#)
[Интерпретируемое машинное обучение »](#)

© 2023-25 [Виктор Китов](#). [Новости проекта](#).



🏠 Интерпретация сложных моделей

Интерпретация сложных моделей

В этом разделе вы изучите подходы к интерпретации сложных моделей.

[□ Интерпретация сложных моделей](#)

[Способы интерпретации сложных моделей машинного обучения \(black-box model interpretation\). Ограничение подходов, вызванное тем, что корреляция не вс...](#)

[□ Анализ ошибок модели](#)

[Анализ ошибок модели с помощью графика зависимости прогнозных значений от реальных. Матрица ошибок.](#)

[□ Прогнозы на типичных и нетипичных объектах](#)

[Метод анализа моделей на типичных объектах \(прототипах\) и нетипичных объектах \(критиках\). Методы выделения прототипов и критиков.](#)

[□ Влияние признаков на качество прогнозов](#)

[Анализ моделей машинного обучения по влиянию каждого признака на точность моделей. Методом перестановочной важности признаков \(permutation featu...](#)

[□ Значения Шепли](#)

[Значения Шепли \(Shapley values\) для оценки важности признаков как для отдельного прогноза, так и в целом по выборке. Пример использования и формула д...](#)

[□ Локальное объяснение интерпретируемой моделью](#)

[Алгоритм LIME для объяснения и интерпретации прогнозов сложными моделями, такими как нейросети. Особенности применения и пример использования.](#)

[□ Влияние фрагментов](#)

[Интерпретация моделей методом оценки влияния фрагментов изображения на прогноз. Выделение фрагментов методом разбиения изображения на суперпи...](#)

[□ Зависимость прогноза от признаков](#)

[Интерпретация моделей за счёт анализа влияния отдельного признака на прогноз модели. Методы Partial Dependence Plot, Individual Conditional Expectation, M...](#)

[□ Контрфактические объяснения](#)

[Анализ и интерпретация прогнозов модели с помощью контрфактических объяснений \(counterfactual explanations\). Метод расчёта и примеры контрфактичес...](#)

[□ Влияние обучающих объектов](#)

[Анализ моделей за счёт оценки влияния на них отдельных обучающих объектов. Примеры применения этого подхода для лучшей настройки моделей машинн...](#)

[□ Вопросы](#)

[Вопросы по интерпретации моделей машинного обучения.](#)

[Предыдущая страница](#)

[« Вопросы](#)

[Следующая страница](#)

[Интерпретация сложных моделей »](#)



🏠 Заключение

Содержание этой страницы

Заключение

💡 УЧЕБНИК ДОРАБАТЫВАЕТСЯ...

Учебник дорабатывается, периодически могут появляться новые темы и улучшаться описания существующих. Обновления учебника вы можете отслеживать в [телеграм-канале проекта](#).

Вы ознакомились с основными задачами, понятиями и моделями машинного обучения. Изучили, какие базовые предположения лежат в основе каждой модели, какие гиперпараметры нужно специфицировать, и на что они влияют. Научились строить прогнозы с помощью не одной модели, а целого набора, а также интерпретировать модели и оценивать их качество.

Анализ данных и машинное обучение - практические области, поэтому настоятельно советуем дополнить полученные теоретические знания практическими навыками. Для этого рекомендуется изучить одну из самых популярных библиотек на питоне `scikit-learn` по отличной документации [\[1\]](#) с объяснением и примерами использования основных методов. Существует и неофициальный перевод этой документации на русский язык [\[2\]](#), правда, для более старой версии.

Рекомендуем ознакомиться с сайтом kaggle.com, на котором регулярно выкладываются прикладные задачи машинного обучения из разных областей с необходимыми данными, а также поучаствовать в kaggle-соревнованиях [\[3\]](#). Это даст вам практические навыки работы с данными и опыт решения реальных задач! При этом стоит активно пользоваться форумом, на котором участники обсуждают задачу (раздел Discussion) и делятся своими решениями (раздел Code). Стоит отметить, что сайт содержит и собственные обучающие материалы [\[4\]](#) по практической работе с данными.

На этом ваше путешествие в мир машинного обучения на нашем сайте не заканчивается! В задачах машинного обучения всё больше начинают доминировать нейросети, позволяющие эффективно обрабатывать сложные и многомерные типы данных, такие как изображения, тексты и графы. Поэтому приглашаем прочитать [второй учебник](#), посвящённый нейросетям.

Желаем успехов!

Литература

1. [Документация sklearn](#).
2. [Документация sklearn на русском](#).
3. [kaggle.com competitions](https://kaggle.com/competitions).
4. [kaggle.com learn](https://kaggle.com/learn).

[Предыдущая страница](#)

[« Вопросы](#)



Посвящается моим родителям Владимиру и Ольге.

Введение в машинное обучение

Неформальное определение

В повседневной жизни все мы ежедневно сталкиваемся с принятием решений:

- при текущей дорожной ситуации надо ли ехать на метро или на машине?
- удалить ли письмо в электронной почте как спам или сохранить?
- ожидается ли дождь, и стоит ли брать с собой зонт?
- стоит ли звонить человеку с определённым предложением или он от него, скорее всего, откажется?
- стоит ли докупить хлеба и молока или их хватит до конца недели?
- какое фокусное расстояние на фотоаппарате установить, чтобы лицо фотографируемого человека получилось чётким?

Аналогичные проблемы принятия решений решаются организациями в массовом порядке:

- сколько хлеба и молока закупить магазину, чтобы удовлетворить спрос до конца недели?
- как почтовому сервису автоматически разделять письма на полезные и спам?
- какими способами и по каким маршрутам отправлять грузы?
- какую погоду предсказать сервису прогноза погоды на оставшийся конец дня?
- как автоматически устанавливать фокусное расстояние на производимых фотоаппаратах?

При массовом и повторяющемся принятии решений целесообразно процесс принятия этих решений автоматизировать. Можно разработать явную систему правил этого процесса. Например, при определении важности письма мы можем смотреть на то, переписывались ли мы ранее с отправителем, принадлежит ли отправитель надёжной и известной компании, включает ли текст письма определённые ключевые слова, которые нам заранее не интересны? В этом случае мы как бы явно программируем алгоритм принятия решений. Но проблема заключается в том, что

- Сложно разработать универсальный алгоритм, который бы подходил всем пользователям. Одних может не интересовать получение кредита или психологическая консультация, а для других это может оказаться актуальным.
- Сложно учесть всё многообразие ситуаций. Например, "бесплатная психологическая консультация" может быть сформулирована как "консультация психолога без оплаты", и изначальное правило уже перестанет действовать.

В подобных случаях полезно использовать **машинное обучение** (machine learning).

Машинное обучение - это процесс, в результате которого компьютер по наблюдаемым данным обучается лучше решать заданную задачу. Метод решения задачи при этом ищется в широком классе функций, параметризованном вектором параметров, который и подбирается по наблюдаемым данным.

Вместо явного прописывания четкой системы правил принятия решений в идеологии машинного обучения эти правила подбираются автоматически по данным. Под компьютером при этом может пониматься любое вычислительное устройство, например смартфон или процессор робота. Рассмотрим более детальное определение:

Машина учится на заданном **опыте** решать некоторую **задачу**, относительно некоторого **показателя качества**, если

показатель качества растёт на задаче после получения опыта.

В нашем примере задача - это классификация писем на спам/не спам, показатель качества - доля верно классифицированных писем, а опыт - коллекция прошлых писем, которые до этого были вручную размечены по классам.

В другом примере задачей выступает предсказание времени в пути, отталкиваясь от текущего времени суток, дня недели, погоды и загрузки дорог, показатель качества - модуль отклонения предсказанного времени от фактического, а опытом - история предыдущих передвижений в известных условиях и с известным временем в пути.

Примеры задач

Приведём примеры популярных задач, решаемых с помощью машинного обучения:

- Предсказать, уйдёт ли клиент к конкурентам? (churn prediction)
- Является ли последовательность финансовых транзакций мошеннической? (fraud detection)
- Предсказание пробок и времени в пути при планировании маршрута (traffic prediction).
- Стоит ли показывать заданный товар покупателю в качестве рекомендации? (recommender systems)
- Рекомендовать ли человека в качестве друга в социальной сети?
- Является ли аккаунт в социальной сети ботом?
- Голосовой ассистент: распознавание речи, автоматический ответ на вопросы, генерация речевого ответа.
- Идентификация человека по лицу. Распознавание номера машины на камерах.
- Подсчёт и отслеживание людей по камерам видеонаблюдения (object tracking). Обнаружение неправомерных действий (activity recognition).
- Автоматическое управление машинами (self-driving cars): распознавание ситуации, планирование маршрута.
- Автоматическая торговля на бирже (algorithmic trading).
- Перевод с одного языка на другой (machine translation).
- Постановка медицинских диагнозов по жалобам пациента и результатам обследований.
- Рекомендация веб-страниц по поисковому запросу (information retrieval).
- Автоматическая оценка ожидаемой зарплаты кандидата по резюме.
- Игра компьютера в шахматы, управление игровыми персонажами.
- Автоматическая оценка квартиры по её характеристикам.
- Хвалит или ругает пользователь товар в своём отзыве? (sentiment analysis)
- Генерация иллюстраций к тексту. Текстовое описание, что показано на изображении.
- Прогноз погоды. Рекомендации фермерам, когда сажать/поливать/удобрять посевы.
- Автоматическое написание программного кода (no code AI).
- Автоматический выбор, каким пользователям какую онлайн-рекламу показать (targeted ads).
- Генерация химических соединений, обладающих требуемыми свойствами:
 - крепкий, но легкий и термостойкий материал с повышенной проводимостью (material design)
 - препарат, обеспечивающий лечение и обладающий минимальными побочными эффектами (drug discovery)

Типы обучения

Машинное обучение (machine learning) описывает в целом подходы про подготовку данных, настройку и оценку прогнозирующих алгоритмов. Этому посвящена первая книга сайта, которую вы сейчас читаете.

Глубокое обучение (deep learning) - подраздел машинного обучения про сложные многоуровневые модели (нейросети), способные

решать более сложные задачи прогнозирования. С ростом вычислительных мощностей и объёма данных существует устойчивый тренд на замену классических алгоритмов машинного обучения на нейросетевые, обеспечивающие большую точность и возможность генерировать не только численные ответы, но и ответы в виде сложно структурированных данных, таких как текст, речь, изображение и видео. Глубокому обучению посвящена [вторая часть книги](#).

Обучение с подкреплением (reinforcement learning) - также подраздел машинного обучения, в котором строится не однократный прогноз независимо для каждого объекта, а вырабатывается интерактивная стратегия поведения в изменяемой среде.

Примером обучения с подкреплением может служить автоматическая игра в шахматы, в которой необходимо последовательно генерировать каждый следующий ход. Успех генерации определяется не только текущим ходом, но и всей последовательностью решений в течение партии. Обучение с подкреплением также применяется в управлении игровыми персонажами в играх, машинами-роботами на дорогах, дронами, продвинутыми чат-ботами и роботизированными ассистентами.

Структура книги

Учебник посвящён классическому машинному обучению. В [первой части](#) изучаются основные постановки задач машинного обучения и фундаментальные понятия, необходимые для настройки моделей. Во [второй части](#) рассказывается про подготовку данных перед их использованием прогнозирующими моделями. В [третьей части](#) представлен общий вид классификаторов, необходимый для понимания их работы. [Четвёртая часть](#) посвящена метрическим методам регрессии и классификации, которые строят прогнозы, отталкиваясь от расстояний между изучаемыми объектами. [Пятая часть](#) знакомит читателя с линейной регрессией вместе с её всевозможными обобщениями и усложнениями. Оценивание качества прогнозов в задаче регрессии представлено в [шестой части](#). В [седьмой части](#) даётся определение линейных классификаторов в общем виде, а также рассказывается про популярные методы этого класса - метод опорных векторов и логистическую регрессию. Построению многоклассовых классификаторов из набора бинарных посвящена [восьмая часть](#). В [девятой части](#) описываются основные методы градиентной оптимизации, применяемые в классическом машинном обучении. В [десятой части](#) описаны методы оценки точности работы классификаторов. [Одиннадцатая часть](#) посвящена решающим деревьям. В [двенадцатой части](#) читатель познакомится с понятием переобученных и недообученных моделей на примере разложения на смещение и разброс. В [тринадцатой части](#) описывается принцип построения прогнозов, используя не одну модель, а сразу несколько, и приводятся описания популярных методов построения композиций моделей, включая усреднение, голосование, бэггинг, стэкинг и другие. [Четырнадцатая часть](#) описывает алгоритм бустинга - самого популярного и успешного метода построения композиций моделей. [Пятнадцатая](#) и [шестнадцатая](#) части посвящены различным подходам к интерпретации простых и более сложных моделей машинного обучения.

Учебник не затрагивает тему использования многослойных нейросетей (глубокого обучения). Этой теме посвящён [второй учебник](#) сайта.

Примеры кода для запуска методов

Многие методы, описанные в учебнике, сопровождаются примерами их запуска на языке python с использованием библиотек sklearn, numpy и matplotlib. Для этих библиотек использовались версии 1.3.0, 1.26.0 и 3.8.4 соответственно. Для анализа данных и тестирования различных методов машинного обучения удобно использовать бесплатную среду разработки [jupyterlab](#). Для удобства установки рекомендуется использовать менеджер пакетов [anaconda](#). В приводимых примерах для генерации данных используются следующие функции:

```
import numpy as np
from sklearn.datasets import make_moons

def get_demo_classification_data():
    X, Y = make_moons(n_samples=3000, noise=0.3, random_state=0) # генерируем данные для классификации
    X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.4, stratify=Y, random_state=0) # раз
    return X_train, X_test, Y_train, Y_test

def get_demo_regression_data():
    np.random.seed(0)
    X = np.random.normal(size=[3000, 5])
    NOISE = 0.3 * np.random.normal(size=[3000])
    Y = X.mean(axis=1) + (X**2).mean(axis=1) + NOISE # генерируем данные для регрессии
    X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.4, random_state=0) # разбиваем выбор
    return X_train, X_test, Y_train, Y_test
```

Примеры запуска методов будут идти после описания самих методов, но вы также можете посмотреть код сразу всех примеров [по ссылке](#) с результатами его работы.

[Предыдущая страница](#)
[« Машинное обучение](#)

[Следующая страница](#)
[Основы машинного обучения »](#)

