1. Введение

Data Mining - один из шагов в knowledge discovery in databases.

- Сбор данных
- Выделение признаков
- Применение алгоритмов машинного обучения

Data Analysis

- Exploratory DA - невооруженным алгоритмами взглядом пытаемся понять закономерности в данных
- Confirmatory DA - выдвигаем гипотезы, пытаемся подтвердить
- Perdictory DA
- Визуализация данных

Data Science

- Сбор данных
- Интеграция данных (data integration)
- Хранение данных (data warehousing)
- Анализ данных
- Высокопроизводительные вычисления (high-performance computing)

Machine learning - чисто алгоритмы, которые мы потом хотим где-то применить для решении проблемы.

2. Обучение с учителем

Задача индукционного обучения: найти закономерность по небольшому числу известных нам фактов, чтобы в дальнейшем обобщить ее на все возможные в дальшейшем ситуации.

2.1. **Формулировка задачи.**

X - множество объектов;
Y - множество меток (ответов);
$y : X \to Y$ - неизвестная целевая функция (зависимость).

Про часть объектов что-то знаем: $D = \{(x_i, y_i)\}$ - размеченный набор данных, где $\{x_1, .., x_{|D|}\} \subset X$ - объекты, а $y_i = y(x_i)$ - известные метки (значения целевой функции).

Нужно найти алгоритм a: $X \to Y$ решающую (классифицирующую) функцию, приближающую целевую y на X.

2.2. **Объекты.**

$f_j : X -> D_j$ - признаки объектов.
Типы признаков:

- бинарный - $\{0, 1\}$
- категориальный - D_j конечно (цвет)
- порядковый - D_j конечно и упорядочено (сорт муки)
- численный - $\mathbb{R}$ (длина)

Объект - вектор значений признаков этого объекта $(f_1(x), .., f_n(x))$.

Очень плохо преобразовывать напрямую категориальные в численные, так как численные сравниваются по расстоянию между ними, а это может не совпадать с разницей у категориальных (пример: 1 - blue, 2 - yellow, 3 - orange – расстояния между 1,2 и 2,3 одинаковые, но по факту это не так).

2.3. **Ответы.**

- Классификация
  - $Y = \{-1, +1\}$ - бинарная классификация (любит ли человек бургер)
  - $Y = \{1, .., M\}$ - выбор из M непересекающихся классов (самое любимое блюдо)
  - $Y = \{0, 1\}^M$ - для каждого из M пересекающихся классов выбрать свою метку (гражданином каких стран человек является)
- Ранжирование
  - Y - конечно (частично) упорядоченное множество (ранжирование по предпочтительности)
- Регрессия
  - $Y = \mathbb{R}$ или $Y = \mathbb{R}^m$ (с какой вероятностью человек посетит конкретную страну / все страны мира)

2.4. **Как найти алгоритм a?**
Предсказательная модель - параметрическое семейство отображений $A = \{M(x, \theta)|\theta \in \Theta\}$, где $M : X \times \Theta -> Y$ - некоторая функция (зафиксировали θ, по входящему x получаем некоторое y), а Θ - множество возможных значений параметра θ.

Если будем рассматривать полиномы 1ой степени, то получаем линейную модель, где Θ будет n-мерным вектором $\mathbb{R}^n$, где n - число признаков.

Метод обучения - отображение из мн-ва датасетов в мн-во алгоритмов. Зафиксировали модели (хотим найти лучшую из них), пришел датасет, мы хотим, чтобы метод обучения по датасету выбрал лучшую модель.

Метод обучения:

- Валидационный метод - как ставим задачу
- Модель обучения - что за множество алгоритмов, из которого мы выбираем

2.5. **Насколько хорошо a приближает y?**
Функция потерь - величина ошибки алгоритма a на объекте x.

- классификация - угадали или нет
- регрессия - насколько сильно угадали (расстояние - обычно, квадратичная функция потерь)

Чтобы оценить качество алгоритма, эмпирический риск - среднее по ошибкам по всем объектам.

2.6. **Проблема переобучения.**
Проблема переобучения — начиная с определенного уровня сложности предсказательной модели, чем лучше алгоритм показывает себя на тренировочном наборе данных D, тем хуже он работает на реальных объектах.