Введение в машинное обучение

[Искусственный интеллект и экспертные системы]

Капырин Николай, старший преподаватель каф. 305

Москва, 2018

Московский Авиационный Институт

Введение

Анализ данных (машинное обучение)

наука, изучающая способы *извлечения закономерностей* из ограниченного количества примеров.

Машинное обучение посвящено строгому изучению методов извлечения закономерностей из данных (см. *математический анализ*)

Анализ данных – название ремесла, направленного на решение прикладных задач (см. *инженер*)

Искусственный интеллект

наука и технология создания интеллектуальных машин, особенно интеллектуальных компьютерных программ; свойство интеллектуальных систем выполнять творческие функции, которые традиционно считаются прерогативой человека.

Алгоритм

набор инструкций, описывающих порядок действий исполнителя для достижения некоторого результата

Экспертная система

компьютерная система, способная частично заменить специалиста-эксперта в разрешении проблемной ситуации

Объекты (машинного обучения) – это абстрактные сущности (точки размещения ресторанов, графические образы, синтаксические конструкции), которыми компьютеры не умеют оперировать напрямую.

$$O$$

Вектор всех признаков объекта x называется признаковым описанием этого объекта.

$$P = \{x_1^O, \ldots, x_n^O\}$$

С точки зрения машинного обучения, объект тождественен своему вектору признаков.

$$P = O$$

Разработка признаков (feature engineering) для любой задачи является одним из самых сложных и самых важных этапов анализа данных.

Признаки могут быть:

- бинарными (да/нет)
- вещественными (1.43, 99.99%)
- категориальными (красный, оранжевый, ...)
- ординальными (значения из неупорядоченного множества)
- множественными (set-valued, подмножеством)

С учителем, без учителя

Задачи обучения с учителем

Задача машинного обучения: Имеется множество объектов (ситуаций) и множество возможных ответов (откликов, реакций).

Существует некоторая зависимость между ответами и объектами, но она неизвестна.

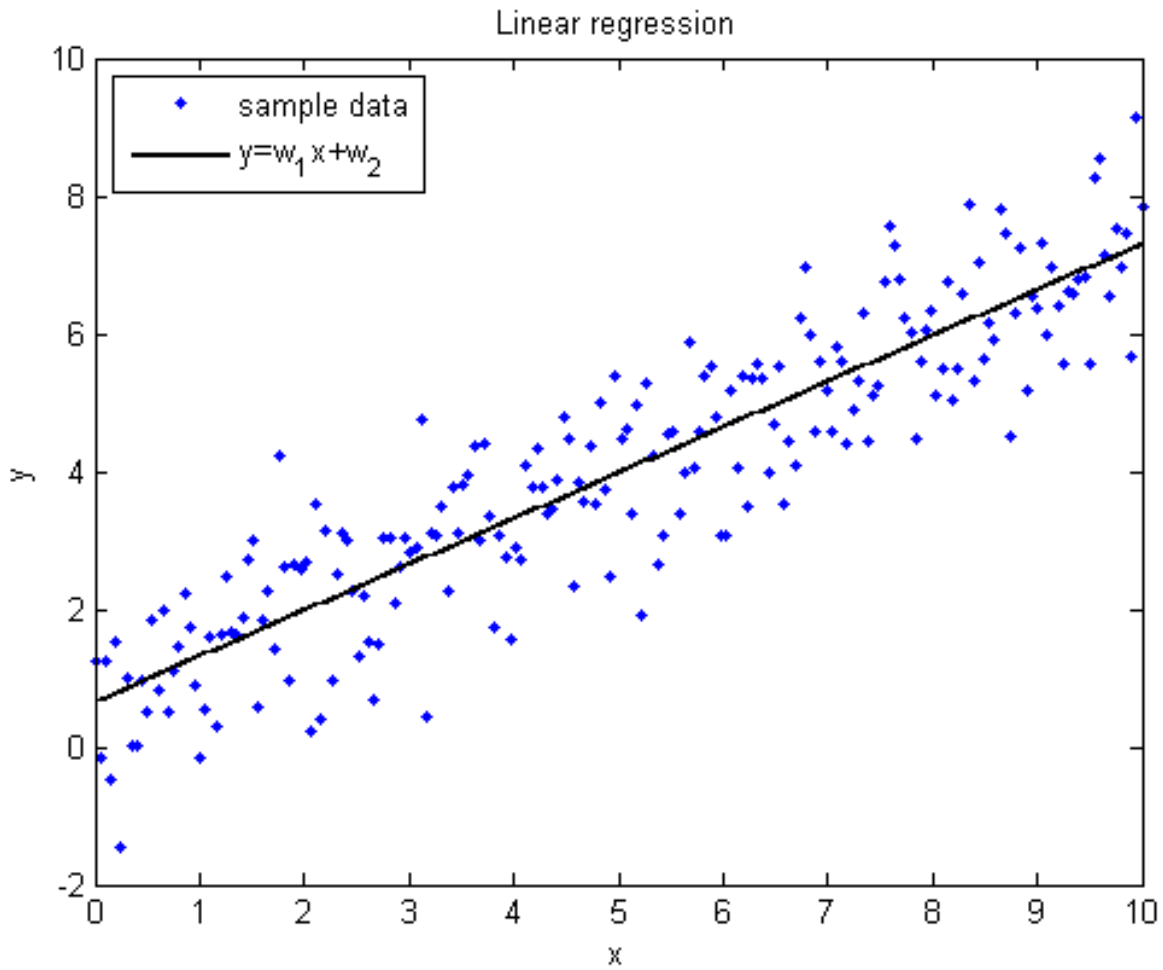
Известна только конечная совокупность прецедентов – пар «объект, ответ», называемая обучающей выборкой.

На основе этих данных требуется восстановить зависимость, построить алгоритм, способный для любого объекта выдать достаточно точный ответ.

Под учителем понимается либо сама обучающая выборка, либо тот, кто указал на заданных объектах правильные ответы. Существует также обучение без учителя, когда на объектах выборки ответы не задаются.

Задачи обучения с учителем

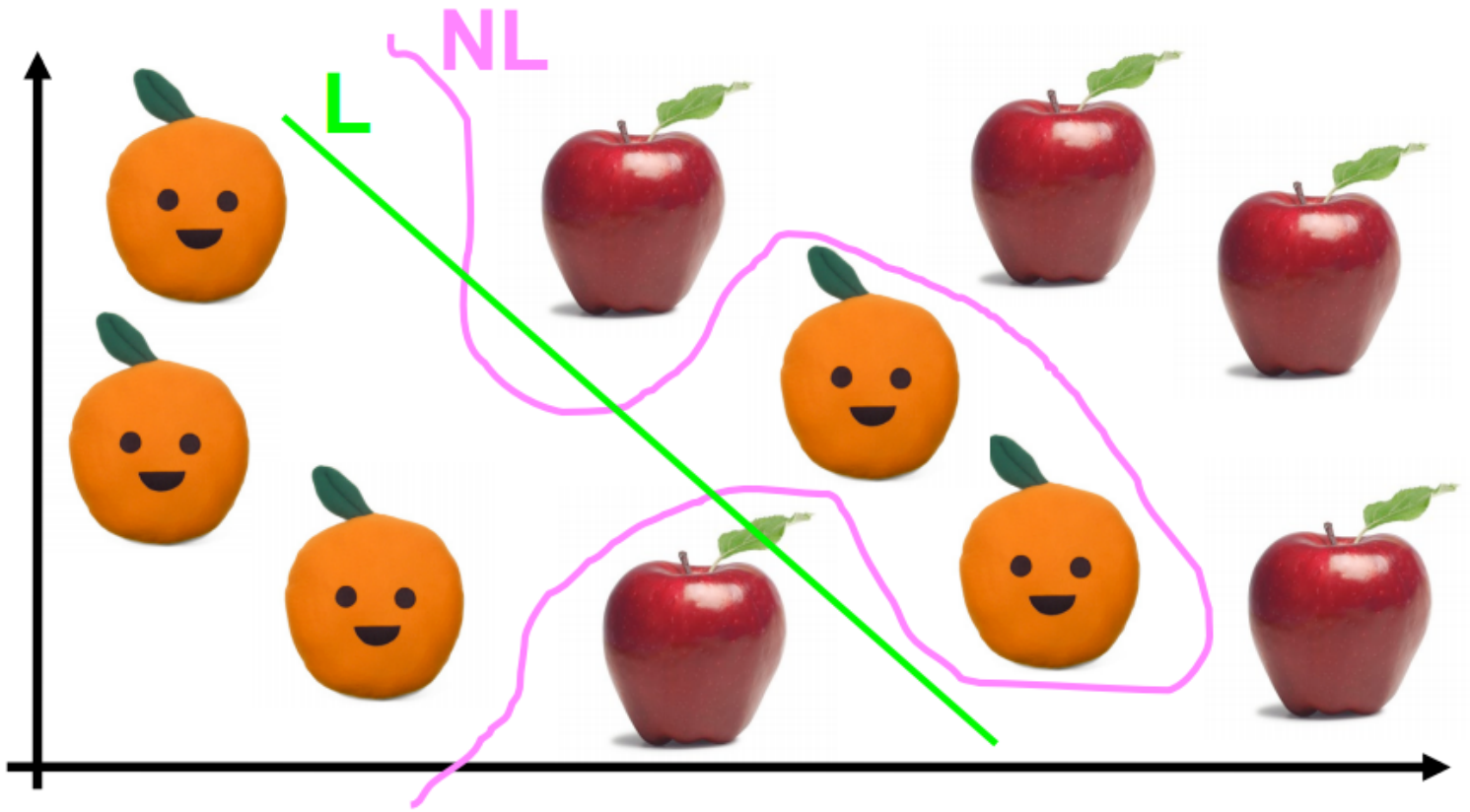
Регрессия – задача с вещественной целевой переменной.



Задачи обучения с учителем

$\mathbb{Y} = \{0, 1\}$ – бинарная классификация.

Например, мы можем предсказывать, кликнет ли пользователь по рекламному объявлению, вернет ли клиент кредит в установленный срок, сдаст ли студент сессию, случится ли определенное заболевание с пациентом (на основе его генома).


NL
L

Задачи обучения с учителем

$\mathbb{Y} = \{1 \ldots M\}$ – многоклассовая (multi-class) классификация.

Примером может служить определение предметной области для научной статьи (математика, биология, психология и т.д.).

Задачи обучения с учителем

$\mathbb{Y} = \{0, 1\}^M$ – многоклассовая классификация с пересекающимися классами (multi-label classification).

Примером может служить задача медицинской диагностики, где для пациента нужно определить набор заболеваний, которыми он страдает.

Задачи обучения с учителем

Ранжирование – задача, в которой требуется восстановить порядок на некотором множестве объектов.

Основным примером является задача ранжирования поисковой выдачи, где для любого запроса нужно отсортировать все возможные документы по релевантности этому запросу.

Частичное обучение

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).

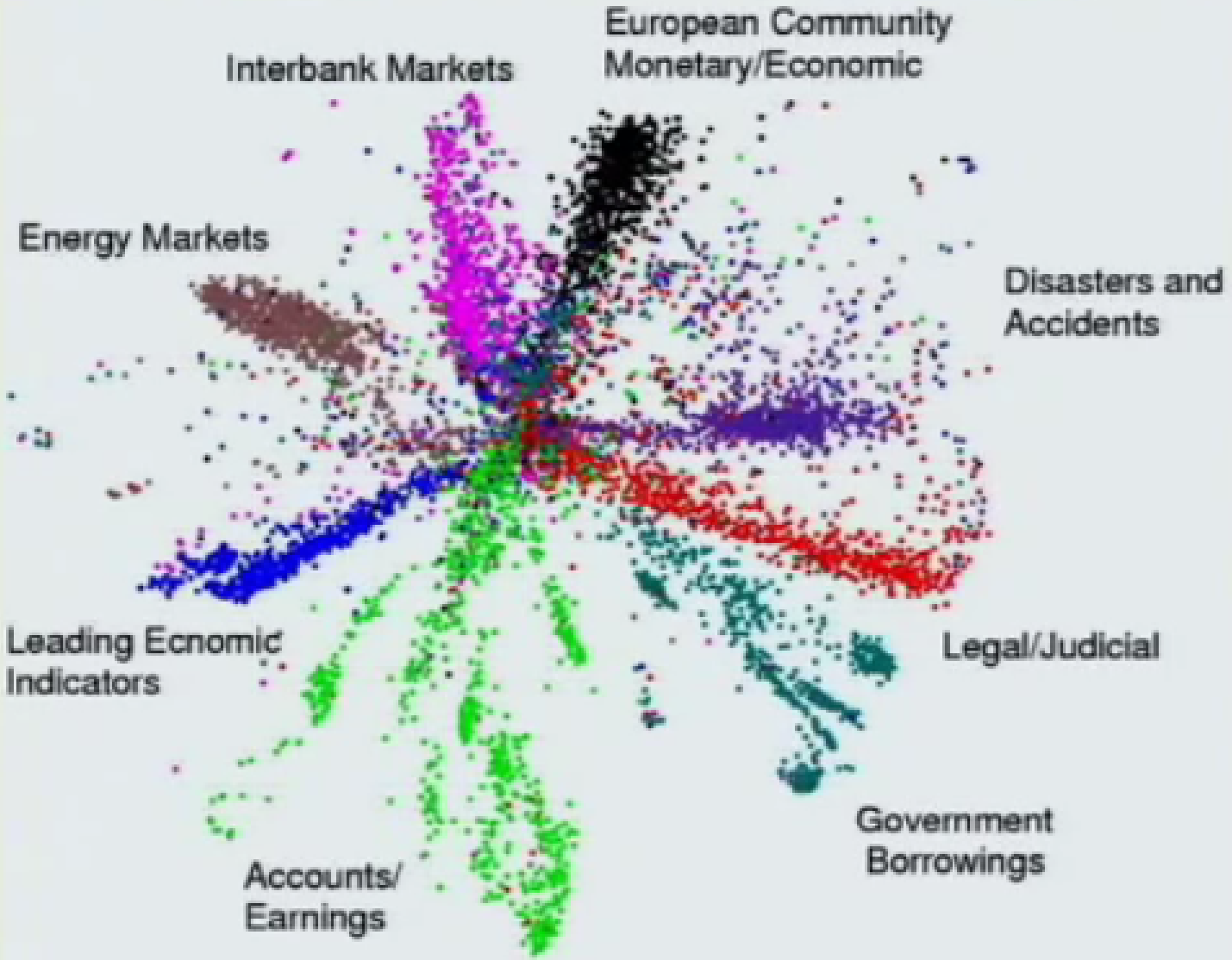
Обучение без учителя

Класс задач, где ответы неизвестны или вообще не существуют, и требуется найти некоторые закономерности в данных лишь на основе признаковых описаний.

Кластеризация – задача разделения объектов на группы, обладающие некоторыми свойствами.

Примером может служить кластеризация документов из электронной библиотеки или кластеризация абонентов мобильного оператора.

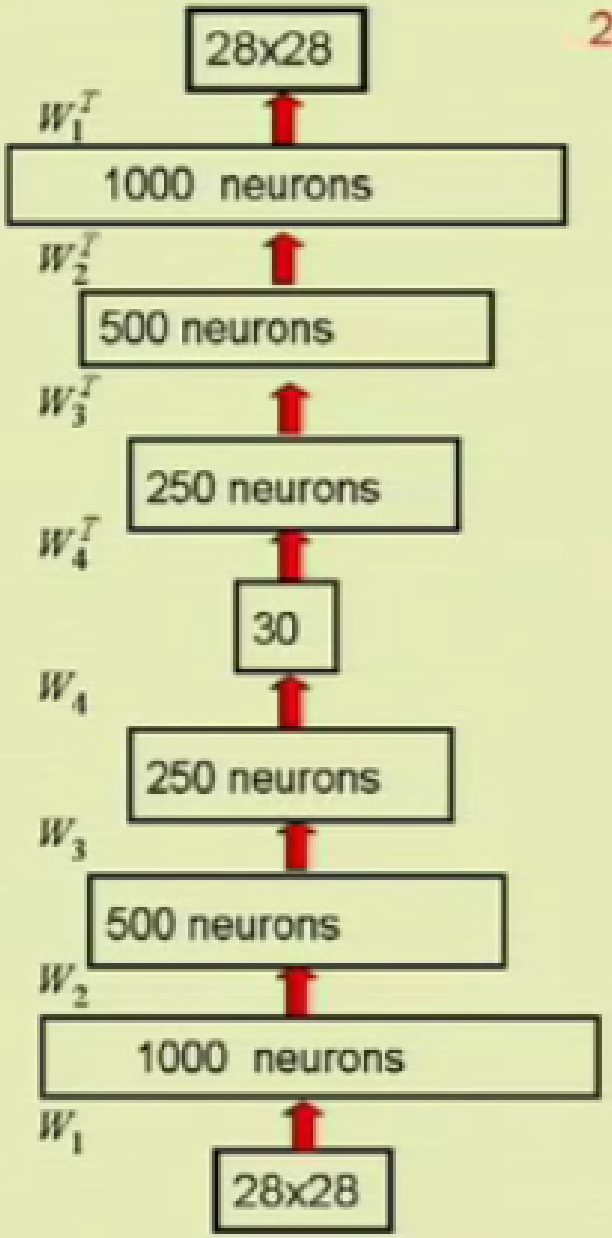
First compress all documents to 2 numbers. 30
Then use different colors for different document categories



Deep Autoencoders

(Ruslan Salakhutdinov)

- They always looked like a really nice way to do non-linear dimensionality reduction:
  - But it is very difficult to optimize deep autoencoders using backpropagation.
- We now have a much better way to optimize them:
  - First train a stack of 4 RBM's
  - Then "unroll" them.
  - Then fine-tune with backprop.



Обучение без учителя

Класс задач, где требуется найти некоторые закономерности в данных лишь на основе признаковых описаний. Правильные ответы в обучающей выборке не даны или вообще не существуют.

Оценивание плотности – задача приближения распределения объектов.

Примером может служить кластеризация документов из электронной библиотеки или кластеризация абонентов мобильного оператора.

Построение моделей обучения

Функционал качества

Нашей задачей является построение функции $a : X \to Y$, которая для любого объекта будет предсказывать ответ.

Такая функция называется алгоритмом или моделью (hypothesis).

Понятно, что нам подойдет далеко не каждый алгоритм – например, вряд ли мы извлечем какую-то выгоду из алгоритма $a(x) = 0$, независимого от признаков.

Функционал качества

Чтобы формализовать соответствие алгоритма нашим ожиданиям, нужно ввести *функционал качества*, измеряющий качество работы алгоритма. Крайне популярным функционалом в задаче регрессии является среднеквадратичная ошибка (mean squared error, MSE):

$$Q(a, X^l) = \frac{1}{l} \sum_{i=1}^{l} (a(x_i) - y_i)^2$$

Удобства: дифференцируемость и простота описания.

Именно функционал качества определяет, какой алгоритм является лучшим, поэтому с плохим функционалом...

Построение алгоритма обучения

Как только функционал качества зафиксирован, можно приступать к построению алгоритма $a(x)$.

Как правило, для этого фиксируют некоторое семейство алгоритмов $\mathcal{A}$, и пытаются выбрать из него алгоритм, наилучший с точки зрения функционала.

В машинном обучении было изобретено большое количество семейств алгоритмов, и самым простым и тщательно изученным является **семейство линейных моделей**, которые дают предсказание, равное линейной комбинации признаков:

$$\mathcal{A} = \{a(x) = w_0 + w_1 x^1 + \cdots + w_d x^d | w_0, w_1, \ldots w_d \in \mathbb{R}\}$$

$$\mathcal{A} = \{a(x) = w_0 + w_1 x^1 + \cdots + w_d x^d | w_0, w_1, \ldots w_d \in \mathbb{R}\}$$

x_i – значение i-го признака у объекта x.

Лучшая из таких моделей будет выбираться минимизацией MSE-функционала:

$$\frac{1}{l} \sum_{i=1}^{l} \left(w_0 + \sum_{j=1}^{d} w_j x_i^j - y \right)^2 \to \min_{w_0, w_1, \ldots, w_d}$$

Процесс поиска оптимального алгоритма называется *обучением*.

Зачастую возникает потребность в предобработке данных до начала построения модели.

Здесь может идти речь о некотором ряде манипуляций:

Нормализация. Некоторые модели хорошо работают только при выполнении определенных требований. Так, для линейных моделей крайне важно, чтобы признаки были нормированными, то есть измерялись в одной шкале. Примером способа нормировки данных является вычитание среднего и деление на дисперсию каждого столбца в матрице «объекты-признаки».

Фильтрация. Бывает, что в выборку попадают выбросы – объекты, которые не являются корректными примерами из-за неправильно посчитанных признаков, ошибки сбора данных. Их наличие может сильно испортить модель.

Переобучение. Некоторые признаки могут оказаться шумовыми, то есть не имеющими никакого отношения к целевой переменной и к решаемой задаче. Примером, скорее всего, может служить признак «фаза луны в день первого экзамена» в задаче предсказания успешности прохождения сессии.

Переобучение (overfitting)

Простейшая предобработка данных может радикально улучшить качество итоговой модели.

Пример. Допустим, что мы выбрали очень богатое семейство алгоритмов, состоящее из всех возможных функций: $\mathcal{A} = a : X \to Y$.

В этом семействе всегда будет алгоритм, не допускающий ни одной ошибки на обучающей выборке, который просто запоминает ее:

$$a(x) = \begin{cases} y_i, & x = x_i \\ 0, & x \in X^l \end{cases}$$

Очевидно что для любого нового объекта, алгоритм покажет нулевой прогноз – модель переобучена.

Идея для борьбы с переобучением

В нашем примере переобучение возникло из-за большой сложности семейства – алгоритмом могла оказаться любая функция.

Очевидно, что если бы мы ограничили себя только линейными моделями, то итоговый алгоритм уже не смог бы запомнить всю выборку.

Таким образом, можно бороться с переобучением путём контроля *сложности семейства алгоритмов* – чем меньше у нас данных для обучения, тем более простые семейства следует выбирать.

Отложенная выборка

После того, как модель построена, нам нужно оценить, насколько хорошо она будет работать на новых данных.

Для этого, например, можно в самом начала отложить часть обучающих объектов и не использовать их при построении модели.

Тогда можно будет измерить качество готовой модели на этой отложенной выборке, получив тем самым оценку того, насколько она готова к работе на новых данных.

Существуют и более сложный класс методов, называемый кросс-валидацией...

Решение задачи анализа данных

Проектировщик

Постановка задачи

$\Downarrow$

Выделение признаков

$\Downarrow$

Формирование выборки

$\Downarrow$

Выбор метрики качества

$\Downarrow$

Подготовка/переработка данных

$\Downarrow$

Построение алгоритма обучения

Оценка качества модели

Пользователь

Обучение

Использование