

Лекция 1. Введение в дисциплину

Искусственный интеллект (ИИ) стал неотъемлемой частью различных коммерческих и исследовательских проектов, начиная от постановки медицинского диагноза с последующим лечением и заканчивая поиском друзей в социальных сетях. Многие полагают, что технологии ИИ могут использовать только крупные компании, обладающие мощными командами аналитиков, однако, это не так: небольшие системы ИИ можно построить самостоятельно.

ИИ заключается в извлечении знаний из данных. Это научная область, находящаяся на пересечении статистики, искусственного интеллекта и компьютерных наук и также известная как прогнозная аналитика или статистическое обучение. В последние годы применение методов ИИ в повседневной жизни стало обыденным явлением. Многие современные веб-сайты и устройства используют алгоритмы ИИ, начиная с автоматических рекомендаций по просмотру фильмов, заказа еды или покупки продуктов, и заканчивая персонализированными онлайн радиотрансляциями и распознаванием друзей на фотографиях. Когда вы видите сложный сайт типа Facebook или Google, то скорее всего, каждый его раздел содержит несколько моделей ИИ.

На заре появления «интеллектуальных» приложений многие системы использовали жесткие правила «if» и «else» для обработки данных или корректировки информации, введенной пользователем. Вспомните о спам-фильтре, чья работа состоит в том, чтобы переместить соответствующие входящие сообщения электронной почты в папку «Спам». Вы можете составить черный список слов, которые будут идентифицировать письмо как спам. Это пример использования системы экспертных правил для разработки «интеллектуального» приложения. Разработка правил принятия решений в ручном режиме допустимо в некоторых задачах, особенно в тех, где люди четко понимают процесс моделирования. Однако, использование жестких решающих правил имеет два основных недостатка:

- Логика, необходимая для принятия решения, относится исключительно к одной конкретной области и задачи. Даже несущественное изменение задачи может повлечь за собой переписывание всей системы.

- Разработка правил требует глубокого понимания процесса принятия решения.

Один из примеров, где этот жесткий подход потерпит неудачу – это распознавание лиц на изображениях. На сегодняшний день каждый смартфон может распознать лицо на изображении. Тем не менее, распознавание лиц была нерешенной проблемой, по крайней мере, до начала XXI века. Основная проблема заключается в том, что способ, с помощью которого компьютер «воспринимает» пиксели, формирующие изображение на компьютере, очень сильно отличается от человеческого восприятия лица. Эта разница в принципе не позволяет человеку сформулировать подходящий набор правил, описывающих лицо с точки зрения цифрового изображения. Однако, благодаря машинному обучению, простого предъявления большого количества изображений с лицами будет достаточно для того, чтобы алгоритм определил, какие признаки необходимы для идентификации лица.

Рассмотрим краткую историческую справку основных этапов развития методов машинного обучения. Историю современного искусственного интеллекта принято отсчитывать от 1959 года, в котором Артур Самуэль (Arthur Samuel), исследователь в области искусственного интеллекта и изобретатель первой самообучающейся компьютерной программы игры в шашки, ввел в научный обиход термин «машинное обучение». Самуэль определил машинное обучение как процесс, в результате которого компьютеры способны показать поведение, которое в них не было явно запрограммировано. Ранее, в 1950 Алан Тьюринг (Alan Turing) создает Тьюринг тест для оценки интеллекта компьютера. Тогда же, в 1959 Марвин Минский (Marvin Minsky) стал одним из сооснователей лаборатории Массачусетского технологического института. Профессор Минский создал первую обучающуюся машину SNARC со случайно связанной нейросетью.

В 1963 Ларри Робертс (Larry Roberts) сформулировал тезисы компьютерного зрения в своей диссертации в MIT. Далее, методы машинного обучения активно развивались и многие ведущие мировые ученые посвящали свои работы проблемам обработки данных с использованием интеллектуальных алгоритмов:

- 1967 Написан метрический алгоритм классификации (Метод k ближайших соседей). Алгоритм позволил компьютерам использовать простые шаблоны распознавания.

- 1981 Gerald Dejong представляет концепцию, основанную на обучении (Explanation Based Learning).

- 1985 Терри Сейновски (Terry Sejnowski) создает NetTalk искусственную нейронную сеть.

- 1986 Дэвидом Румельхартом (David Rumelhart) и Робби Вильямсом был заново открыт и популяризирован алгоритм обратного распространения ошибки. Этот алгоритм также был получен другими учеными независимо друг от друга. Впервые он был предложен Полом Вербосом (Paul Werbos) в 1974 году.

- 1997 Компьютер Deep Blue обыграл чемпиона мира по шахматам Гарри Каспарова.

- 2006 Джеффри Хинтон (Geoffrey Hinton), ученый в области искусственных нейросетей, ввел в обиход термин «Глубинное обучение» (Deep learning).

- 2012 Google запускает облачный сервис Google Prediction API для машинного обучения, помогающий анализировать неструктурированные данные.

- 2015 Microsoft создает платформу Distributed Learning Machine Toolkit, который предназначена для децентрализованного машинного обучения.

Наряду с разработками в области машинного обучения в процессе развития науки интеллектуального анализа данных, модернизацию претерпевала и философия искусственного интеллекта. Так, была выделена

гипотеза понятии сильного и слабого искусственного интеллекта, согласно которой некоторые формы искусственного интеллекта могут действительно обосновывать и решать проблемы. Теория сильного искусственного интеллекта предполагает, что компьютеры могут приобрести способность мыслить и осознавать себя как отдельную личность (в частности, понимать собственные мысли), хотя и не обязательно, что их мыслительный процесс будет подобен человеческому, в то время как теория слабого ИИ отвергает такую возможность.

Термин «сильный ИИ» был введен в 1980 году Джоном Сёрлом (в работе, описывающей мысленный эксперимент «Китайская комната»), впервые охарактеризовавшим его следующим образом: «Соответствующим образом, запрограммированный компьютер с нужными входами и выходами и будет разумом, в том смысле, в котором человеческий разум — это разум».

Основными *требованиями* к созданию *сильного искусственного интеллекта* являются:

- принятие решений, использование стратегий, решение головоломок и действия в условиях неопределенности;
- представление знаний, включая общее представление о реальности;
- планирование;
- обучение;
- общение на естественном языке;
- сила воли.

Объединение всех этих способностей воедино для достижения общих целей. Ведутся работы для создания машин, имеющих все эти способности, и предполагается, что Сильный ИИ будет иметь либо их все, либо большую часть из них.

Все задачи, решаемые с помощью ML, относятся к одной из следующих категорий:

- задачи регрессии – прогноз на основе выборки объектов с различными признаками. На выходе должно получиться вещественное число (2, 35, 76.454

и др.), к примеру цена квартиры, стоимость ценной бумаги по прошествии полугода, ожидаемый доход магазина на следующий месяц, качество вина при слепом тестировании.

– задачи классификации – получение категориального ответа на основе набора признаков. Имеет конечное количество ответов (как правило, в формате «да» или «нет»): есть ли на фотографии кот, является ли изображение человеческим лицом, болен ли пациент раком.

– задачи кластеризации – распределение данных на группы: разделение всех клиентов мобильного оператора по уровню платёжеспособности, отнесение космических объектов к той или иной категории (планета, звезда, чёрная дыра и т. п.).

– задачи снижения размерности – сведение большого числа признаков к меньшему (обычно 2–3) для удобства их последующей визуализации (например, сжатие данных).

– задачи выявления аномалий – отделение аномалий от стандартных случаев. На первый взгляд она совпадает с задачей классификации, но есть одно существенное отличие: аномалии – явление редкое, и обучающих примеров, на которых можно натаскать машинно обучающуюся модель на выявление таких объектов, либо исчезающе мало, либо просто нет, поэтому методы классификации здесь не работают. На практике такой задачей является, например, выявление мошеннических действий с банковскими картами.

Основные виды машинного обучения

Основная масса задач, решаемых при помощи методов машинного обучения, относится к двум разным видам: обучение с учителем (supervised learning) либо без него (unsupervised learning). Однако этим учителем вовсе не обязательно является сам программист, который стоит над компьютером и контролирует каждое действие в программе. «Учитель» в терминах машинного обучения – это само вмешательство человека в процесс обработки информации. В обоих видах обучения машине предоставляются исходные

данные, которые ей предстоит проанализировать и найти закономерности. Различие лишь в том, что при обучении с учителем есть ряд гипотез, которые необходимо опровергнуть или подтвердить. Эту разницу легко понять на примерах.

Машинное обучение с учителем

Предположим, в нашем распоряжении оказались сведения о десяти тысячах московских квартир: площадь, этаж, район, наличие или отсутствие парковки у дома, расстояние от метро, цена квартиры и т. п. Нам необходимо создать модель, предсказывающую рыночную стоимость квартиры по её параметрам. Это идеальный пример машинного обучения с учителем: у нас есть исходные данные (количество квартир и их свойства, которые называются признаками) и готовый ответ по каждой из квартир – её стоимость. Программе предстоит решить задачу регрессии.

Практические примеры постановки задач классификации, решаемые с использованием обучения с учителем: необходимо подтвердить или опровергнуть наличие заболевания у пациента, зная все его медицинские показатели; выяснить, является ли входящее письмо спамом, проанализировав его текст.

Машинное обучение без учителя

В случае обучения без учителя, когда готовых «правильных ответов» системе не предоставлено, всё обстоит ещё интереснее. Например, у нас есть информация о весе и росте какого-то количества людей, и эти данные нужно распределить по трём группам, для каждой из которых предстоит пошить рубашки подходящих размеров. Это задача кластеризации. В этом случае предстоит разделить все данные на 3 кластера (но, как правило, такого строгого и единственно возможного деления нет).

Если взять другую ситуацию, когда каждый из объектов в выборке обладает сотней различных признаков, то основной трудностью будет графическое отображение такой выборки. Поэтому количество признаков

уменьшают до двух или трёх, и становится возможным визуализировать их на плоскости или в 3D. Возникает *задача уменьшения размерности*.

Алгоритмы машинного обучения можно описать как обучение целевой функции f , которая наилучшим образом соотносит входные переменные X и выходную переменную Y : $Y = f(X)$. Цель машинного обучения определить вид функции f – закон отображения множества X в Y . Алгоритмы машинного обучения позволяют установить зависимость между данными. Наиболее распространённой задачей в машинном обучении является предсказание значений Y для новых значений X . Это называется прогностическим моделированием, цель исследования – сделать как можно более точное предсказание. Рассмотрим некоторые методы, направленные на формализацию механизмов обработки и анализа данных с использованием инструментов машинного обучения, в дальнейшем разберем основные методы более подробно, применительно к решаемым задачам.

Линейная регрессия

Линейная регрессия – один из наиболее известных и понятных алгоритмов в обработке статистических данных и машинном обучении. Прогностическое моделирование в первую очередь касается минимизации ошибки модели или, другими словами, как можно более точного прогнозирования. Как правило, принцип минимизации ошибки строится исходя из метода наименьших квадратов (МНК) – минимизации среднего квадратического отклонения полиномиальных значений регрессионной модели от соответствующих фактических значений точно заданной функции Y . Линейную регрессию можно представить в виде уравнения, которое описывает прямую, наиболее точно показывающую взаимосвязь между входными переменными X и выходными переменными Y . Для составления этого уравнения нужно найти определённые коэффициенты для уравнений полинома $P_1(x) = a_0 + a_1x + \dots + a_nx$. Оценка ошибки модели производится на основе анализа среднего квадратического отклонения, определяемого по следующей формуле:

$$\sigma_m = \sqrt{\frac{1}{n-m} \sum_{k=0}^n \varepsilon_k^2},$$

где $\varepsilon_k = P_m(x_k) - Y_k$ – отклонение полинома регрессионной модели от значения Y_k в точке x_k .

Графически, линейная регрессионная модель анализа данных представлена на рисунке 1, включающем расстояния, определяющие ошибки (отклонения) модели от реальных значений.

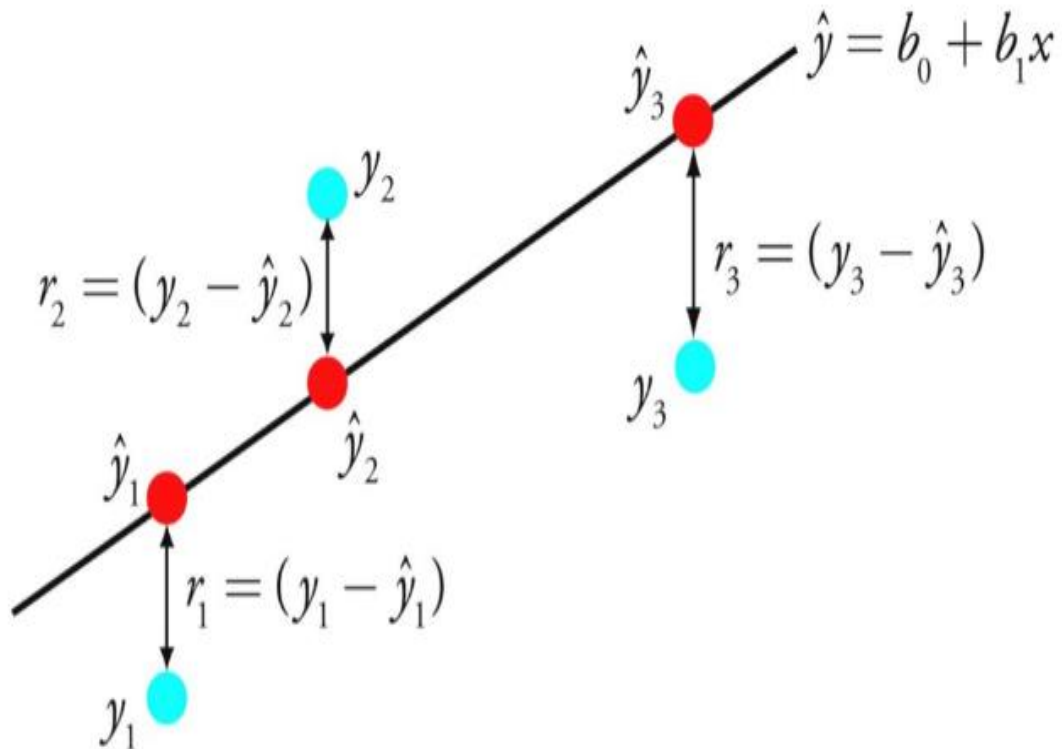


Рисунок 1 – Пример линейной регрессионной модели

Отметим, что принципы построения регрессионных моделей предполагают использование различных мер для оценки ошибки, однако, метод наименьших квадратов является одним из наиболее простых и эффективных подходов, а потому – популярным.

Логистическая регрессия

Для задач бинарной классификации целесообразно использовать более сложные модели, корнями, однако, также уходящие в статистику, как, например, логистическая регрессия. Благодаря тому, как обучается модель, предсказания логистической регрессии можно использовать для отображения

вероятности принадлежности образца к классу 0 или 1. Это полезно в тех случаях, когда нужно иметь больше обоснований для прогнозирования (рисунок 2).

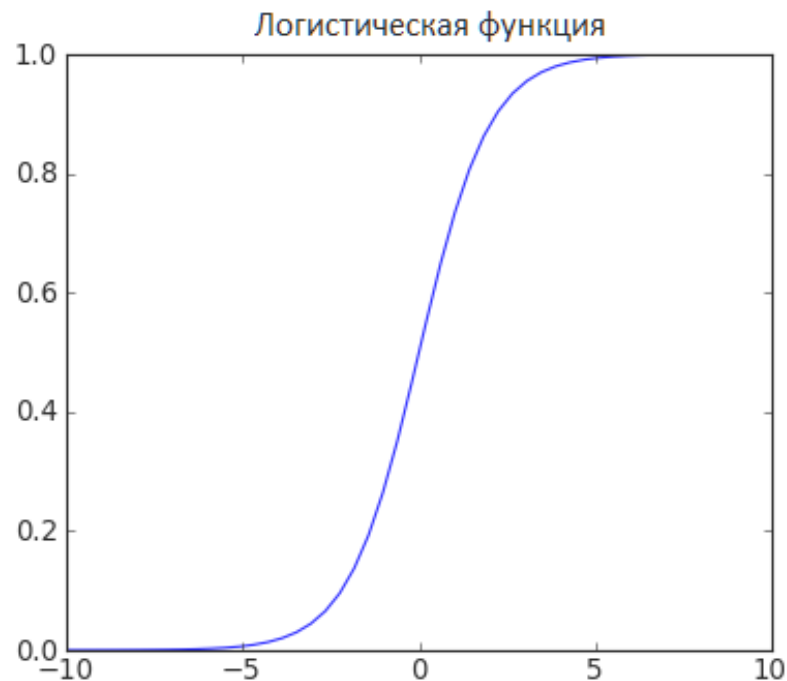


Рисунок 2 – Пример логистической регрессионной модели

Как и в случае с линейной регрессией, логистическая регрессия выполняет свою задачу лучше, если убрать лишние и похожие переменные. Модель логистической регрессии быстро обучается и хорошо подходит для задач бинарной классификации.

Линейный дискриминантный анализ (LDA)

Логистическая регрессия используется, когда нужно отнести образец к одному из двух классов. Если классов больше, чем два, то лучше использовать алгоритм LDA (Linear discriminant analysis).

Представление LDA довольно простое. Оно состоит из статистических свойств данных, рассчитанных для каждого класса. Для каждой входной переменной это включает:

- среднее значение для каждого класса;
- дисперсию, рассчитанную по всем классам.

ЛДА тесно связан также с методом главных компонент (МГК, англ. Principal Component Analysis, PCA) и факторным анализом тем, что они ищут линейные комбинации переменных, которые лучшим образом способны объяснить данные. ЛДА явным образом пытается моделировать разницу между классами данных, в то время как МГК, с другой стороны, не принимает во внимание какую-либо разницу в классах, а факторный анализ строит комбинации признаков, опираясь скорее на различия, а не на сходства. Дискриминантный анализ отличается также от факторного анализа тем, что не является независимой техникой — для его работы должно быть определено различие между независимыми зависимыми переменными (последние называются также критериальными переменными). Отметим, что ЛДА применим, когда измерения, сделанные на независимых переменных для каждого наблюдения, являются непрерывными величинами. Когда имеем дело с качественными независимыми переменными, эквивалентной техникой является дискриминантный анализ соответствий. Дискриминантный анализ используется, когда группы известны априори (в отличие от кластерного анализа). Каждый случай должен иметь значение в одной или нескольких мер количественного предсказания и значение на групповой мере. Иными словами, анализ дискриминантных функций является классификацией, разбивающей объекты на группы, классы или категории некоторого типа (рисунок 3).

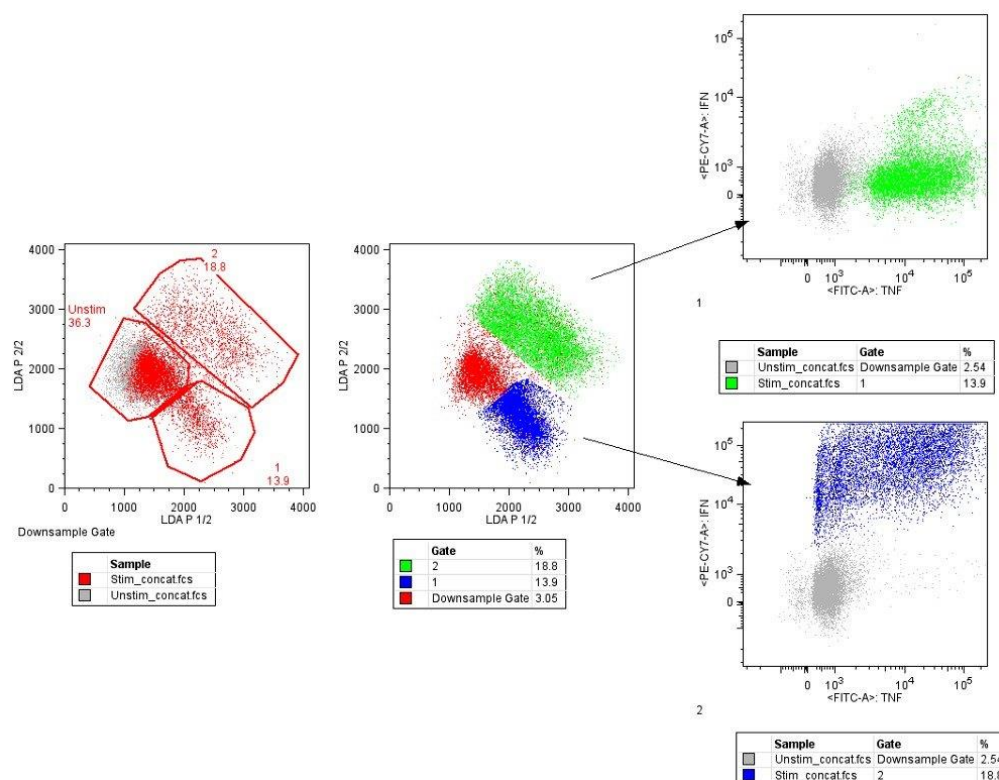


Рисунок 3 – Линейный дискриминантный анализ двух множеств

Предсказания производятся путём вычисления дискриминантного значения для каждого класса и выбора класса с наибольшим значением. Предполагается, что данные имеют нормальное распределение, поэтому перед началом работы рекомендуется удалить из данных аномальные значения. Это простой и эффективный алгоритм для задач классификации.

Деревья принятия решений

Деревья принятия решений широко используются в прикладных задачах анализа данных и поддержки принятия решений. С использованием деревьев принятия решений допустимо моделировать экономические, производственные задачи, проблемы классификации. Решающие деревья широко используются в принятии управленческих решений во всех сферах профессиональной деятельности человека: от финансов до медицины и автоматизации производства.

Дерево решений можно представить в виде двоичного дерева, знакомого многим по алгоритмам и структурам данных. Каждый узел представляет собой

входную переменную и точку разделения для этой переменной (при условии, что переменная — число). Листовые узлы — это выходная переменная, которая используется для предсказания. Предсказания производятся путём прохода по дереву к листовому узлу и вывода значения класса на этом узле. Деревья быстро обучаются и делают предсказания. Кроме того, они точны для широкого круга задач и не требуют особой подготовки данных. Конечно, решающие деревья не являются деревом в прямом смысле этого слова — это направленный граф, включающий «родительские» узлы, листья, характеризующие состояния системы и «ветви» — условия перехода, представленные ребрами графа (рисунок 4).

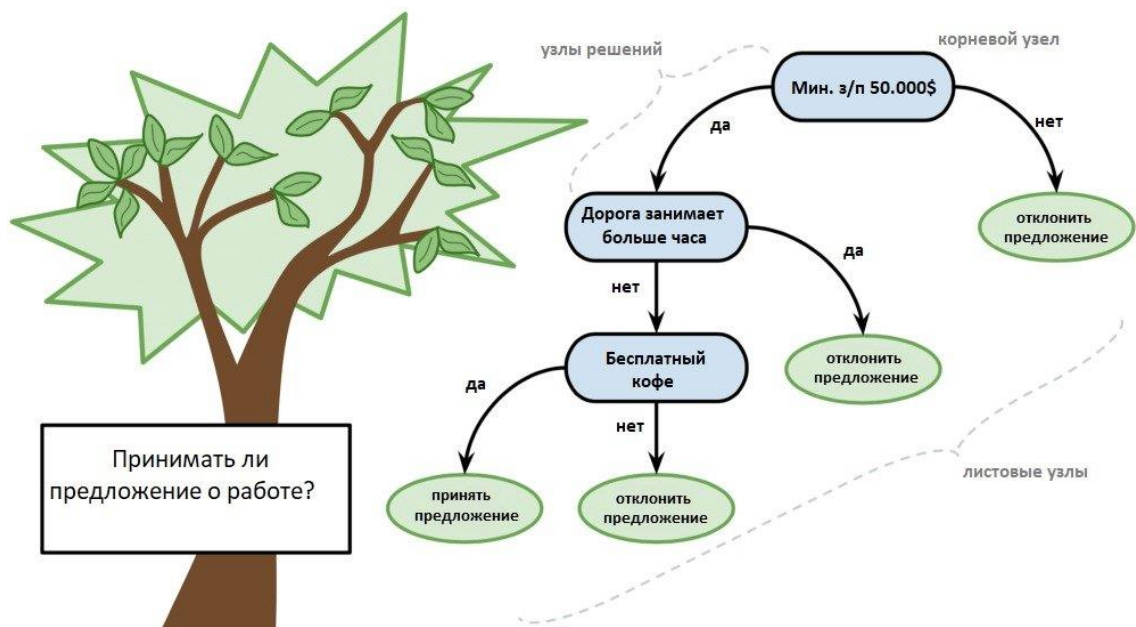


Рисунок 4 – Дерево принятия решений

Деревья принятия решений широко используются в машинном обучении благодаря применению принципов бинарной классификации — разделения на классы исходя из состава начальных данных.

Метод К-ближайших соседей (KNN)

Метод К-ближайших соседей — очень простой и очень эффективный алгоритм. Модель KNN (K-nearest neighbors) представлена всем набором тренировочных данных. Предсказание для новой точки делается путём поиска

K ближайших соседей в наборе данных и суммирования выходной переменной для этих K экземпляров.

Вопрос лишь в том, как определить сходство между экземплярами данных. Если все признаки имеют один и тот же масштаб (например, сантиметры), то самый простой способ заключается в использовании евклидова расстояния — числа, которое можно рассчитать на основе различий с каждой входной переменной (рисунок 5).

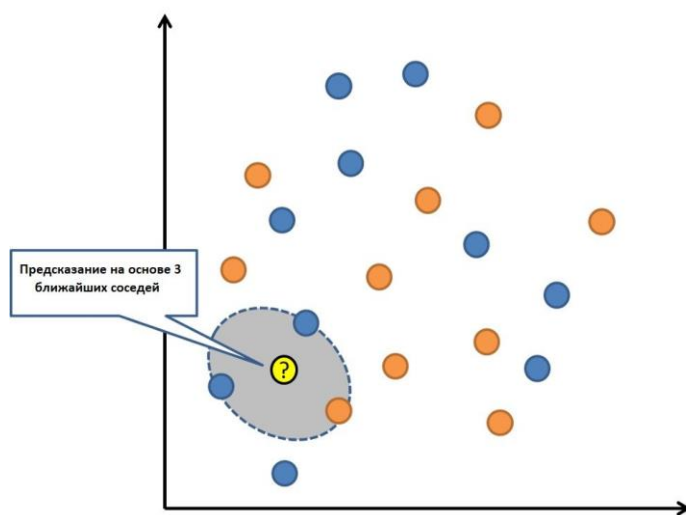


Рисунок 5 – Метод ближайших соседей

Метод KNN может потребовать много памяти для хранения всех данных, но зато быстро сделает предсказание. Также обучающие данные можно обновлять, чтобы предсказания оставались точными с течением времени. Идея ближайших соседей может плохо работать с многомерными данными (множество входных переменных), что негативно скажется на эффективности алгоритма при решении задачи, что вызвано большой размерностью начальных данных. Иными словами, стоит использовать лишь наиболее важные для предсказания переменные.

Представленный перечень методов машинного обучения не является исчерпывающим, однако позволяет представить основные направления развития интеллектуальных алгоритмов решения задач анализа данных.

Контрольные вопросы по теме:

1. Искусственный интеллект как направление знаний. Основные направления.
2. Перечислите характерные особенности «сильного» искусственного интеллекта.
3. Сформулируйте основные задачи, решаемые с помощью машинного обучения.
4. Методы машинного обучения, как подмножество методов искусственного интеллекта.
5. Как вы понимаете термины машинного обучения с учителем и без учителя?
6. Сформулируйте основные понятия задач регрессии.
7. В чем суть линейного дискриминантного анализа?
8. Где используются деревья принятия решений?
9. В чем суть метода К-ближайших соседей?