# Основы машинного обучения

## Применение машинного обучения

Все задачи можно разделить на 2 типа: на те, в которых можно явным образом установить связь, и те, в которых такой возможности нет. В задачам первого типа относятся нахождение количества минут по часам, нахождение силы по массе и ускорению и многие другие. А ко второму типу относятся анализ тональности текста, прогнозирование спроса, исследование кредитного риска, анализ изображений, предсказывание текста. **Машинное обучение** применяется именно ко второму типу задач.

## Области машинного обучения

В машинном обучении выделяют несколько основных подходов:

* **Обучение с учителем:** в этом варианте машинного обучения важным условием применимости является наличие примеров решения задач, подобных данной. Именно на основе этих решений и строится связь. Иными словами, **обучение с учителем занимается восстановлением сложных зависимостей по конечному числу примеров**.
* **Обучение с подкреплением** представляет из себя несколько отличный от обучения с учителем подход, основанный на **непрерывном обучении**, то есть исходная модель выполняет какие-то действия, а в ответ получает **награду** за них, и в зависимости от неё **перестраивает свое поведение**.

## Формализация задачи обучения с учителем

Нам дано пространство всех возможных входных данных , называемых **объектами**. По этим входным данными нам необходимо предсказать значение **целевой переменной** , которая принимает свои значения в пространстве . Предсказание мы делаем на основе **обучающей выборки** размером , которая представляет из себя множество пар объект–ответ . Так как компьютер не умеет обрабатывать объекты любой природы, объекты в машинном обучении представляют из себя –мерный картеж , каждая проекция которого называется **фактором**. Итак, целью машинного обучения является построение **модели** , то есть зависимости целевой переменной от объекта. Если представляет собой полином , то такую модель называют **линейной** с параметрами . Для выбора модели используется **функция потерь**, которая отображает, насколько точно конкретная модель предсказывает ответ. Наиболее частно применимой функцией потерь является **квадратичное отклонение**, то есть . На основе функции потерь строиться **функционал ошибки** , который показывает насколько точно модель предсказывает ответ на всей выборке. Также используется понятие **метрики качества**, которым называется функционал, значение которого больше на точных моделях. Функционал ошибки на основе квадратичного отклонения называется **среднеквадратичной ошибкой (MSE)** . Таким образом, если мы имеем семейство алгоритмов , то **обучением** называется поиск .

## Типы задач

Выделяют несколько основных задач обучения c учителем:

* **Регрессия:** задача, в которой целевая переменная является вещественным числом, то есть .
* **Классификация:** задача, в которой целевая переменная может принимать значения только из ограниченного множества, то есть .
  + **Бинарная классификация:** .
  + **Многоклассовая классификация:** .
  + **Классификация с пересекающимися классами:**.
* **Ранжирование:** задача, в которой необходимо отсортировать набор документов по релевантности к запросу , то есть найти оценку релевантности .

Выделяют несколько основных задач обучения без учителем:

* **Кластеризация:** даны объекты, нужно их сгруппировать.

## Типы признаков

Признаки отличают по тому, как выглядит их множество возможных значений :

* **Бинарный признак:** .
* **Вещественный признак:** .
* **Категориальный признак:**  является неупорядоченным множеством.
* **Порядковый признак:**  является упорядоченным множеством, но без операций.

## Метод k ближайших соседей (kNN)

### Гипотеза компактности

Гипотеза компактности утверждает то, что если **объекты похожи по признакам, то и целевые переменные у них похожи**.

### Этапы алгоритма классификации с помощью kNN

1. Запоминаем выборку .
2. Выбираем метрику .
3. Выбираем число .
4. Получаем новый объект .
5. Сортируем объекты выборки по расстоянию до нового объекта .
6. Выбираем ближайших объектов: .
7. Выбираем наиболее популярный класс среди соседей: .

### Выбор метрики

Метрика — обобщение расстояния на многомерные пространства, которая представляет из себя функцию от двух аргументов такую, что

Рассмотрим основные метрики:

* **Евклидова метрика:** .
* **Манхэттенская метрика:** .
* **Метрика Минковского:**.
* **Считающая метрика:** .

### Оценка модели

* В качестве функции ошибки в задаче классификации обычно берется .
* В качестве функционала ошибки обычно берется .
* Или обычно выбирается метрика качества .
* При анализе доли ошибок важно учитывать соотношение классов.

### Выбор количества соседей (оценка качества модели)

При модель дает наибольшее качество на выборке, но зачастую заметно меньшее на новых данных. Такую ситуацию называют **переобучением** (overfitting) и говорят, что у модели низкая **обобщающая способность**. Параметры модели, которые нельзя подбирать по обучающей выборке называются **гиперпараметрами**. В этом случае оптимальным вариантом будет отделить **отложенную выборку**, то есть на одной части выборки модель будет обучается, а на другой тестироваться. Тут встает вопрос о том, в какой пропорции разбивать выборку. Слишком большая обучающая выборка может быть причиной нерепрезентативной тестовой выборки, а слишком большая тестовая выборка может помешать модели достаточно обучиться. Обычно выборку разбивают в отношении 70/30 или 80/20 в пользу обучающей выборки. Однако возникает следующая проблема: неравномерность данных в выборке, и следовательно, зависимость качества модели от выбора тестовой части. Для решения этой проблемы используют метод **кросс-валидации**: выборку делят на одинаковых частей и каждую из них используют в качестве тестовой, потом показатели метрики качества между моделями усредняют. Такой метод надежнее отложенной выборки, но медленнее. Хорошим вариантом является количество фолдов равное количеству данных, но обычно берут 3, 5 или 10. Важно помнить, что отложенная выборка и кросс-валидация лишь способ сравнить разные параметры модели на этапе подготовки, обучение лучше проводить на всей выборке. Набилее предпочтительным вариантом является следующий: разбиваем выборку на 3 части, обучающую выборку используем для обучения, валидационную для выбора гиперпараметорв, а тестовую для проверки качества модели, этот способ лучше, так как качество модели на валидационной выборке завышено из-за того, что мы выбирали модель с наибольшим качеством. Тут обычно принцип такой: 70/20/10 или 80/10/10.

Заметим также, что при .

### Взвешенный kNN

В стандартный подход классификации kNN можно усовершенствовать, добавив веса соседям, то есть получить формулу , где или и так далее, главное, чтобы убывали. Так же популярным вариантом является **парзеновское окно**: , где есть **ядро**, а есть **ширина окна**. Ядро всегда невозрастает на положительной полуоси. **Гаусовское ядро** есть . Также можно в качестве ядра использовать стандартную гиперболу и ширину окна брать единичной, тогда .

### Регрессионный kNN

1. Запоминаем выборку .
2. Выбираем метрику .
3. Выбираем число .
4. Получаем новый объект .
5. Сортируем объекты выборки по расстоянию до нового объекта .
6. Выбираем ближайших объектов: .
7. Берем среднее арифметическое среди соседей: .

Вместо среднего арифметического можно брать медиану или добавить веса (формула Надарая-Ватсона): .

Функция ошибки (средняя абсолютная ошибка, MAE): . (слабее штрафует за сильные отклонения).

### Резюме

Плюсы kNN

* Если данных много и для любого объекта найдется похожий в выборке, то это лучшая модель
* Очень простое обучение (только запомнить выборку)
* Мало гиперпараметров (, , )

Задачи, где kNN уместна (выполняется гипотеза компактности)

* Классификация изображений
* Классификация текстов на большое количество классов

Минусы kNN

* Часто другие модели оказываются лучше
* Надо хранить в памяти всю обучающую выборку
* Искать соседей довольно долго
* Мало способов настроить модель

## Линейная модель

### Линейная регерссия

Линейная регрессия имеет следующий вид , где есть свободный коэффициент/сдвиг, а есть коэффициенты/веса. Уравнение линейной регрессии также можно записать через скалярное произведение: , где . Можно еще больше упростить запись, взяв для всех объектов, и получив , где и .

### Парная регрессия

Если у нас есть один признак, то линейная модель имеет следующий вид: , где и — параметры сдвига и наклона.

### Кодирование признаков

* **One-hot кодирование категориального признака:** если есть категорий, то создаем новых бинарных признаков на каждую категорию.
* **Кодирование нелинейного признака:** превращаем признак в категориальный (разбиваем ось на интервалы) и применяем one-hot кодирование.
* **Кодирование нелинейного признака:** добавляем новый признак .

### Резюме

Плюсы линейной модели

* Есть простор для фантазии (подготовка признаков)

Задачи, где линейная модель уместна

Минусы линейной модели

* Нет гарантий, что целевая переменная именно так зависит от признаков
* Надо формировать признаки так, чтобы модель подходила