Лекция 1.

Введение в машинное обучение

(Казанский (Приволжский) федеральный университет)

М. К. Арабов (cool.araby@mail.ru)

Казань 2024

Актуальность. I

В последнее время наблюдается всплеск данных. Университеты тщательно документируют информацию о своих студентах: их академические достижения, выбранные курсы, результаты экзаменов, прохождение стажировок и практик, темы курсовых работ, а также последующую карьеру. Банки, мобильные операторы, страховые компании и авиакомпании также обладают обширными данными о своих клиентах. Автомобили и самолеты генерируют огромные объемы информации каждую минуту. Как использовать всю эту информацию? Можно проводить аналитику, отслеживать тенденции, искать закономерности в поведении пользователей, например, выявлять неэффективные тарифы или убыточные направления. Однако часто хочется не просто анализировать данные, а использовать их для принятия решений: предлагать конкретные страховки клиентам,

Актуальность. II

оптимизировать работу самолетов, учитывать успехи на олимпиадах при отборе студентов или показывать определенные страницы в поисковой выдаче. Для таких задач не существует четких решений, поскольку мы еще недостаточно понимаем процессы, лежащие в их основе, чтобы создать точные модели, как это делается в физике. Однако у нас есть много данных с примерами успешных решений! Уже известно, какие страховки подходят каждому клиенту и кто из студентов успешно завершил обучение. Именно здесь начинает работать машинное обучение — методы построения моделей на основе данных, а не полного понимания природы явлений. Именно об этом мы поговорим в рамках нашего курса.

1.1. Основные определения. I

Искусственный интеллект (Artificial Intelligence, AI) - это область компьютерных наук, которая занимается созданием систем и программ, способных выполнять задачи, обычно требующие человеческого интеллекта. Целью искусственного интеллекта является разработка алгоритмов и моделей, которые позволяют компьютерам обучаться, делать выводы, принимать решения и выполнять задачи, которые ранее считались возможными только для человека. В области искусственного интеллекта используются различные методы, такие как машинное обучение, нейронные сети, обработка естественного языка и многое другое.

Машинное обучение (Machine Learning) - широкий раздел искусственного интеллекта, который исследует методы создания алгоритмов, способных обучаться. Машинное обучение находится

1.1. Основные определения. II

на пересечении математической статистики, методов оптимизации и классических математических дисциплин, но также имеет свою уникальную специфику, связанную с проблемами вычислительной эффективности и переобучения. Многие методы индуктивного обучения были разработаны как альтернатива традиционным статистическим подходам. Многие из этих методов тесно связаны с технологиями извлечения информации и интеллектуальным анализом данных (Data Mining).

Data Mining (анализ данных) - это процесс исследования и обнаружения скрытых знаний в необработанных данных с использовани алгоритмов и методов искусственного интеллекта. Эти знания ранее были неизвестны, имеют нетривиальное значение, являются практически полезными и могут быть интерпретированы человеком. Термин "Data Mining"был предложен Григорием Пятецким-Шапиро

1.1. Основные определения. III

в 1996 году. Основная цель Data Mining заключается в извлечении знаний из больших объемов данных.

Большие данные - это технологии сбора, обработки и хранения структурированных и неструктурированных массивов информации, отличающихся значительным объемом и высокой скоростью изменений включая работу в реальном времени, что требует специальных инструментов и методов для работы с ними.

Теоретические аспекты машинного обучения объединены в отдельное направление - теорию вычислительного обучения (Computational Learning Theory, COLT).

"Computational Learning Theory"(COLT) - это область науки, которая изучает математические модели и алгоритмы для формализац и понимания процесса обучения компьютерных систем.

Машинное обучение тесно связано с цифровыми технологиями.

1.1. Основные определения. IV

Цифровые технологии (Digital technologies) - это совокупность методов сбора, хранения, обработки, поиска, передачи и представления данных в электронном формате. "Сквозные"цифровые технологии используются для выполнения этих операций и основаны на программн и аппаратном обеспечении, которые создают новые рынки и изменяют бизнес-процессы.

1.2. Типы задач машинного обучения. I

- **Задача регрессии (Regression task)**: – 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.
- **Задача классификации (Classification task)**: – 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.

1.2. Типы задач машинного обучения. II

- **Задача кластеризации (Clustering task)**: – распределение данных на группы: разделение всех клиентов мобильного оператора по уровню платёжеспособности, отнесение космических объектов к той или иной категории (планета, звёзда, чёрная дыра и т. п.).
- **Задача уменьшения размерности (Task of dimensionality reduction)**: – сведение большого числа признаков к меньшему (обычно 2–3) для удобства их последующей визуализаци (например, сжатие данных).
- **Обнаружение объектов (Object Detection)**: Это задача компьютерного зрения, где модель должна определить и классифицировать различные объекты на изображениях или видео. Обнаружение объектов позволяет не только выявлять присутствие объектов, но и точно определять их местоположение в изображении с помощью ограничивающих рамок.

1.2. Типы задач машинного обучения. III

- **Сегментация (Semantic Segmentation)**: В этой задаче каждый пиксель изображения присваивается к определенному классу, что позволяет делать более точное и детальное разделение объектов на изображении. Сегментация используется в медицински исследованиях, автомобильной промышленности, а также для анализа снимков из космоса.
- **Генерация контента (Content Generation)**: Этот тип задач включает создание новых данных, таких как изображения, тексты или звуки с использованием глубоких нейронных сетей. Примерами могут служить генерация фотореалистичных изображе лиц, автоматическое создание текстов или музыкальных композици

1.2. Типы задач машинного обучения. IV

- **Ранжирование (Ranking)**: Задача ранжирования заключается в упорядочивании объектов по их значимости или релевантности для конкретной задачи. Это часто используется в поисковых системах, рекомендательных системах, а также в электронной коммерции для предоставления пользователю наиболее релевантно информации.
- **Обработка естественного языка (Natural Language Processing - NLP)**:- включает в себя широкий спектр задач по анализу и обработке текстовых данных, таких как машинный перевод, анализ тональности текста, генерация текста, извлечение информации, семантический анализ и другие. NLP позволяет компьютерам понимать и взаимодействовать с естественным языком, что находит применение в различных областях, от анализа социальных медиа до автоматизации клиентского обслуживания.

1.2. Типы задач машинного обучения. V

- **Обучение с подкреплением (Reinforcement Learning)**: Этот тип задачи включает в себя обучение агента, который взаимодействует со средой и принимает решения для достижения определенной цели. Агент получает награду или штраф за свои действия, что позволяет ему учиться на основе опыта.
- **Полу-обучение (Semi-Supervised Learning)**: - В этом типе обучения модель обучается на небольшом количестве размеченных данных и большом количестве неразмеченных данных. Это позволяет эффективно использовать доступные данные и повысить качество модели.

1.3. Основные виды машинного обучения. I

Основная часть задач, которые решаются с помощью методов машинного обучения, можно разделить на два основных типа: обучение с учителем (supervised learning) и обучение без учителя (unsupervised learning). Однако "учителем"в данном случае не обязательно является сам программист, который контролирует каждое действие компьютера. В терминах машинного обучения "учителем"является вмешательство человека в процесс обработки информации. В обоих типах обучения модели предоставляются исходные данные, которые анализируются для выявления закономерно
Основное различие заключается в том, что при обучении с учителем имеются гипотезы, которые нужно проверить. Эту разницу легко понять на примерах.

1.3. Основные виды машинного обучения. II

Машинное обучение с учителем (Supervised Learning) - это метод обучения моделей, при котором модель обучается на помеченных данных, где каждый пример имеет входные данные и соответствующие им выходные метки. Этот подход позволяет модели находить закономерности между входными данными и выходными метками, что позволяет делать прогнозы или классификацию на новых данных. Такой тип обучения широко применяется в задачах прогнозиро классификации, и распознавания образов.

Представим, что у нас есть данные о десяти тысячах московских квартир: их площадь, этаж, район, наличие парковки, расстояние от метро, стоимость и другие характеристики. Нам требуется разработать модель, способную предсказывать рыночную стоимость квартиры на основе ее параметров. Этот пример иллюстрирует машинное обучение с учителем: у нас есть исходные данные

1.3. Основные виды машинного обучения. III

(признаки квартир) и соответствующие ответы (их стоимость). Программа должна будет решить задачу регрессии.
Еще один пример: определить наличие рака у пациента, используя его медицинские показатели; определить, является ли электронное письмо спамом, проанализировав его содержание. Все эти задачи относятся к классификации.
Примерами методов машинного обучения с учителем являются:

- **Линейная регрессия**: Используется для прогнозирования числовых значений на основе зависимостей между независимыми и зависимыми переменными.
- **Классификация**: Например, метод опорных векторов (SVM), случайный лес, нейронные сети и др., которые используются для разделения объектов на классы на основе обучающих данных.

1.3. Основные виды машинного обучения. IV

- **Рекомендательные системы**: Алгоритмы, такие как коллаборати фильтрация, используются для предсказания предпочтений пользователей и рекомендации товаров или контента.
- **Обработка естественного языка**: Методы машинного обучения, такие как рекуррентные нейронные сети (RNN) или трансформеры применяются для анализа и генерации текста.

Машинное обучение без учителя (Unsupervised Machine Learning) - это метод обучения моделей, при котором алгоритмы стремятся выявить скрытые закономерности в данных без размеченных примеров. В таких методах модель самостоятельно находит структуру в данных, выявляет паттерны и группы, не имея информации о правильных ответах. Эти методы часто используются для кластеризаци данных, снижения размерности, поиска аномалий и других задач, где требуется выявление структуры в данных без учителя.

1.3. Основные виды машинного обучения. V

- **Кластеризация покупателей в магазине**: Представим, что у нас есть данные о покупках клиентов в магазине - их товарные предпочтения, суммы покупок, частота посещений и т.д. С помощью алгоритмов кластеризации можно выделить группы клиентов с похожими покупательскими привычками. Это позволит магазину провести персонализированные маркетинго кампании, сделать рекомендации и улучшить обслуживание.
- **Снижение размерности данных в анализе изображений**: При работе с изображениями, каждое изображение представлено большим количеством пикселей, что создает высокую размерность данных. Методы снижения размерности, такие как метод главных компонент (PCA), могут помочь сократить количество признаков, сохраняя при этом наиболее значимую информацию, что упрощает анализ и обработку изображений.

1.3. Основные виды машинного обучения. VI

- **Обнаружение аномалий в финансовых транзакциях**: Путем анализа обычных шаблонов финансовых транзакций (суммы, частоты, местоположения и т.д.) можно выявить аномальные операции, которые могут указывать на мошенническую деятельность. Алгоритмы обнаружения аномалий помогут автоматически выявлять подозрительные транзакции для последующего расследования.

Обучение с подкреплением (reinforcement learning) - это тип машинного обучения, в котором агент принимает решения в определенной среде с целью максимизации некоторой награды. Агент обучается на основе опыта, получая обратную связь в виде награды или штрафа за свои действия.
Некоторые из основных алгоритмов обучения с подкреплением включают в себя:

1.3. Основные виды машинного обучения. VII

- **Q-Learning**: Этот алгоритм используется для обучения агента, который принимает решения в среде с дискретным пространством действий. Q-Learning обновляет значения Q-функции, которая оценивает ожидаемую награду для выполнения определенного действия в конкретном состоянии.
- **Deep Q-Networks (DQN)**: DQN - это метод, который комбинирует Q-Learning с глубокими нейронными сетями. Он позволяет обучать агента в средах с большим пространством состояний и действий, используя нейронную сеть для оценки Q-функции.
- **Policy Gradient**: В отличие от Q-Learning, этот метод напрямую оптимизирует стратегию агента, не требуя оценки Q-функции. Policy Gradient ищет направление изменения стратегии, которое увеличит ожидаемую награду.

1.3. Основные виды машинного обучения. VIII

- **Actor-Critic**: Этот подход комбинирует элементы Policy Gradient и функции ценности (Value Function). Актер (Actor) обновляет стратегию агента, в то время как Критик (Critic) оценивает качество действий и помогает улучшить стратегию.
- **Proximal Policy Optimization (PPO)**: PPO - это метод обучения с подкреплением, который обновляет стратегию агента таким образом, чтобы изменения были ограничены (proximal). Это помогает стабилизировать обучение и предотвраща сильные изменения в стратегии.
- **Deep Deterministic Policy Gradient (DDPG)**: DDPG - это алгоритм, предназначенный для задач с непрерывным пространств действий. Он комбинирует элементы Actor-Critic методов с использованием глубоких нейронных сетей для обучения агента в сложных средах.

1.3. Основные виды машинного обучения. IX

Каждый из этих алгоритмов имеет свои преимущества и недостатки, и выбор конкретного метода зависит от характеристик задачи обучения с подкреплением.

1.3. Основные виды машинного обучения. X

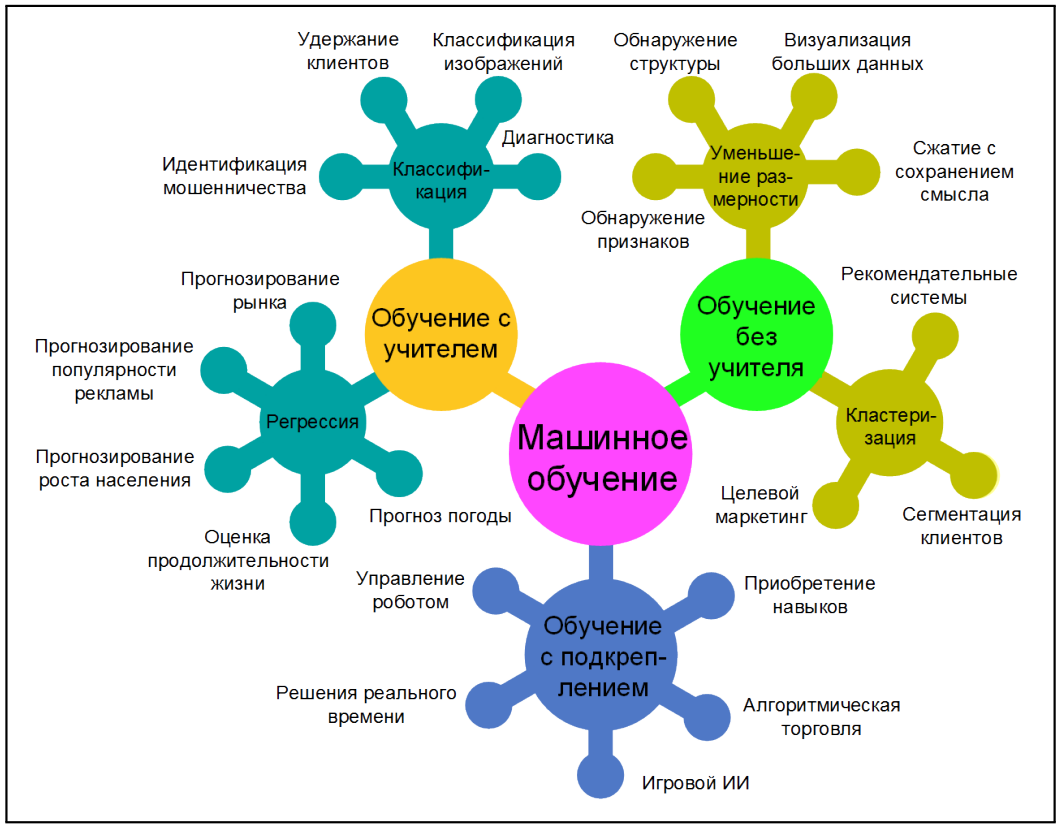

Рис.: 1. Основные виды машинного обучения.

1.4 Основные алгоритмы моделей машинного обучения I

Дерево принятия решений (Decision tree):- это метод поддержки принятия решений, основанный на построении древовидной структуры, где каждый узел представляет вопрос или решение, а каждое ребро - возможный результат ответа на этот вопрос.
Этот метод позволяет систематизировать проблему путем последовател представления вопросов, на которые можно ответить "да"или "нет и в конечном итоге прийти к правильному решению. Дерево принятия решений структурирует процесс принятия решений, что облегчает анализ данных и принятие логически обоснованных выводов.
Для бизнес-процессов дерево принятия решений может быть использовано для оптимизации процессов, прогнозирования результато или классификации данных на основе задаваемых вопросов. Важно

1.4 Основные алгоритмы моделей машинного обучения II

правильно построить дерево и выбрать оптимальные вопросы, чтобы получить точные и информативные результаты.
Наивный байесовский классификатор (Naive Bayes classifier):- это простой вероятностный классификатор, основанный на теореме Байеса. Он делает строгое предположение о независимости между признаками (наивное предположение), что позволяет эффективно работать с большими объемами данных. Этот метод на практике применяется в различных областях машинного обучения:

- Определение спама в электронной почте;
- Автоматическое размещение новостных статей по темам;
- Анализ тональности текста для выявления эмоциональной окраски;

1.4 Основные алгоритмы моделей машинного обучения III

- Распознавание лиц и других паттернов на изображениях.

Наивный байесовский классификатор широко используется из-за своей простоты и эффективности в решении различных задач классификации и анализа данных.

Метод наименьших квадратов (Least square method-LSA):
- является одним из методов реализации линейной регрессии, который используется для подгонки прямой, проходящей через набор точек. Этот метод заключается в минимизации суммы квадратов расстояний от каждой точки до линии регрессии. Путем нахождения такой линии, сумма расстояний будет наименьшей, что позволяет получить наилучшую подгонку.

Линейная регрессия с использованием метода наименьших квадратов часто применяется в задачах машинного обучения для анализа и

1.4 Основные алгоритмы моделей машинного обучения IV

прогнозирования данных. Этот метод помогает минимизировать ошибки и создать метрику ошибок для определения оптимальной линии, проходящей через точки с минимальным отклонением от истинных значений.

Логистическая регрессия (Logistic regression): - это метод определения связи между переменными, где одна переменная является категориальной зависимой, а другие независимыми. Для этого используется логистическая функция (логистическое кумулятивное распределение). Логистическая регрессия является мощным статистическим методом прогнозирования событий с использованием одной или нескольких независимых переменных. Этот метод имеет практическое применение в различных областях, включая:

1.4 Основные алгоритмы моделей машинного обучения V

- Кредитный скоринг;
- Оценка эффективности рекламных кампаний;
- Прогнозирование прибыли от определенного товара;
- Оценка вероятности возникновения землетрясения в конкретную дату.

Логистическая регрессия широко используется для классификации и предсказания вероятностей в различных областях, где требуется оценка вероятности событий на основе независимых переменных.
Метод опорных векторов (Support Vector Machine - SVM):
- это набор алгоритмов, используемых для классификации и регрессионного анализа. В случае, когда объекты в N-мерном пространстве принадлежат к одному из двух классов, SVM строит

1.4 Основные алгоритмы моделей машинного обучения VI

гиперплоскость размерности (N - 1), чтобы разделить объекты на две группы. Гиперплоскость создается таким образом, чтобы быть максимально удаленной от ближайших точек каждой группы. SVM и его вариации помогают решать сложные задачи машинного обучения, такие как сплайсинг ДНК, определение пола по фотографии или показ рекламных баннеров на веб-сайтах. Этот метод эффективен в построении моделей для различных задач классификации и регрессии, где требуется точное разделение объектов на различные классы.

Метод ансамблей (Ensemble method): - это подход в машинном обучении, который основан на создании множества классификаторов и объединении их результатов путем усреднения или голосования. Начально метод ансамблей был частным случаем байесовского

1.4 Основные алгоритмы моделей машинного обучения VII

усреднения, но позже стал более сложным и включил дополнительные алгоритмы:

- Бустинг (boosting) - превращает слабые модели в сильные, комбинируя их в ансамбль классификаторов;
- Бэггинг (bagging) - собирает усложненные классификаторы, обучая их параллельно;
- Корректирование ошибок выходного кодирования.

Метод ансамблей является более мощным инструментом по сравнению с отдельными моделями прогнозирования, потому что:

- Минимизирует влияние случайностей путем усреднения ошибок каждого базового классификатора;

1.4 Основные алгоритмы моделей машинного обучения VIII

- Снижает дисперсию, так как несколько различных моделей, основанных на разных гипотезах, имеют больше шансов на правильный результат;
- Исключает "выход за пределы множества": если агрегированная гипотеза оказывается за пределами множества базовых гипотез, то метод расширяет множество базовых гипотез для включения этой гипотезы.

Алгоритмы кластеризации (Clustering algorithms): - это методы, позволяющие разделить множество объектов на кластеры таким образом, чтобы объекты внутри каждого кластера были максимально похожи друг на друга.
Существует несколько алгоритмов кластеризации, включая:

1.4 Основные алгоритмы моделей машинного обучения IX

- На основе центра тяжести треугольника;
- На базе подключения;
- Сокращение размерности;
- Основанные на плотности (пространственная кластеризация);
- Вероятностные;
- Машинное обучение, включая нейронные сети.

1.4 Основные алгоритмы моделей машинного обучения

X

Алгоритмы кластеризации применяются в различных областях, таких как биология (анализ геномов, социология (обработка результатс социологических исследований), информационные технологии и многие другие, чтобы выделить группы схожих объектов для дальнейшего анализа.

Метод главных компонент (Principal Component Analysis, PCA) : - это метод машинного обучения без учителя, который используется для снижения размерности данных путем преобразования их в новое пространство признаков. Цель PCA заключается в том, чтобы найти линейные комбинации исходных признаков, которые объясняют наибольшую дисперсию данных.

Применение PCA позволяет уменьшить размерность данных, сохраняя при этом наиболее значимые характеристики. Это может помочь

1.4 Основные алгоритмы моделей машинного обучения XI

улучшить производительность моделей машинного обучения, уменьшит избыточность данных и улучшить их интерпретируемость.

Сингулярное разложение (Singular Value Decomposition, SVD):
- это метод линейной алгебры, который используется для разложения прямоугольной матрицы на унитарные матрицы и диагональную матрицу. Этот метод часто применяется в задачах анализа данных, сжатия информации, рекомендательных системах и других областях. Одним из частных случаев сингулярного разложения является метод главных компонент (PCA), который используется для снижения размерности данных и выделения наиболее значимых признаков. Сингулярное разложение и PCA были широко использованы в ранних технологиях компьютерного зрения для анализа данных и поиска паттернов, например, в распознавании лиц.

1.4 Основные алгоритмы моделей машинного обучения XII

Современные алгоритмы сингулярного разложения в машинном обучении стали более сложными и эффективными, но основные принципы их работы остаются теми же - разложение матрицы на более простые компоненты для анализа и извлечения информации из данных.

Анализ независимых компонент (Independent Component Analysis, ICA): - это статистический метод, который помогает выявить скрытые факторы, влияющие на случайные величины или сигналы. ICA строит модель, описывающую данные как смесь независимых компонентов, которые считаются негауссовскими сигналами.

В отличие от метода главных компонент, связанного с линейной зависимостью данных, ICA ищет независимые компоненты в данных,

1.4 Основные алгоритмы моделей машинного обучения XIII

что делает его более эффективным в обнаружении скрытых причин явлений. Этот метод нашел широкое применение в различных областях, таких как астрономия, медицина, распознавание речи, финансовый анализ и другие.

1.5. Примеры применения в реальной жизни I

Пример 1. Диагностика заболеваний
В данном случае пациенты рассматриваются как объекты, а признаками являются все наблюдаемые у них симптомы, анамнез, результаты анализов, уже предпринятые лечебные меры (по сути, вся история болезни, структурированная и разделенная на отдельные критерии). Некоторые признаки, такие как пол, наличие или отсутствие головной боли, кашля, сыпи и другие, рассматриваются как бинарные. Оценка тяжести состояния (крайне тяжелое, средней тяжести и т. д.) является порядковым признаком, в то время как многие другие – количественными: доза лекарственного препарата, уровень гемоглобина в крови, показатели артериального давления и пульса, возраст, вес. Собрав информацию о состоянии пациента, включающую множество таких признаков, ее можно загрузить в компьютер

1.5. Примеры применения в реальной жизни II

и с помощью программы, способной к машинному обучению, выполнить следующие задачи:

- провести дифференциальную диагностику (определение вида заболевания);
- выбрать наиболее оптимальную стратегию лечения;
- спрогнозировать развитие болезни, ее длительность и исход;
- рассчитать риск возможных осложнений;
- выявить синдромы – наборы симптомов, сопутствующие данному заболеванию или нарушению.

1.5. Примеры применения в реальной жизни III

Ни один врач не в состоянии обработать весь объем информации по каждому пациенту мгновенно, обобщить большое количество других подобных историй болезни и немедленно выдать четкий результат. Поэтому машинное обучение становится для врачей незаменимым помощником.

Пример 2. Поиск месторождений полезных ископаемых

Здесь информация, полученная через геологическую разведку, выступает в качестве признаков: наличие определенных пород на территории (бинарный признак), их физические и химические свойства (которые можно разделить на количественные и качественные признаки).

Для обучающего набора данных используются два типа прецедентов: районы с известными месторождениями полезных ископаемых и районы с похожими характеристиками, где такие ископаемые не

1.5. Примеры применения в реальной жизни IV

были обнаружены. Однако добыча редких полезных ископаемых имеет свои особенности: в большинстве случаев количество признаков превышает число объектов, и традиционные методы статистики неэффективны в таких ситуациях.

Поэтому в машинном обучении акцент делается на выявление закономерностей в имеющихся данных. Для этого определяются небольшие и наиболее информативные группы признаков, которые существенно влияют на ответ на вопрос исследования – присутствует ли определенное ископаемое в данной местности. Можно провести аналогию с медициной: у месторождений также можно выделить свои собственные "синдромы". Применение машинного обучения в этой области ценно не только с практической точки зрения, но и представляет значительный научный интерес для геологов и геофизиков.

1.5. Примеры применения в реальной жизни V

Пример 3. Оценка надежности и платежеспособности кандидатов на получение кредитов

Банки, выдающие кредиты, ежедневно сталкиваются с этой задачей. Необходимость автоматизации процесса возникла давно, уже в 1960-1970-х годах, когда в США и других странах стал популярен выпуск кредитных карт.

Лица, обращающиеся в банк за заемом, рассматриваются как объекты, а признаки различаются в зависимости от того, является ли заявитель физическим или юридическим лицом. Данные о частном лице, запрашивающем кредит, формируются на основе заполненной им анкеты. После этого анкета дополняется другими данными о клиенте, полученными банком.

Некоторые из них относятся к бинарным признакам (пол, наличие телефонного номера), другие - к порядковым (образование, должность)

1.5. Примеры применения в реальной жизни VI

а большинство являются количественными (сумма кредита, задолженн в других банках, возраст, семейное положение, доход, трудовой стаж) или номинальными (имя, название работодателя, профессия, адрес).

Для машинного обучения создается выборка с клиентами, у которых известна кредитная история. Заемщики разделяются на классы, обычно два: "хорошие"и "плохие"заемщики, и решение о выдаче кредита принимается в пользу "хороших".

Более сложный алгоритм машинного обучения, известный как кредитный скоринг, предполагает присвоение каждому заемщику условных баллов за каждый признак, и решение о кредите зависит от общей суммы баллов. Системы кредитного скоринга начисляют баллы для каждого признака и определяют условия кредитования

1.5. Примеры применения в реальной жизни VII

(срок, процентная ставка и другие параметры). Кроме того, существует алгоритм обучения системы на основе прецедентов.

1.6. Задачи ML. I

Надо признаться, что обсуждение терминологии вряд ли нам сильно поможет, поэтому лучше перейдем к конкретному примеру. Когда Алан Тьюринг работал над первыми компьютерами, он пытался расшифровать сообщения немецких военных, закодированные машиной Энигма. Поиск расшифровки требовал перебора массы вариантов. Люди с этой задачей справлялись плохо, зато машина могла решить её сравнительно быстро. Очевидно, далеко не для каждой задачи, с которой люди справляются с трудом, можно написать программу для эффективного поиска решения. Более того, есть целый класс задач (так называемые NP-трудные задачи), которые нельзя решить за разумное время. Можно даже явно доказать, что никакой компьютер здесь чуда тоже не совершит. Самое интересное это то, что бывают и задачи, которые для

1.6. Задачи ML. II

людей особенного труда не составляют, но которые почему-то крайне трудно запрограммировать, например:

- перевести текст с одного языка на другой;
- диагностировать болезнь по симптомам;
- сравнить, какой из двух документов в интернете лучше подходит под данный поисковый запрос;
- сказать, что изображено на картинке;
- оценить, по какой цене удастся продать квартиру.

1.6. Задачи ML. III

У всех этих задач есть много общего. Во-первых, их решение можно записать как функцию, которая отображает объекты или примеры (samples) в предсказания (targets). Например, больных надо отобразить в диагнозы, а документы в оценку релевантности. Во-вторых, вряд ли у этих задач есть единственно верное, идеальное решение. Даже профессиональные переводчики могут по-разному перевести один и тот же текст, и оба перевода будут верными. Так что лучшее в этих задачах — враг хорошего. В конце концов, и доктора иногда делают ошибки в диагнозах, и вы не всегда можете сказать, что же именно изображено на картинке. В-третьих, у нас есть много примеров правильных ответов (скажем, переводов предложения на другой язык или подписей к заданной картинке), а примеры неправильных ответов (если они нужны), как правило, не составляет труда сконструировать.

1.6. Задачи ML. IV

Мы назовём функцию, отображающую объекты в предсказания, — моделью, а имеющийся у нас набор примеров — обучающей выборкой или датасетом. Обучающая выборка состоит из:

- объектов (к примеру, скачанные из интернета картинки, истории больных, активность пользователей сервиса и так далее);
- и ответов (подписи к картинкам, диагнозы, информация об уходе пользователей с сервиса), которые мы также будем иногда называть таргетами.

Описанные выше задачи являются примерами задач обучения с учителем (supervised learning), так как правильные ответы для каждого объекта обучающей выборки заранее известны. Задачи обучения с учителем делятся на следующие виды в зависимости от множества всех возможных ответов (таргетов):

1.6. Задачи ML. V

- $\mathbb{Y} = \mathbb{R}$ или $\mathbb{Y} = \mathbb{R}^{\mathbb{N}}$ — регрессия. Примерами задач регрессии являются предсказание продолжительности поездки на каршеринг спрос на конкретный товар в конкретный день или погода в вашем городе на завтра (температура, влажность и давление — это несколько вещественных чисел, которые формируют вектор предсказания).
- $\mathbb{Y} = 0, 1$ — бинарная классификация. Например, мы можем предсказывать, кликнет ли пользователь по рекламному объявлени вернёт ли клиент кредит в установленный срок, сдаст ли студент сессию, случится ли определённое заболевание у пациента, есть ли на картинке банан.
- $\mathbb{Y} = 1, 2, ..., K$ — многоклассовая классификация. Например, определение предметной области для научной статьи (математика биология, психология и т. д.).

1.6. Задачи ML. VI

- $\mathbb{Y} = 0, 1^K$ — многоклассовая классификация с пересекающимися классами (multilabel classification). Например, задача автоматическ проставления тегов для ресторанов (логично, что ресторан может одновременно иметь несколько тегов).
- $\mathbb{Y}$ — конечное упорядоченное множество — ранжирование. Основным примером является задача ранжирования поисковой выдачи, где для любого запроса нужно отсортировать все возможные документы по релевантности этому запросу; при этом оценка релевантности имеет смысл только в контексте сравнения двух документов между собой, её абсолютное значение информации не несёт.

1.6. Задачи ML. VII

- **Частичное обучение (semi-supervised learning)** — задача, в которой для одной части объектов обучающей выборки известны и признаки, и ответы, а для другой только признаки. Такие ситуации возникают, например, в медицинских задачах, где получение ответа является крайне сложным (например, требует проведения дорогостоящего анализа).

Существует небольшой класс задач, связанных с обучением без учителя (unsupervised learning), где известны только данные, а ответы неизвестны или вообще не существуют. В таких задачах поиск "правильных"ответов не является главной целью. Классическим примером обучения без учителя является кластеризация — процесс разделения объектов на группы с некоторыми неизвестными, но, предположительно, интерпретируемыми свойствами.

1.6. Задачи ML. VIII

- Кластеризация — задача разделения объектов на группы, обладающие некоторыми свойствами. Примером может служить кластеризация документов из электронной библиотеки или кластеризация абонентов мобильного оператора.
- Оценивание плотности — задача приближения распределения объектов. Примером может служить задача обнаружения аномалий, в которой на этапе обучения известны лишь примеры «правильного» поведения оборудования (или, скажем, игроков на бирже), а в дальнейшем требуется обнаруживать случаи некорректной работы (соответственно, незаконного поведения игроков). В таких задачах сначала оценивается распределение «правильных» объектов, а затем аномальными объявляются все объекты, которые в рамках этого распределения получают слишком низкую вероятность.

1.6. Задачи ML. IX

- Визуализация — задача изображения многомерных объектов в двумерном или трехмерном пространстве таким образом, чтобы сохранялось как можно больше зависимостей и отношений между ними.
- Понижение размерности — задача генерации таких новых признаков, что их меньше, чем исходных, но при этом с их помощью задача решается не хуже (или с небольшими потерями качества, или лучше — зависит от постановки). К этой же категории относится задача построения латентных моделей, где требуется описать процесс генерации данных с помощью некоторого (как правило, небольшого) набора скрытых переменных. Примерами являются задачи тематического моделирования и построения рекомендаций.

1.6. Задачи ML. X

Существуют и другие виды (и даже парадигмы) машинного обучения. Если вы столкнетесь с задачей, которую сложно отнести к одному из перечисленных типов, не отчаивайтесь, так как в рамках данного курса мы обязательно рассмотрим и такие задачи.
Бывают и более сложные постановки, например, обучение с подкреплен (reinforcement learning), где алгоритм на каждом шаге наблюдает ситуацию, выбирает одно из доступных действий, получает награду и корректирует свою стратегию. Задачей алгоритма при этом является максимизация награды в некотором смысле. В такую постановку хорошо вписывается, например, задача создания беспилотн автомобиля: машина видит текущее окружение за счёт сенсоров и должна решить, как сейчас повернуть руль, как сильно ускориться или затормозить. При этом она должна приехать в конкретную точку, не нарушая правила, и согласно этим требованиям в каждый

1.6. Задачи ML. XI

момент она получает некоторую награду. Однако задача обучения с подкреплением очень сложна, и пока что построение беспилотных автомобилей исключительно на таком подходе скорее является исследовательским вопросом.

1.7. Данные I

Машинное обучение начинается с данных. Важно, чтобы их было достаточно много и чтобы они были достаточно качественными. Некоторые проекты приходится откладывать на неопределенный срок из-за того, что просто невозможно собрать данные.
Чем сложнее задача, тем больше данных нужно, чтобы ее решить. Например, существенные успехи в задачах распознавания изображений были достигнуты лишь с появлением очень больших датасетов (и, стоит добавить, вычислительных мощностей). Вычислительные ресурсы продолжают совершенствовать, но во многих ситуациях размеченных данных (то есть объектов, которым кто-то сопоставил ответ) было бы по-прежнему слишком мало: например, для решения задачи аннотирования изображений (image captioning) потребовалось бы огромное количество пар (изображение, описание). В некоторых

1.7. Данные II

случаях можно воспользоваться открытыми датасетами. Сейчас их доступно довольно много и некоторые весьма велики, но чаще всего они создаются для довольно простых задач, например, для задачи классификации изображений. Иногда датасет можно купить. Но для каких-то задач вы нигде не найдете данных. Скажем, авторам неизвестно больших и качественных корпусов телефонных разговоров с расшифровками – в том числе и по причинам конфиденциальности таких данных.

Бороться с проблемой нехватки данных можно двумя способами. Первый – использование краудсорсинга, то есть привлечение людей, готовых разметить много данных. Во многих ситуациях (например, когда речь заходит об оценке поисковой выдачи) без дополнительной разметки никак не обойтись. Некоторые проекты,

1.7. Данные III

в первую очередь научные и социальные, используют также citizen science – разметку данных волонтёрами без какого-либо вознагражд просто за чувство причастности к доброму делу исследования животных Африки или формы галактик.
Второй способ заключается в использовании неразмеченных данных. Например, в задаче аннотирования изображений у нас есть огромное количество изображений и текстов, которые не связаны друг с другом. Тем не менее, мы можем использовать их, чтобы помочь компьютеру понять, какие слова могут сочетаться в предложении. Подходы, связанные с использованием неразмеченных данных для решения задач обучения с учителем, объединяются термином self-supervised learning и широко применяются в настоящее время. Важным аспектом является обучение представлений (representation learning) - процесс создания компактных векторов небольшой

1.7. Данные IV

размерности из сложных данных, таких как изображения, звуки, тексты, графы, чтобы данные схожие по структуре или семантике имели похожие представления. Это можно сделать различными способами - например, используя фрагменты моделей, обученных для решения других задач, или создавая модель, предсказывающую скрытую часть объекта по оставшейся части - например, пропущенное слово в предложении. Этому будет посвящена отдельная глава нашего учебника.

Однако, помимо количества данных также важно их качество и удобство для анализа. Давайте разберемся, что это означает и какие проблемы могут возникнуть.

Для работы с объектом модель должна опираться на его свойства, такие как доход человека, цвет левого верхнего пикселя на изображении или частоту встречаемости слова "интеграл"в тексте. Эти свойства

1.7. Данные V

обычно называются признаками, а совокупность свойств, которые мы выделяем у объекта - это его признаковое описание.
Перечислим несколько простых и распространенных видов признаков:

- **Численные признаки**: Например, рост или доход. Иногда их делят на вещественные (например, рост в сантиметрах) и целочисленные (например, количество детей).
- **Категориальные признаки**: Принимают значения из дискретного множества. Например, профессия человека или день недели.
- **Бинарные признаки**: Принимают два значения, например, "да"и "нет "истина"и "ложь". Их можно обрабатывать как численные или как категориальные признаки.

1.7. Данные VI

- **Ординальные признаки**: Это подтип категориальных признаков, которые имеют упорядоченное значение из дискретного множества. Например, класс опасности химического вещества (от 1 до 4) или год обучения студента.

Приходится иметь дело и с более сложными признаками. Например, описание ресторана может содержать тексты отзывов или фотографии, а профиль человека в социальной сети - список его друзей. Для многих типов данных, таких как изображения, видео, тексты, звук, графы, существует множество методов извлечения признаков, преимущественно нейронных сетей. Более подробную информацию об этом можно найти в разделах о нейронных сетях для соответствующ типов данных. В случае сложных данных может потребоваться дополнительное усилие для извлечения признаков - этот процесс называется feature engineering.

1.7. Данные VII

Удобно представить данные в виде таблицы, где строки соответствуют объектам, а столбцы - признакам. Например:

Имя	Возраст	Пол	Город
Анна	25	Ж	Москва
Иван	30	М	Санкт-Петербург
Мария	28	Ж	Киев

Данные, представленные в таком виде, называются табличными.
Табличные данные – один из самых удобных для анализа форматов.
Успешные пайплайны работы существуют также для текстов, звука, изображений, видео и графов.
Идеально, если все признаки являются численными. Тогда с таблицей можно работать как с объектом линейной алгебры – матрицей объекты-признаки.

1.7. Данные VIII

Создание информативного признакового описания очень важно для дальнейшего анализа. Однако необходимо следить за качеством данных. В процессе работы могут возникнуть следующие проблемы:

- **Пропуски (пропущенные значения)**: объекты или признаки с пропусками можно удалять из выборки, но при множестве пропусков мы рискуем потерять слишком много информации. Наличие пропусков может также содержать информацию о систематических проблемах в сборе данных.
- **Выбросы**: объекты, значительно отличающиеся от большинства, могут возникать из-за ошибок в данных или представлять реальные аномалии. Обычно выбросы удаляют, но в некоторых случаях они могут быть важны и требуют отдельной обработки.

1.7. Данные IX

- **Ошибки разметки**: при сборе данных от людей возможны ошибки в разметке, что может повлиять на качество данных.
- Data drift: с течением времени данные могут меняться, что требует мониторинга и обновления модели.

Кроме указанных проблем, могут возникать и другие сложности, которые требуют внимания и решения.
Данные, которые используются в машинном обучении, называются датасетом. Датасет (dataset) - это набор данных, который представляет собой структурированную коллекцию информации, обычно представле в виде таблицы или файла. Датасеты используются в машинном обучении и анализе данных для обучения моделей и извлечения полезной информации.
В датасете для машинного обучения обычно выделяют три основных подмножества данных:

1.7. Данные X

- **Обучающая выборка (Training set)**: Это подмножество данных, которое используется для обучения модели. Модель "учится"на этих данных путем подбора параметров и поиска закономерностей в данных.
- **Валидационная выборка (Validation set)**: Это подмножество данных, которое используется для настройки гиперпараметров модели и оценки ее производительности во время обучения. Валидационная выборка помогает выбирать лучшую модель из нескольких вариантов и предотвращает переобучение.

1.7. Данные XI

- **Тестовая выборка (Test set)**: Это отдельное подмножество данных, которое не используется ни во время обучения, ни во время настройки гиперпараметров. Она используется для окончательной оценки производительности модели после завершен обучения. Тестовая выборка позволяет оценить, насколько хорошо модель будет работать на новых, ранее не виденных данных.

Разделение датасета на эти три части помогает оценить производитель модели и ее способность обобщаться на новые данные.
Обычно применяют следующее соотношение для разделения датасета на обучающую, валидационную и тестовую выборки:

- **Обучающая выборка (Training set)**: ЭПримерно 60-80% от всего датасета.

1.7. Данные XII

- **Валидационная выборка (Validation set)**: Примерно 60-80% от всего датасета.
- **Тестовая выборка (Test set)**: Около 10-20% от всего датасета.

Рис.: 2. Разделение датасета.

1.7. Данные XIII

Эти процентные соотношения могут варьироваться в зависимости от размера и характера данных, а также от конкретной задачи машинного обучения.

1.8. Модель и алгоритм обучения I

Модель — это способ описания реальности. Например, "Земля плоская"— это модель, которая может быть полезна в небольшом масштабе, где кривизну земли можно игнорировать. Однако для расчета пути из Парижа в Лос-Анджелес модель плоской Земли будет недостаточно точной и придется заменить на более подходящую, например, "Земля круглая"или "Земля в форме эллипсоида в зависимости от требуемой точности и доступных данных.
В начале мы будем изучать преимущественно предсказательные модели, которые стремятся выявить связь между признаками объекта и целевой переменной. Иногда нам придется работать с моделями данных, например, определять распределение признаков.
Чаще всего мы выбираем предсказательные модели из параметрически семейств, где параметры подбираются на основе данных.

1.8. Модель и алгоритм обучения II

Давайте рассмотрим пример задачи прогнозирования цены квартиры. Мы можем выбрать константную модель (предсказываем одно значение цены для всех квартир), так как цена не зависит от определенных признаков. Форма признакового описания не имеет большого значения в этом случае.

$$y = f(x)$$

Где: - y - целевая переменная (например, цена квартиры),
- x - признаки объекта,
- f - функция, описывающая зависимость между признаками и целевой переменной.
Несмотря на то, что для решения большинства практических задач сегодня достаточно знать только два типа моделей — градиентный бустинг на решающих деревьях и нейронные сети

1.8. Модель и алгоритм обучения III

— мы постараемся также рассказать о других моделях, чтобы углубить ваше понимание предмета и дать возможность не только использовать лучшие практики, но и, при желании, участвовать в разработке новых идей и поиске новых методов — уже в роли исследователя, а не только инженера.

Помимо выбора модели, важен также выбор алгоритма обучения.

Алгоритм обучения — это процедура, которая преобразует обучающую выборку в обученную модель. Например, для константной модели мы использовали алгоритм обучения, основанный на поиске нуля градиента. Градиентные методы широко используются для обучения многих моделей и представляют собой богатый класс оптимизационных методов, из которого иногда сложно выбрать оптимальный.

Представим, что нам принадлежит большая сеть ресторанов, и мы хотим открыть в неком городе новое заведение. Мы нашли

1.8. Модель и алгоритм обучения IV

несколько точек в городе, где есть возможность приобрести помещение и организовать там ресторан. Нам важно, чтобы через определенное время он стал приносить прибыль — точнее, хочется открыть его в той точке, в которой прибыль окажется максимальной. Поставим задачу: для каждого возможного размещения ресторана предсказать прибыль, которую он принесет в течение первого года.

Объектом мы будем называть то, для чего хотим сделать предсказание. В нашем случае это конкретная точка размещения ресторана. Обозначать объект мы будем маленькой буквой x, а если их несколько, то будем добавлять нижние индексы. Множество всех возможных точек размещения называется пространством объектов и обозначается через X. Величина, которую мы хотим определять (т.е. прибыль ресторана), называется ответом или целевой переменной,

1.8. Модель и алгоритм обучения V

а множество ее значений — пространством ответов Y. В нашем случае пространство ответов является множеством вещественных чисел: $Y = R$. Отдельные ответы будем обозначать маленькой буквой y.

Мы не являемся специалистами в экономике, поэтому не можем сделать такие прогнозы на основе своих экспертных знаний. У нас есть лишь примеры — поскольку мы владеем целой сетью ресторанов, то имеем данные по достаточно большому числу ранее открытых ресторанов и по их прибыли в течение первого года. Каждый такой пример называется обучающим, а вся их совокупность — обучающей выборкой, которая обозначается как $X = (x_1, y_1), ..., (x_l, y_l)$, где $x_1, ..., x_l$ — обучающие объекты, а l — их количество. Особенность обучающих объектов состоит в том, что для них известны ответы $y_1, ..., y_l$.

1.8. Модель и алгоритм обучения VI

Отметим, что объекты — это некие абстрактные сущности (точки размещения ресторанов), которыми компьютеры не умеют оперировать напрямую. Для дальнейшего анализа нам понадобится описать объекты с помощью некоторого набора характеристик, которые называются признаками (или факторами). Вектор всех признаков объекта x называется признаковым описанием этого объекта. Далее мы будем отождествлять объект и его признаковое описание. Признаки могут быть очень разными: бинарными, вещественными, категориальными (принимают значения из неупорядоченного множеств ординальными (принимают значения из упорядоченного множества), множествозначными (set-valued, значения являются подмножествами некоторого универсального множества). Признаки могут иметь сложную внутреннюю структуру: так, в качестве признака для конкретного человека в задаче предсказания его годового дохода

1.8. Модель и алгоритм обучения VII

может служить фотография. Разумеется, фотографию можно представить и как некоторое количество бинарных или вещественных признаков, каждый из которых кодирует соответствующий пиксель изображения. Однако, работа с изображением как с одной сложной структурой позволяет вычислять по нему различные фильтры, накладывать требование инвариантности ответа к сдвигам и т.д.
На работе со сложными данными специализируется активно развиваю
сейчас глубинное обучение (deep learning). В нашей задаче полезными могут оказаться признаки, связанные с демографией (средний возраст жителей ближайших кварталов, динамика изменения количества жителей) или недвижимостью (например, средняя стоимость квадратного метра в окрестности, количество школ, магазинов, заправок, торговых центров, банков поблизости). Разработк

1.8. Модель и алгоритм обучения VIII

признаков (feature engineering) для любой задачи является одним из самых сложных и самых важных этапов анализа данных. Давайте вернемся к задаче прогнозирования прибыли ресторана. Предположим, что у нас есть обучающая выборка и некоторое количество признаков. Мы получаем матрицу "объекты-признаки" $X \in \mathbb{R}^{l \times d}$ (где l— количество объектов, d — количество признаков), где каждая строка содержит описание одного обучающего объекта. Таким образом, строки в этой матрице представляют собой объекты, а столбцы — признаки. Стоит отметить, что здесь возникает небольшая терминологическая путаница: буквой X мы обозначаем как обучающую выборку (с объектами и ответами), так и матрицу "объекты-признаки"(содержащую только объекты). Однако из контекста всегда будет понятно, о чем идет речь.

1.8. Модель и алгоритм обучения IX

Наша задача заключается в построении функции a : $\mathbb{X} \to \mathbb{Y}$, которая будет предсказывать ответ для любого объекта. Такая функция называется алгоритмом или моделью. Очевидно, что не каждый алгоритм подойдет — например, бесполезен будет алгоритм $a(x) = 0$, который предсказывает нулевую прибыль для любого ресторана независимо от его признаков. Для формализации соответствия алгоритма нашим ожиданиям необходимо ввести функционал качества, измеряющий работу алгоритма. Если функционал следует минимизировать, его логично назвать функционалом ошибки.

Один из популярных функционалов в задаче регрессии — среднеквадратичная ошибка (mean squared error, MSE):

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

1.8. Модель и алгоритм обучения X

Где: - MSE - среднеквадратичная ошибка, - n - количество наблюдений, - y_i - фактическое значение, - $\hat{y}_i$ - предсказанное значение, - $\sum$ - сумма по всем наблюдениям от i=1 до n.

Чем меньше значение этого функционала дает алгоритм, тем лучше он предсказывает целевую переменную. Среднеквадратичная ошибка (MSE) является очень удобной метрикой благодаря своей дифференцируемости и простоте. Однако, как мы увидим дальше в курсе, у нее есть и ряд недостатков, которые могут повлиять на качество модели и требуют учета других аспектов при оценке эффективности алгоритма.

Давайте рассмотрим пример задачи бинарной классификации точек на плоскости, для которой мы выберем линейную модель (См. рисунок 3.):

1.8. Модель и алгоритм обучения XI

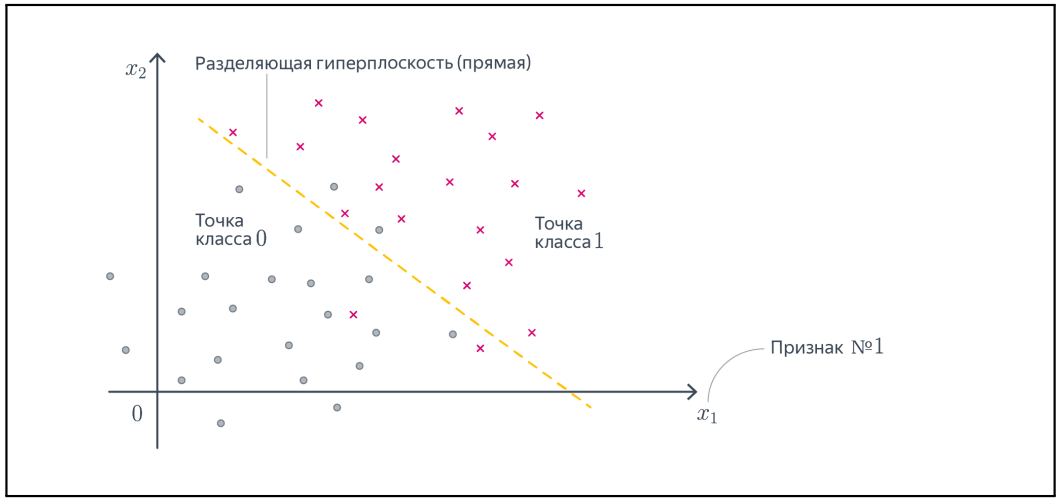

Рис.: 3. Бинарная классификация

1.8. Модель и алгоритм обучения XII

Теперь нам нужно подобрать оптимальную разделяющую прямую по обучающей выборке. В данном случае формула для разделяющей прямой имеет следующий вид:

$$w^T x + b = 0$$

Числа w и b являются настраиваемыми (обучаемыми) параметрами модели, именно их будет восстанавливать алгоритм обучения по выборке. Однако возникает проблема: метрика accuracy не является дифференцируемой. Поэтому необходимо выбрать другую дифференцируемую функцию $L(w, b)$, минимизация которой будет соответствовать оптимизации вероятности. Эта функция называется функцией потерь или лоссом (от англ. loss). Подробнее о видах лосс-функций для бинарной линейной классификации можно узнать в разделе о линейных моделях.

1.8. Модель и алгоритм обучения XIII

В качестве алгоритма обучения мы можем использовать градиентный спуск:

$$w = w - \alpha \nabla L(w, b)$$

$$b = b - \alpha \nabla L(w, b)$$

где α — шаг оптимизации, коэффициент, который влияет на скорость и устойчивость алгоритма. Важно отметить, что разный выбор коэффициента α приводит к различным алгоритмам обучения и результатам: слишком маленький шаг может замедлить достижение оптимума, а слишком большой может вызвать осцилляции вокруг оптимума. Таким образом, важен не только выбор модели, но и выбор алгоритма обучения.

1.8. Модель и алгоритм обучения XIV

Число α является гиперпараметром алгоритма, который задается до начала обучения, но его также можно подбирать на основе данных.

1.9. Выбор модели, переобучение. I

Может показаться, что мы вас обманули, когда пугали сложностями: очевидно, что для любой задачи машинного обучения можно построить идеальную модель, надо всего лишь запомнить всю обучающую выборку с ответами. Такая модель может достичь идеального качества по любой метрике, но радости от нее довольно мало, ведь мы хотим, чтобы она выявила какие-то закономерности в данных и помогла нам с ответами там, где мы их не знаем. Важно понимать, какая у построенной модели обобщающая способност то есть насколько она способна выучить общие закономерности, присущие не только обучающей выборке, и давать адекватные предсказания на новых данных. Для того чтобы предохранить себя от конфуза, поступают обычно так: делят выборку с данными на две части: обучающую выборку и тестовую выборку (train и

1.9. Выбор модели, переобучение. II

test). Обучающую выборку используют для собственно обучения модели, а метрики считают на тестовой.
Такой подход позволяет отделить модели, которые просто удачно подстроились к обучающим данным, от моделей, в которых произошла генерализация (generalization), то есть от таких, которые на самом деле кое-что поняли о том, как устроены данные, и могут выдавать полезные предсказания для объектов, которых не видели.

1. 10. После обучения. I

Когда подобраны все обучаемые параметры и гиперпараметры модели, работа специалиста по машинному обучению не заканчивается. Во-первых, модель чаще всего создается для работы в продакшене. Чтобы модель там успешно функционировала, необходимо эффективно её закодировать, обеспечить параллельную работу и интеграцию с используемыми фреймворками. Процесс выкатки в продакшен называется деплоем или деплойментом (от deploy). После деплоя можно оценить онлайн-метрики. Также рекомендуется провести А/Б-тестирование, сравнив новую модель с предыдущей на случайно выбранных подмножествах пользователей или сессий. Более подробную информацию об А/Б-тестировании можно найти в соответствующей главе. Если новая модель не работает оптимально, должна быть возможность вернуться к предыдущей версии.

1. 10. После обучения. II

После деплоя модели важно продолжать её дообучение или переобучени при поступлении новых данных, а также следить за качеством. Мы уже обсуждали data drift, но также стоит учитывать concept drift - изменение зависимостей между признаками и целевой переменной. Например, если вы создаете музыкальные рекомендации, вам нужно учитывать как появление новых треков, так и изменение вкусов аудитории.

- Machine Learning: хорошая подборка книг для начинающего специалиста. Доступно по: https://habr.com/ru/companies/ru_mts/articles/759668/.
- Учебник по машинному обучению. Доступно по: https://education.yandex.ru/handbook/ml.
- Лекции Евгения Соколова. Доступно по: https://github.com/esokolov/ml-course-hse/tree/master/2023-fall.
- MachineLearning.ru. Доступно по: http://machinelearning.ru/wiki/.

Спасибо за внимание