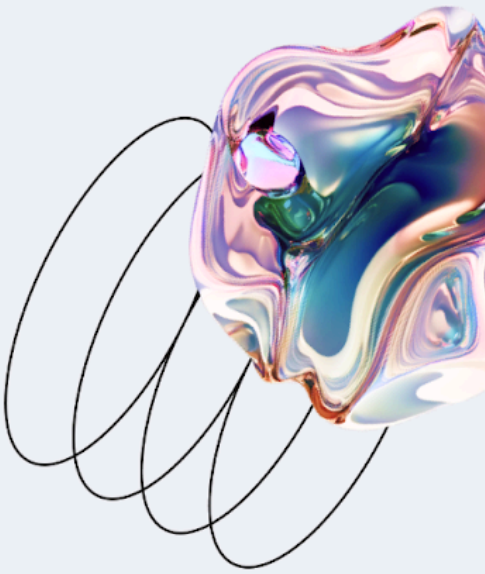
GeekBrains



Машинное обучение. Базовое представление

Data Science

Оглавление

Введение

Всем привет! Это наша пятая лекция на курсе о Data Science. Сегодня мы углубимся в машинное обучение: узнаем, что это, разберем основы, типы алгоритмов и несколько примеров машинного обучения в действии. А также узнаем, в чем разница между искусственным интеллектом и машинным обучением.

Словарь терминов

Искусственный интеллект (Artificial Intelligence, AI) — программа, которая демонстрирует когнитивные способности, подобные человеческим.

Глубокое обучение (Deep learning) — специализированная версия машинного обучения, в которой используются более сложные методы для решения трудных задач.

Обучение с учителем — один из основных типов машинного обучения: ML-алгоритм обучается на размеченных данных.

Машинное обучение

Машинное обучение (Machine Learning, ML) — революционное технологическое достижение последнего десятилетия. В условиях растущей конкуренции машинное обучение позволяет компаниям ускорить цифровую трансформацию и перейти в эпоху автоматизации.

Компаниям не обойтись без искусственного интеллекта и машинного обучения, если они хотят оставаться актуальными в технологичных вертикалях: например, цифровых платежах, обнаружении мошенничества, рекомендательных системах.

ML-алгоритмы распространяются повсеместно. Некоторые компании внедряют машинное обучение в масштабах всей вертикали своего бизнеса.

Сегодня все приложения и программы в интернете используют машинное обучение в той или иной форме. Машинное обучение распространилось настолько, что многие компании теперь решают проблемы с его помощью.

Что такое искусственный интеллект

Чтобы понять, что такое машинное обучение, сперва мы должны рассмотреть основные концепции искусственного интеллекта (ИИ).

Искусственный интеллект (Artificial Intelligence, AI) — это программа, которая демонстрирует когнитивные способности, подобные человеческим. Заставить компьютеры думать и решать проблемы так, как люди, — один из принципов искусственного интеллекта.

ИИ — общий термин для всех программ, способных мыслить так, как люди. Любая программа, которая может самосовершенствоваться, обучаться посредством умозаключений и решать человеческие задачи (например, распознавать изображения и обрабатывать речь), считается формой ИИ.

В область искусственного интеллекта входит машинное обучение и глубокое обучение. Глубокое обучение — это специализированная версия машинного обучения, в которой используются более сложные методы для решения трудных задач.

Между машинным обучением и искусственным интеллектом есть разница: машинное обучение носит вероятностный характер (результаты можно объяснить, исключая природу «черного ящика» ИИ), глубокое обучение является детерминированным.

Процесс самообучения путем сбора новых данных по проблеме позволил алгоритмам машинного обучения захватить корпоративное пространство.

Что такое машинное обучение

Алгоритмы машинного обучения позволили ИИ выйти за рамки простого выполнения задач, для которых он был запрограммирован. До того, как машинное обучение стало мейнстримом, ИИ-программы использовались только для автоматизации низкоуровневых задач в бизнесе и на предприятии. Например, для интеллектуальной автоматизации или простой классификации на основе правил.

Алгоритмы ИИ были ограничены только областью их обработки. Однако благодаря машинному обучению компьютеры смогли выйти за рамки того, что они запрограммировали, и начали развиваться с каждой итерацией.

Машинное обучение принципиально отличается от искусственного интеллекта, поскольку способно развиваться. Используя методы программирования, алгоритмы машинного обучения способны обрабатывать большие объемы данных и извлекать полезную информацию. Они могут улучшить свои предыдущие итерации, изучая предоставленные данные.

Мы не можем говорить о машинном обучении, не говоря о больших данных — важнейшем аспекте ML-алгоритмов. Хорошие результаты работы любого типа ИИ обычно зависят от качества его набора данных, поскольку в этой области широко используются статистические методы.

Машинное обучение — не исключение. Для надежного решения ML требуется хороший поток организованных и разнообразных данных. В современном онлайн-мире у компаний как раз есть доступ к большим данным о своих клиентах.

Большие данные требуют много времени и сложны для обработки по человеческим стандартам. Но данные хорошего качества — лучший корм для обучения ML-алгоритма. Чем больше чистых, пригодных для использования и машиночитаемых данных, тем эффективнее обучение алгоритма.

Алгоритмы машинного обучения способны самосовершенствоваться посредством обучения. Сегодня они обучаются с использованием трех известных методов: обучение с учителем, обучение без учителя и обучение с подкреплением.

Типы ML

Есть разные способы обучения ML-алгоритмов, у каждого из них свои плюсы и минусы. Чтобы их понять, сперва мы должны посмотреть, какие данные они потребляют. В ML есть два типа данных:

- **Размеченные данные** имеют как входные, так и выходные параметры в полностью машиночитаемом шаблоне, но для начала нужно много человеческого труда для маркировки данных.
- **Неразмеченные данные** имеют только один или ни одного из параметров в машиночитаемой форме. Это сводит на нет потребность в человеческом труде, но требует более сложных технических решений.

Разберем три основных метода машинного обучения. Есть и другие, но они используются в специфических случаях.

Обучение с учителем

Обучение с учителем — это подход, когда ML-алгоритм обучается на размеченных данных. Чтобы метод работал, данные должны быть размечены точно. Несмотря на это, обучение с учителем остается чрезвычайно эффективным в правильных обстоятельствах.

При обучении с учителем ML-алгоритму предоставляется небольшой набор обучающих данных — меньшая часть большего набора, которая служит для того, чтобы дать алгоритму базовое представление о проблеме, решении и точках данных, с которыми предстоит работать. Набор данных для обучения очень похож на окончательный набор данных по характеристикам и предоставляет алгоритму размеченные параметры, необходимые для решения задачи.

Затем алгоритм находит отношения между заданными параметрами, по существу устанавливая причинно-следственную связь между переменными в наборе данных. В конце обучения алгоритм имеет представление о том, как устроены данные, и о связи между вводом и выводом.

Затем это решение развертывается для использования с окончательным набором данных. Контролируемые алгоритмы машинного обучения будут продолжать совершенствоваться

даже после развертывания, обнаруживая новые закономерности и взаимосвязи по мере обучения на новых данных.

Обучение без учителя

Преимущество неконтролируемого машинного обучения заключается в возможности работать с немаркированными данными. Это значит, что человеческий труд не требуется для того, чтобы сделать набор данных машиночитаемым, что позволяет программе работать с гораздо большими наборами данных.

В обучении с учителем метки позволяют алгоритму найти точную природу взаимосвязи между любыми двумя точками данных. Однако неконтролируемое обучение не имеет ярлыков, на которые можно было бы опереться, что приводит к созданию скрытых структур. Отношения между точками данных воспринимаются алгоритмом абстрактно, без участия человека.

Создание этих скрытых структур делает алгоритмы обучения без присмотра универсальными. Вместо определенной и заданной постановки задачи, алгоритмы обучения без учителя могут адаптироваться к данным путем динамического изменения скрытых структур. Это предлагает больше возможностей для разработки после развертывания, чем алгоритмы обучения с учителем.

Обучение с подкреплением

Обучение с подкреплением напрямую черпает вдохновение из того, как люди учатся на данных в своей жизни. Алгоритм улучшает сам себя и учится на новых ситуациях, используя метод проб и ошибок. Благоприятные результаты поощряются или «подкрепляются», а неблагоприятные — не поощряются или «наказываются».

Основываясь на психологической концепции кондиционирования, обучение с подкреплением работает, помещая алгоритм в рабочую среду с интерпретатором и системой вознаграждения. На каждой итерации алгоритма выходной результат передается интерпретатору, который решает, является ли результат благоприятным или нет.

Если программа находит правильное решение, интерпретатор подкрепляет его, предоставляя алгоритму вознаграждение. Если результат неблагоприятный, алгоритм вынужден повторяться до тех пор, пока не найдет лучший. В большинстве случаев система вознаграждения напрямую связана с эффективностью результата.

В типичных случаях использования обучения с подкреплением (например, при поиске кратчайшего маршрута между двумя точками на карте), решение не является абсолютным значением. Вместо этого получается показатель эффективности, выраженный в процентах. Чем выше процент, тем больше вознаграждение дается алгоритму. Таким образом, программа обучена давать наилучшее возможное решение за наилучшее возможное вознаграждение.

ML-приложения

Алгоритмы машинного обучения используются в обстоятельствах, когда требуется дальнейшее улучшение решения после развертывания. Динамический характер адаптируемых решений для машинного обучения — один из основных аргументов в пользу их принятия компаниями и организациями в различных сферах.

Алгоритмы и решения машинного обучения универсальны и могут использоваться вместо человеческого труда средней квалификации при определенных обстоятельствах. Например, руководители служб поддержки клиентов в крупных B2C-компаниях используют алгоритмы машинного обучения для обработки естественного языка — чат-боты. Чат-боты могут анализировать запросы клиентов, помогать руководителям службы поддержки клиентов или напрямую общаться с клиентами.

Алгоритмы машинного обучения также помогают улучшить пользовательский опыт и настроить онлайн-платформы. Netflix, Google и Amazon используют системы рекомендаций, чтобы предоставлять интересный контент пользователям на основе их симпатий и антипатий.

Социальные сети используют рекомендательные механизмы для своих новостных лент и рекламных сервисов. Netflix собирает пользовательские данные и рекомендует фильмы и сериалы на основе предпочтений пользователя. Google использует машинное обучение для структурирования своих результатов и для системы рекомендаций YouTube. Amazon использует машинное обучение, чтобы размещать релевантные продукты в поле зрения пользователя, максимизируя коэффициент конверсии, рекомендуя продукты, которые пользователь действительно хочет купить.

AI против ML

Американский профессор Дуглас Хофштадтер сказал: «ИИ — это то, что еще не сделано». Это называется эффектом ИИ, при котором новые методы не только заменяют предыдущие, но и делают последние гораздо более доступными и оптимизированными для использования. По этой логике искусственный интеллект относится к любому прогрессу в области когнитивных компьютеров, а машинное обучение является подмножеством ИИ.

Сегодня термин «искусственный интеллект» используется скорее как общий термин для обозначения технологий, обладающих когнитивными характеристиками, подобными человеческим. Как правило, исследования в области ИИ движутся к более обобщенной форме интеллекта, похожей на то, как малыши думают и воспринимают окружающий мир. Это может означать эволюцию ИИ от программы, специально созданной для одной «узкой» задачи, к решению, развернутому для «общих» задач. Вид деятельности, который мы можем ожидать от людей.

Машинное обучение, с другой стороны, — это эксклюзивное подмножество ИИ, зарезервированное только для алгоритмов, которые могут динамически улучшаться сами по себе. Они не программируются статически для одной задачи, как многие программы ИИ, и

могут быть улучшены даже после их развертывания. Это не только делает их подходящими для корпоративных приложений, но и является новым способом решения проблем в постоянно меняющейся среде.

Машинное обучение также включает в себя глубокое обучение — специализированную дисциплину, в которой заложен ключ к будущему ИИ. Глубокое обучение включает в себя нейронные сети — тип алгоритма, основанный на физической структуре человеческого мозга. Нейронные сети кажутся наиболее продуктивным путем для исследований ИИ, поскольку они позволяют гораздо ближе эмулировать человеческий мозг, чем когда-либо прежде.

Примеры использования ML

Диагностика заболеваний. Если загрузить данные осмотра и диагностики в программу, ее можно научить ставить диагнозы примерно так же, как это делают врачи.

Например, искусственный интеллект Corti прослушивает звонки в скорую помощь и распознает остановку сердца на основе ответов звонящих, их голоса и дыхания. В одном эксперименте программа распознала 93,1% остановок сердца, люди обычно распознают 72,9%. Кроме того, Corti работает быстрее — ставит диагноз за 48 секунд против 79 у диспетчеров-людей.

Сейчас систему внедряют в нескольких европейских городах — она будет работать в службе спасения вместе с диспетчерами.

Автоматические роботизированные операции. Машинное обучение помогает учить медицинских роботов самостоятельно оперировать пациентов, учитывая множество факторов.

Экономия топлива и повышение производительности транспорта. Топливо — одна из главных статей расходов в логистике. С помощью машинного обучения можно сократить его расход: оптимизировать маршруты или понять, как сократить количество автомобилей, сохранив производительность.

Предотвращение сбоев в поставках. Задержка даже одного транспортного средства приводит к сбою во всей цепочке поставок: простоям, потере денег и недовольству клиентов. Машинное обучение помогает этого избежать: предсказывает риски, помогает вовремя их предотвращать и корректировать время доставки с учетом всех факторов.

Оценка кредитоспособности. Обычно в банках кредитоспособность клиента оценивают менеджеры. Сотрудники тратят на оценку много времени и часто ошибаются — отклоняют кредиты тем, кто мог бы их платить, и выдают неплатежеспособным.

Алгоритм можно научить оценивать кредитоспособность клиентов банка. Для этого в него загружают информацию о ранее выданных кредитах: выплачены они или нет, были ли просрочки или досрочное погашение. Все это помогает банку автоматизировать выдачу кредитов.

Борьба с мошенничеством. Банки и их клиенты регулярно теряют деньги из-за мошеннических операций. Распознавать такие операции помогает машинное обучение — специальные алгоритмы учатся выявлять признаки мошеннических операций и вовремя их блокировать.

Минимизация простоев на производстве. Простои из-за поломок, сбоев или нехватки сырья могут стоить заводу миллионы долларов. Машинное обучение помогает их предотвратить. Для этого с датчиков на оборудовании собирают данные, а потом смотрят, при каких показателях возникают сбои. В будущем с помощью этой информации можно предсказать, когда и почему случится простой, как его избежать.

К примеру, может оказаться, что перед поломкой оборудования в цехе всегда поднимается температура. Тогда при повышении температуры система оповестит инженеров, а они вовремя предотвратят проблему.

Выявление угроз безопасности. Машинное обучение помогает сделать производство безопаснее: выявлять незначительные изменения в работе оборудования и вовремя оповещать о возможной катастрофе.

Разведка новых месторождений. Одна из главных проблем нефтегазовой и горнодобывающей промышленности — сложность в обнаружении новых месторождений.

Машинное обучение помогает ускорить этот процесс. На основе данных о прошлых месторождениях искусственный интеллект строит модели, которые с высокой точностью предсказывают, где искать новые залежи газа или руды.

У компании «Газпром» есть проект «Цифровой керн». Это цифровая лаборатория, где анализируют пробы пласта с помощью технологий машинного обучения. Алгоритмы моделируют условия там, откуда взята проба, и помогают создать цифровой двойник месторождения. С его помощью оценивают запасы полезных ископаемых и подбирают индивидуальный подход к разработке. Это позволяет в 1,5−2 раза увеличить добычу полезных ископаемых из конкретного месторождения, а также искать новые.

Регрессия

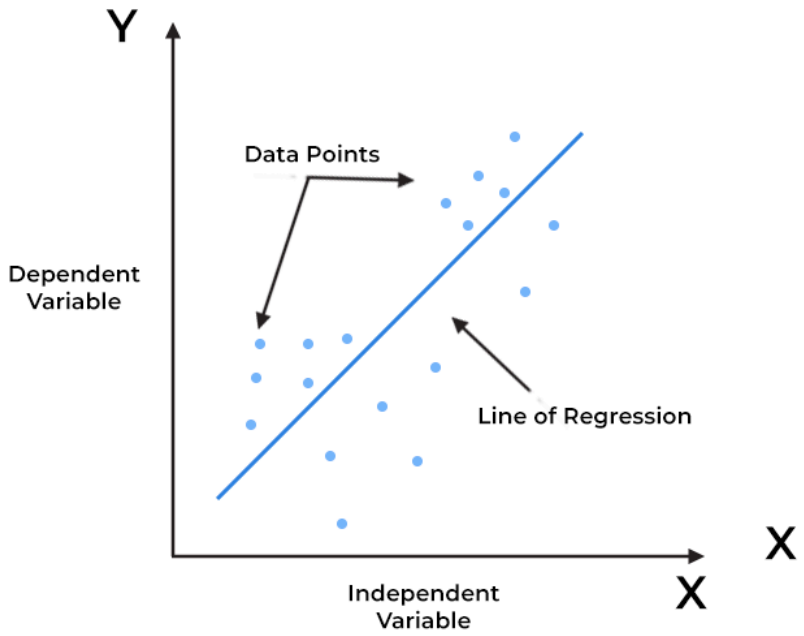
Линейная регрессия — это алгоритм, который обеспечивает линейную связь между независимой и зависимой переменной для прогнозирования исхода будущих событий. Это статистический метод, используемый в науке о данных и машинном обучении для прогнозного анализа.

Независимая переменная также является предиктором или объясняющей переменной, которая остается неизменной из-за изменения других переменных. Однако зависимая переменная изменяется при колебаниях независимой переменной. Модель регрессии предсказывает значение зависимой переменной, которая представляет собой анализируемую или изучаемую переменную ответа или результата.

Таким образом, линейная регрессия — это контролируемый алгоритм обучения, который моделирует математическую связь между переменными и делает прогнозы для непрерывных или числовых переменных, таких как продажи, зарплата, возраст, цена продукта и так далее.

Этот метод анализа выгоден, когда в данных доступны как минимум две переменные, как это наблюдается в прогнозировании фондового рынка, управлении портфелем, научном анализе и так далее.

Наклонная прямая линия представляет модель линейной регрессии.



На рисунке:

Ось X = независимая переменная

Ось Y = выход / зависимая переменная

Линия регрессии = линия наилучшего соответствия модели

Здесь строится линия для заданных точек данных, которые подходят для всех проблем. Следовательно, это называется «линия наилучшего соответствия». Цель алгоритма линейной регрессии — найти эту наиболее подходящую линию, показанную на рисунке выше.

Преимущества линейной регрессии

Линейная регрессия — это популярный статистический инструмент, используемый в науке о данных. Перечислим его основные преимущества.

1. **Простота реализации.** Модель линейной регрессии проста в вычислительном отношении для реализации, поскольку она требует больших инженерных затрат ни перед запуском модели, ни во время ее обслуживания.

2. **Интерпретируемость.** В отличие от других моделей глубокого обучения (нейронных сетей), линейная регрессия относительно проста. Этот алгоритм опережает модели черного ящика, которые не могут объяснить, какая входная переменная вызывает изменение выходной переменной.

3. **Масштабируемость.** Линейная регрессия не требует больших вычислительных ресурсов и, следовательно, хорошо подходит для случаев, когда необходимо масштабирование. Например, модель может хорошо масштабироваться с учетом увеличения объема данных (большие данные).

4. **Оптимально для онлайн-настроек.** Простота вычислений этих алгоритмов позволяет использовать их в онлайн-настройках. Модель можно обучать и переобучать с каждым новым примером, чтобы генерировать прогнозы в режиме реального времени, в отличие от нейронных сетей или машин опорных векторов, которые требуют больших вычислительных ресурсов и значительного времени ожидания для переобучения на новом наборе данных. Все эти факторы делают такие ресурсоемкие модели дорогими и непригодными для приложений реального времени.

Математика

Учитывая простое линейное уравнение y = mx + b, мы можем рассчитать значения MSE:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - (mx_i + b)\right)^2$$

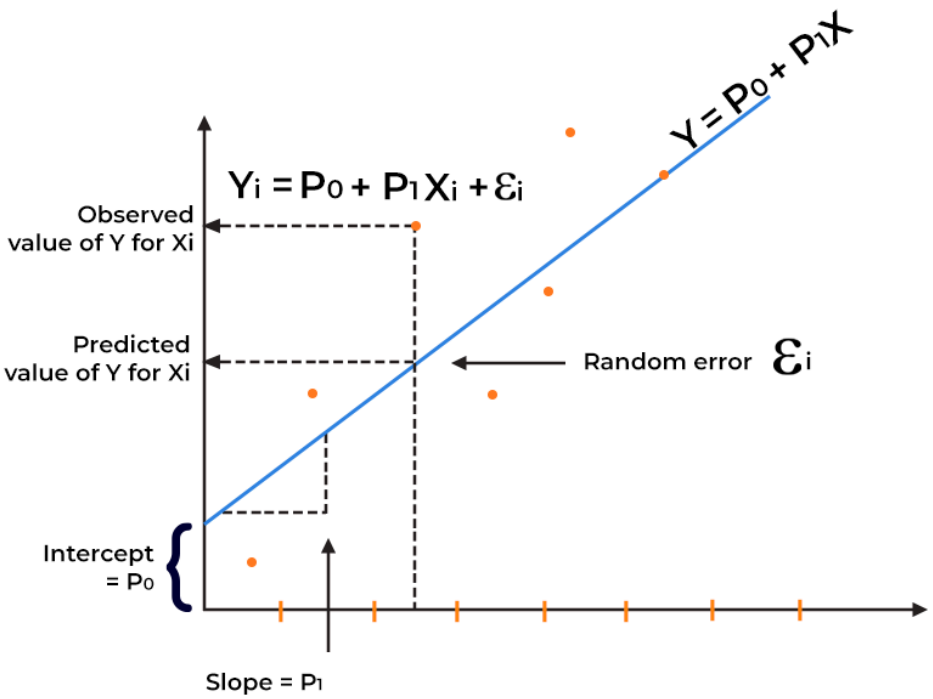
Уравнение для расчета значений MSE

Где:

- N = общее количество наблюдений (точки данных)
- 1/N∑ni=1 = среднее
- y i = фактическое значение наблюдения
- mx i +b = предсказание

Наряду с функцией стоимости алгоритм «градиентного спуска», используется для минимизации MSE и поиска наиболее подходящей линии для данного набора обучающих данных за меньшее количество итераций, тем самым повышая общую эффективность регрессионной модели.

Уравнение линейной регрессии можно представить в виде:



Типы линейной регрессии с примерами

Линейная регрессия была важной движущей силой многих приложений ИИ и науки о данных. Этот статистический метод полезен для бизнеса, поскольку это простой, интерпретируемый и эффективный метод оценки тенденций и создания оценок или прогнозов.

Разберем типы моделей линейной регрессии.

1. Простая линейная регрессия — выявляет корреляцию между зависимой переменной (вход) и независимой переменной (выход). Прежде всего, этот тип регрессии описывает следующее:

- Сила связи между заданными переменными.

  Пример: связь между уровнями загрязнения и повышением температуры.

- Значение зависимой переменной основано на значении независимой переменной.

  Пример: значение уровня загрязнения при определенной температуре.

2. Множественная линейная регрессия — устанавливает связь между независимыми переменными (двумя или более) и соответствующей зависимой переменной. Здесь независимые переменные могут быть либо непрерывными, либо категориальными. Этот тип регрессии помогает предвидеть тенденции, определять будущие значения и предсказывать влияние изменений.

Пример: рассмотрим задачу расчета артериального давления. В этом случае рост, вес и количество упражнений можно считать независимыми переменными. Здесь мы можем использовать множественную линейную регрессию для анализа взаимосвязи между тремя независимыми переменными и одной зависимой переменной, поскольку все рассматриваемые переменные являются количественными.

3. Логистическая регрессия, также называемая логистической моделью, применима в случаях, когда имеется одна зависимая переменная и несколько независимых переменных. Фундаментальное различие между множественной и логистической регрессией заключается в том, что целевая переменная в логистическом подходе является дискретной (двоичной или порядковой). Подразумевается, что зависимая переменная является конечной или категориальной — либо P, либо Q (бинарная регрессия), либо диапазон ограниченных вариантов P, Q, R или S.

Значение переменной ограничено только двумя возможными результатами линейной регрессии. Однако логистическая регрессия решает эту проблему, поскольку может возвращать оценку вероятности, которая показывает шансы любого конкретного события.

Пример: можно определить вероятность выбора предложения на сайте (зависимая переменная). В целях анализа вы можете просмотреть различные характеристики посетителей: сайты, с которых они пришли, количество посещений вашего сайта и активность на вашем сайте (независимые переменные). Это может помочь определить вероятность того, что определенные посетители с большей вероятностью примут предложение. В результате это позволяет вам принимать более обоснованные решения о том, следует ли продвигать предложение на вашем сайте или нет.

Кроме того, логистическая регрессия широко используется в алгоритмах машинного обучения в таких случаях, как обнаружение спама в электронной почте, прогнозирование суммы кредита для клиента и многое другое.

4. Порядковая регрессия — включает одну зависимую дихотомическую переменную и одну независимую переменную, которая может быть порядковой или номинальной. Это облегчает взаимодействие между зависимыми переменными с несколькими упорядоченными уровнями с одной или несколькими независимыми переменными.

Для зависимой переменной с m категориями будет создано (m -1) уравнений. Каждое уравнение имеет разные точки пересечения, но одинаковые коэффициенты наклона для переменных-предикторов. Таким образом, порядковая регрессия создает несколько уравнений прогнозирования для различных категорий. В машинном обучении порядковая регрессия относится к ранжированию обучения или ранжированию, вычисляемому с использованием обобщенной линейной модели (GLM).

Пример: опрос, в котором респонденты должны ответить «согласен» или «не согласен». В некоторых случаях такие ответы бесполезны, так как нельзя сделать окончательный вывод, что усложняет обобщение результатов. Однако вы можете соблюдать естественный порядок категорий, добавляя уровни ответов: согласен, полностью согласен, не согласен и полностью не согласен. Таким образом, порядковая регрессия помогает прогнозировать зависимую

переменную, имеющую «упорядоченные» несколько категорий с использованием независимых переменных.

5. Полиномиальная логистическая регрессия (MLR) — выполняется, когда зависимая переменная является номинальной с более чем двумя уровнями. Определяет взаимосвязь между одной зависимой номинальной переменной и одной или несколькими независимыми переменными непрерывного уровня (интервальными, относительными или дихотомическими). Здесь номинальная переменная относится к переменной без внутреннего порядка.

Пример: полиномиальный логит можно использовать для моделирования программ, выбранных учащимися. Выбор программы в этом случае относится к профессиональной программе, спортивной программе и академической программе. Выбор типа программы можно предсказать, учитывая различные атрибуты, такие как то, насколько хорошо учащиеся могут читать и писать по заданным предметам, пол и полученные ими награды.

Здесь зависимой переменной является выбор программ с несколькими уровнями (неупорядоченными). В таком случае для предсказания используется метод полиномиальной логистической регрессии.

Модели линейной регрессии основаны на простой и понятной математической формуле, которая помогает делать точные прогнозы. Они находят применение в бизнесе и академических областях, таких как социальные науки, менеджмент, экология и вычислительная наука.

На научной основе доказано, что линейная регрессия надежно предсказывает будущие тенденции. Она получила широкое распространение, поскольку эти модели легко интерпретировать, понимать и быстро обучать.

Деревья решений

Что такое деревья решений лучше понять на примере повседневной жизни. Подумайте о том, как часто оказываетесь в ситуациях, когда делаете выбор, основываясь на определенных условиях, где одно решение приводит к определенному результату или последствию.

Деревья решений в жизни

Деревья решений — это, по сути, схематические подходы к решению проблем. Предположим, что во время вождения автомобиля вы доезжаете до перекрестка и вам нужно решить, повернуть налево или направо. Вы примете это решение в зависимости от того, куда едете.

Другие примеры — организация шкафа или покупка автомобиля. Здесь применяется тот же логический пошаговый подход. При покупке автомобиля мы смотрим на разные модели и выбираем одну на основе характеристик: стоимости, производительности, пробеге, типе используемого топлива, внешнем виде и так далее.

Мы применяем логический подход, чтобы разбить сложную ситуацию или набор данных. Такой же подход применяется в деревьях решений.

Подход к деревьям решений

Если нам дали задачу, мы можем использовать графический подход для анализа и объяснения концепции принятия решений на основе условий. Диаграмма будет выглядеть как перевернутое дерево с корнем сверху и раскидистыми ветвями снизу.

Корень — это начальная позиция, где есть набор данных или опций, которые мы анализируем с помощью определенных атрибутов, а затем выбираем действие. В перевернутой древовидной диаграмме корень называется корневым узлом, а ветви, которые представляют результат решения, — конечными узлами.

Диаграмматический подход помогает визуально объяснить другим концепцию вероятности и результата. Если бы мы говорили на простом английском или писали псевдокод (в программном подходе), он был бы записан как операторы «ЕСЛИ… ИНАЧЕ… ЕСЛИ», и количество уровней зависело бы от количества условий. Они часто имеют вложенную или циклическую форму для обработки множества итераций, необходимых для прохождения сложных данных.

Классификация, сегрегация, регрессия

В машинном обучении мы используем деревья решений также для понимания классификации, разделения и получения числового вывода или регрессии.

В автоматизированном процессе мы используем набор алгоритмов и инструментов для выполнения фактического процесса принятия решений и ветвления на основе атрибутов данных. Первоначально несортированные данные — по крайней мере, в соответствии с нашими потребностями — должны быть проанализированы на основе множества атрибутов в несколько этапов и разделены для достижения более низкой случайности или достижения более низкой энтропии.

Выполняя эту сегрегацию (учитывая, что один и тот же атрибут может появляться более одного раза), алгоритм должен учитывать вероятность повторного появления атрибута. Следовательно, мы также можем относиться к дереву решений как к типу дерева вероятностей. Данные в корневом узле довольно случайны, и степень случайности или беспорядочности называется энтропией. По мере того, как мы разбиваем и сортируем данные, мы получаем более высокую степень точности отсортированных данных и достигаем различных степеней информации или «прироста информации».

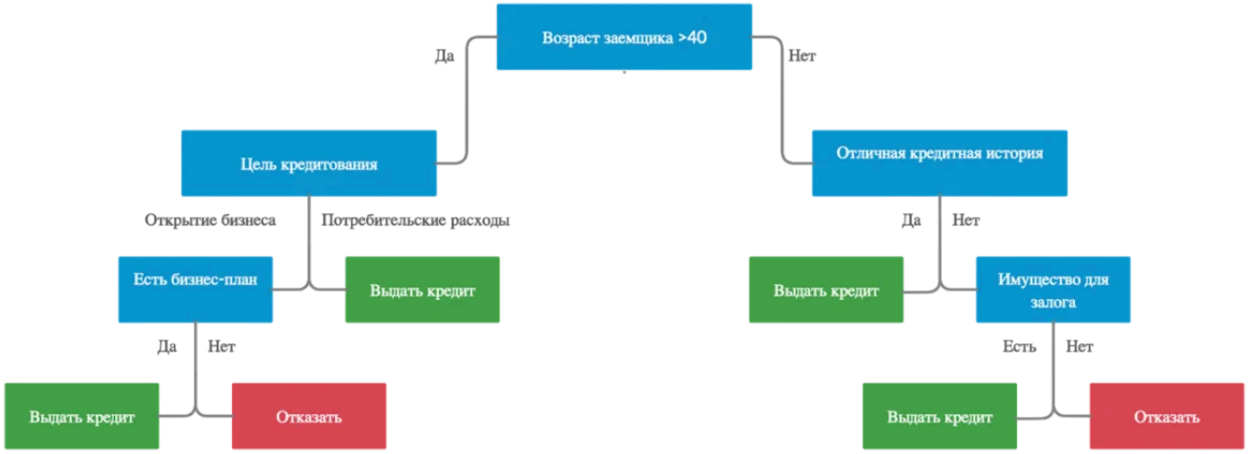
Структура дерева решений

Дерево решений — метод представления решающих правил в определенной иерархии, включающей в себя элементы двух типов — узлов (node) и листьев (leaf). Узлы включают в

себя решающие правила и производят проверку примеров на соответствие выбранного атрибута обучающего множества.

Простой случай: примеры попадают в узел, проходят проверку и разбиваются на два подмножества:

- первое — те, которые удовлетворяют установленное правило;
- второе — те, которые не удовлетворяют установленное правило.



Далее к каждому подмножеству снова применяется правило, процедура повторяется. Это продолжается, пока не будет достигнуто условие остановки алгоритма. Последний узел, когда не осуществляется проверка и разбиение, становится листом.

Лист определяет решение для каждого попавшего в него примера. Для дерева классификации это класс, ассоциируемый с узлом, а для дерева регрессии — соответствующий листу модальный интервал целевой переменной. В листе содержится не правило, а подмножество объектов, удовлетворяющих всем правилам ветви, которая заканчивается этим листом.

Пример попадает в лист, если соответствует всем правилам на пути к нему. К каждому листу есть только один путь. Таким образом, пример может попасть только в один лист, что обеспечивает единственность решения.

Преимущества и недостатки дерева решений

Преимущества:

- Формируют четкие и понятные правила классификации. Например, «если возраст < 40 и нет имущества для залога, то отказать в кредите». То есть деревья решений хорошо и быстро интерпретируются.
- Способны генерировать правила в областях, где специалисту трудно формализовать свои знания.
- Легко визуализируются, то есть могут «интерпретироваться» не только как модель в целом, но и как прогноз для отдельного тестового субъекта (путь в дереве).

- Быстро обучаются и прогнозируют.
- Не требуется много параметров модели.
- Поддерживают как числовые, так и категориальные признаки.

Недостатки:

- Деревья решений чувствительны к шумам во входных данных. Небольшие изменения обучающей выборки могут привести к глобальным корректировкам модели, что скажется на смене правил классификации и интерпретируемости модели.
- Разделяющая граница имеет ограничения, из-за чего дерево решений по качеству классификации уступает другим методам.
- Возможно переобучение дерева решений, из-за чего приходится прибегать к методу «отсечения ветвей», установке минимального числа элементов в листьях дерева или максимальной глубины дерева.
- Сложный поиск оптимального дерева решений: это приводит к необходимости использования эвристики типа жадного поиска признака с максимальным приростом информации, которые в итоге не дают 100-процентной гарантии нахождения оптимального дерева.
- Дерево решений делает константный прогноз для объектов, находящихся в признаковом пространстве вне параллелепипеда, который охватывает не все объекты обучающей выборки.

Где применяют деревья решения?

Модули для построения и исследования деревьев решений входят в состав множества аналитических платформ. Это удобный инструмент, применяемый в системах поддержки принятия решений и интеллектуального анализа данных.

Успешнее всего деревья применяют в следующих областях:

- **Банковское дело.** Для оценки кредитоспособности клиентов банка при выдаче кредитов.
- **Промышленность.** Для контроля качества продукции (обнаружение дефектов в готовых товарах), испытания без нарушений (например, проверка качества сварки) и так далее.
- **Медицина.** Для диагностики заболеваний разной сложности.
- **Молекулярная биология.** Для анализа строения аминокислот.
- **Торговля.** Для классификации клиентов и товара.

Это не все области применения дерева решений. Круг использования постоянно расширяется, а деревья решений постепенно становятся важным инструментом управления бизнес-процессами и поддержки принятия решений.

Заключение

Сегодня мы поговорили о том, какие типы машинного обучения существуют и в чем их принципиальные различия. Мы рассмотрели что же такое линейная регрессия, как строиться модель линейной регрессии. Мы обсудили в каких случаях можно и нужно применять регрессионные модели и какие есть плюсы и минусы у этих моделей. Также мы узнали что такое дерево решений и как оно строится. Надеюсь, вам было интересно и до новых встреч.