

Введение в машинное обучение

[Искусственный интеллект и экспертные системы]

Капырин Николай, старший преподаватель каф. 305

Москва, 2018

Московский Авиационный Институт

Введение

Введение

С учителем, без учителя

Введение

С учителем, без учителя

Построение моделей обучения

Введение

Анализ данных (машинное обучение)

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

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 (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.

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

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

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

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.

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

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.

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

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.

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

Признаки могут быть:

- бинарными (да/нет)

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%)

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, ...)

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)

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

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

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

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

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.

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**.

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

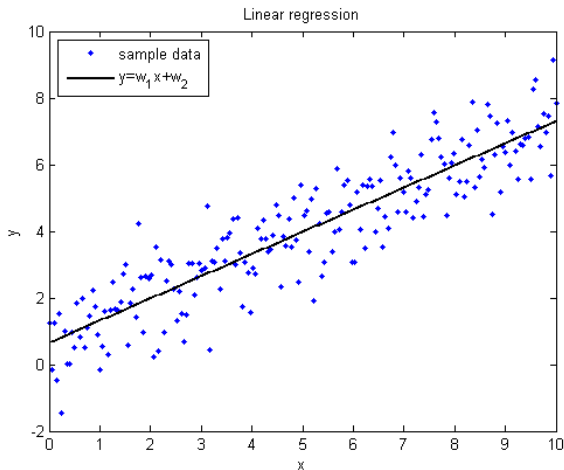
Существует некоторая зависимость между ответами и объектами, но она неизвестна.

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**.

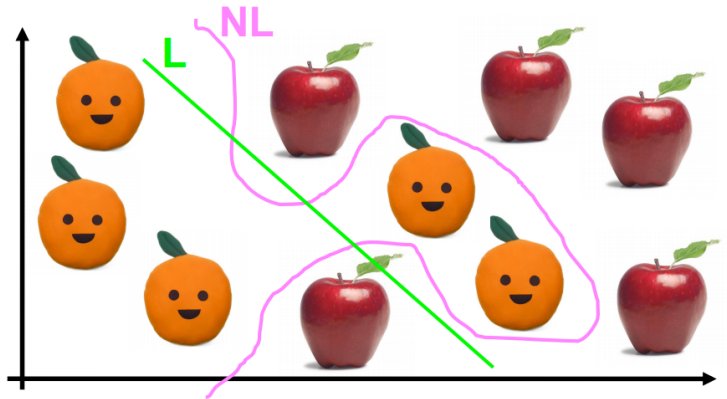
Под учителем понимается либо сама обучающая выборка, либо тот, кто указал на заданных объектах правильные ответы. Существует также обучение без учителя, когда на объектах выборки ответы не задаются.

Регрессия – задача с вещественной целевой переменной.



$\mathbb{Y} = \{0, 1\}$ – 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).



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

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

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

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

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 (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).

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

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

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

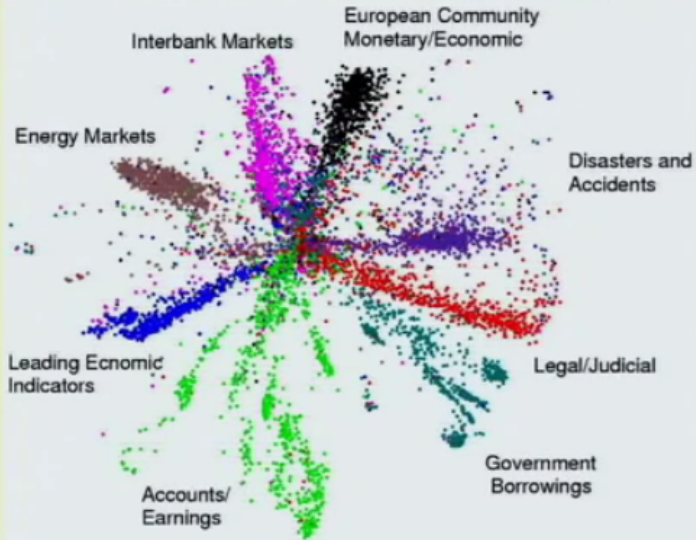
Класс задач, где ответы неизвестны или вообще не существуют, и требуется найти некоторые закономерности в данных лишь на основе признаков описаний.

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.
Then use different colors for different document categories

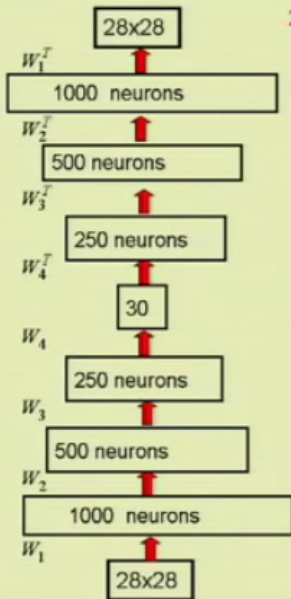
30



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.



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

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

Оценивание плотности – задача приближения распределения объектов.

Класс задач, где требуется найти некоторые закономерности в данных лишь на основе признаков 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

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

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

Такая функция называется алгоритмом или моделью (hypothesis).

Нашей задачей является построение функции $a : X \rightarrow Y$, 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.

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):

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

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...

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

Как только функционал качества зафиксирован, можно приступить 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.

Как только функционал качества зафиксирован, можно приступить 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:

$$\mathcal{A} = \{a(x) = w_0 + w_1x^1 + \dots + w_dx^d \mid w_0, w_1, \dots, w_d \in \mathbb{R}\}$$

$$\mathcal{A} = \{a(x) = w_0 + w_1x^1 + \dots + w_dx^d \mid w_0, w_1, \dots, w_d \in \mathbb{R}\}$$

x_i – значение i -го признака у объекта x .

$$\mathcal{A} = \{a(x) = w_0 + w_1x^1 + \dots + w_dx^d | w_0, w_1, \dots, w_d \in \mathbb{R}\}$$

x_i – значение i -го признака у объекта x .

Лучшая из таких моделей будет выбираться минимизацией MSE-функционала:

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

$$\mathcal{A} = \{a(x) = w_0 + w_1x^1 + \dots + w_dx^d | w_0, w_1, \dots, w_d \in \mathbb{R}\}$$

x_i – значение i -го признака у объекта x .

Лучшая из таких моделей будет выбираться минимизацией MSE-функционала:

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

Процесс поиска оптимального алгоритма называется *обучением*.

Обычно необходимо переработать данные перед началом построения модели.

Обычно необходимо переработать данные перед началом построения модели.

- **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 (вычитание среднего и деление на дисперсию каждого столбца в матрице "объекты/признаки").

Обычно необходимо переработать данные перед началом построения модели.

- **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 (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.

Обычно необходимо переработать данные перед началом построения модели.

- **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 (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 студентом.

Простейшая предобработка данных может радикально улучшить качество итоговой модели.

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

Простейшая предобработка данных может радикально улучшить качество итоговой модели.

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

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

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: $\mathcal{A} = a : X \rightarrow Y$.

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.

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

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.

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.

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

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.

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.

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...

Проектировщик

Пользователь

Постановка задачи

Проектировщик

Пользователь

Постановка задачи



Выделение признаков

Проектировщик

Пользователь

Постановка задачи



Выделение признаков



Формирование выборки

Проектировщик

Пользователь

Постановка задачи



Выделение признаков



Формирование выборки



Выбор метрики качества

Проектировщик

Пользователь

Постановка задачи



Выделение признаков



Формирование выборки



Выбор метрики качества



Подготовка/переработка данных

Проектировщик

Пользователь

Постановка задачи



Выделение признаков



Формирование выборки



Выбор метрики качества



Подготовка/переработка данных



Построение алгоритма обучения

Проектировщик

Пользователь

Постановка задачи



Выделение признаков



Формирование выборки



Выбор метрики качества



Подготовка/переработка данных



Построение алгоритма обучения

Обучение

Проектировщик

Пользователь

Постановка задачи



Выделение признаков



Формирование выборки



Выбор метрики качества



Подготовка/переработка данных



Построение алгоритма обучения

Обучение

Оценка качества модели

Проектировщик

Пользователь

Постановка задачи



Выделение признаков



Формирование выборки



Выбор метрики качества



Подготовка/переработка данных



Построение алгоритма обучения

Оценка качества модели

Обучение

Использование

Спасибо за внимание!