

Машинное обучение

Лекция 3. Методы кластеризации

Катя Тузова

Разбор летучки

Что такое прецедент?

Разбор летучки

Задача обучения с учителем.

Множество объектов X

Множество допустимых ответов Y

Прецедент - пара объект-ответ (x_i, y_i)

$x_i \in X$ $y_i \in Y$

Разбор летучки

К какому типу задач относятся:

- Прогнозирования потребительского спроса. У компании есть 1000 продуктов, которые она производит. Требуется предсказать сколько будет продано в следующие полгода.
- Вы владелец фейсбука и пишете алгоритм, который определяет был ли взломан пользователь.
- В задачах медицинской диагностики в роли объектов выступают пациенты. Найти вид заболевания.
- Задача кредитного скоринга (Оценка кредитоспособности клиента, на основании которой принимается решение о выдаче кредита)

Разбор летучки

К какому типу задач относятся:

- Прогнозирования потребительского спроса. (регрессия)
- Взломан ли пользователь. (бинарная классификация)
- Найти вид заболевания. (классификация)
- Задача кредитного скоринга. (классификация)

Разбор летучки

Какие из следующих задач являются задачей обучения без учителя?

- Спам фильтр
- Рубрикация текстов (Группировка статей по темам)
- Оценить есть ли у нового пациента диабет
- Прогнозирование времени следующего землетрясения на определенной территории.
- Разделение людей по психотипу.

Разбор летучки

Какие из следующих задач являются задачей обучения без учителя?

- Спам фильтр
- + Рубрикация текстов (Группировка статей по темам)
- Оценить есть ли у нового пациента диабет
- Прогнозирование времени следующего землетрясения на определенной территории.
- + Разделение людей по психотипу.

Разбор летучки

- Пол
- Средний школьный балл
- Номер школы
- Город школы
- Доля пропущенных лекций
- Оценка по мнению родителей
- Пиво/неделя
- Друзей в ВКонтакте
- Расстояние от дома до универа
- Ряд в аудитории
- Наличие планшета
- Периметр головы

Разбор летучки

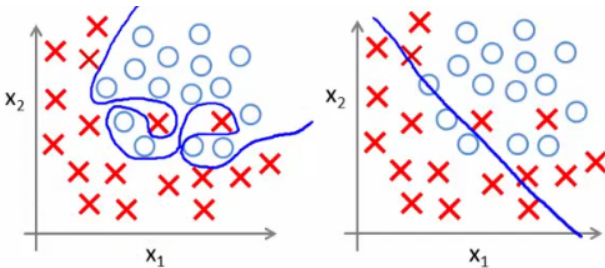
- Пол (бинарный)
- Средний школьный балл (количественный)
- Номер школы (номинальный)
- Город школы (номинальный)
- Доля пропущенных лекций (количественный)
- Оценка по мнению родителей (порядковый)
- Пиво/неделя (количественный)
- Друзей в ВКонтакте (количественный)
- Расстояние от дома до универа (количественный)
- Ряд в аудитории (порядковый)
- Наличие планшета (бинарный)
- Периметр головы (количественный)

Разбор летучки

Приведите пример переобучения и недообучения.

Разбор летучки

Приведите пример переобучения и недообучения.



Разбор летучки 2

Что такое cross-fold validation?

Разбор летучки 2

Что такое cross-fold validation?

Способ разбиения обучающей выборки на два множества L и T .

Разбор летучки 2

Гипотеза компактности:

Схожие объекты, как правило, лежат в одном классе.

Разбор летучки 2

$$a(u, X^l) = \arg \max_{y \in Y} \underbrace{\sum_{i=1}^l [y_u^i = y] w(i, u)}_{\Gamma_y(u)}$$

Смысл параметров w , i , u , $\Gamma_y(u)$

Разбор летучки 2

$$a(u, X^l) = \arg \max_{y \in Y} \underbrace{\sum_{i=1}^l [y_u^i = y] w(i, u)}_{\Gamma_y(u)}$$

$w(i, u)$ - вес i -го соседа u

i - порядковый номер соседа u в упорядоченном множестве

u - объект, для которого проводится классификация

$\Gamma_y(u)$ - оценка близости объекта u к классу y

Разбор летучки 2

Мотивация для использования Парзеновского окна. В чем минусы зависимости веса объекта только от его порядкового номера?

Разбор летучки 2

Мотивация для использования Парзеновского окна. В чем минусы зависимости веса объекта только от его порядкового номера?

Объекты, находящиеся на одинаковом расстоянии будут взяты с разными весами. Далекие объекты могут быть взяты со слишком большим весом.

Разбор летучки 2

Какими свойствами должна обладать функция K , чтобы использовать ее в качестве ядра?

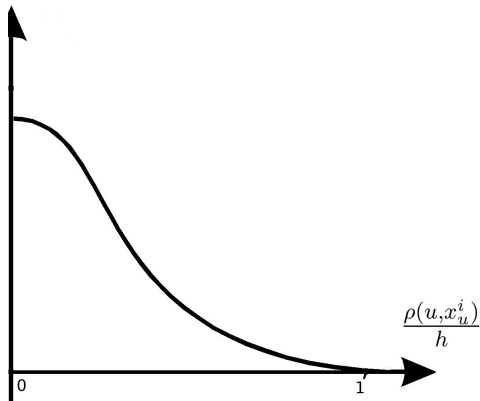
Разбор летучки 2

Какими свойствами должна обладать функция K , чтобы использовать ее в качестве ядра?

Невозрастающая функция, положительная на отрезке $[0, 1]$

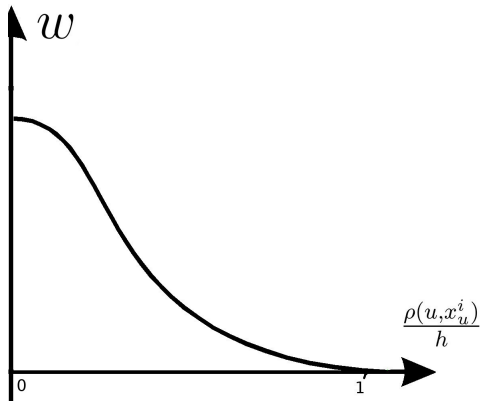
Разбор летучки 2

Что по оси ординат?



Разбор летучки 2

Что по оси ординат?



Разбор летучки 2

Как подбирать функцию расстояния?

Разбор летучки 2

Как подбирать функцию расстояния?

Максимизировать сумму расстояний между объектами разных классов при этом сохраняя сумму расстояний между объектами одного класса небольшой.

$$\max \sum_{x_i, x_j \in D} \rho(x_i, x_j)$$

$$\sum_{x_i, x_j \in S} \rho^2(x_i, x_j) \leq 1$$

Разбор летучки 2

Что такое проклятие размерности?

Разбор летучки 2

Что такое проклятие размерности?

Если используемая метрика $\rho(u, x_u^i)$ основана на суммировании различий по всем признакам, а число признаков очень велико, то все точки выборки могут оказаться практически одинаково далеки друг от друга.

Разбор летучки 2

Жадное добавление признаков – как определить, что признаков уже достаточно?

Разбор летучки 2

Жадное добавление признаков – как определить, что признаков уже достаточно?

Все время минимизируем функционал скользящего контроля (leave-one-out):

$$LOO(k, X^l) = \sum_{i=1}^l [a(x_i; X^l \setminus \{x_i\}, k) \neq y] \rightarrow \min_k$$

Добавляем признаки, пока LOO не увеличивается

Разбор летучки 2

Чем эталонный объект отличается от надежно классифицируемого?

Разбор летучки 2

Чем эталонный объект отличается от надежно классифицируемого?

Эталонные объекты имеют большой положительный отступ, плотно окружены объектами своего класса и являются наиболее типичными его представителями.

Надежно классифицируемые (неинформативные) объекты – изъятие этих объектов из выборки не влияет на качество классификации. Фактически, они не добавляют к эталонам никакой новой информации.

Быстрый поиск ближайшего соседа

k-d дерево

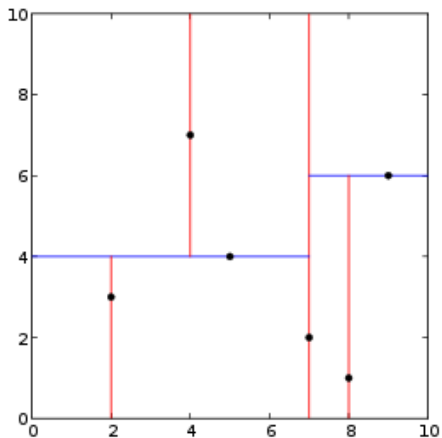
Идея: разложим множество по поторому будем искать в бинарное дерево с простыми условиями и конкретными точками в узлах.

1. По циклу, или рандомно выбираем ось.
2. Ищем точку, разбивающую множество на как можно более равные части.
3. Повторяем 1-2 для каждого из получившихся подмножеств

Сложность построения: $O(n \log n)$

Сложность поиска: в лучшем случае $O(\log n)$, в худшем — $O(n)$

2-d дерево



k-d дерево. Особенности

- + Один из наиболее простых методов
- Работает только при малом количестве параметров
- Затратный алгоритм перестроения

Locality Sensitive Hash

R-соседи – соседи в радиусе R от объекта.

Хэш-функция $h(R, cR, p_1, p_2)$:

$$\|u - v\| \leq R \Rightarrow p(h(u) = h(v)) \geq p_1$$

$$\|u - v\| \geq cR \Rightarrow p(h(u) = h(v)) \leq p_2$$

Постановка задачи кластеризации

Кластеризация – задача разделения объектов одной природы на несколько групп так, чтобы объекты в одной группе обладали одним и тем же свойством.

Кластеризация – это обучение без учителя.

Постановка задачи кластеризации

X – пространство объектов

$X^l = \{x\}_{i=1}^l$ – обучающая выборка

$\rho : X \times X \rightarrow [0, \infty)$ – функция расстояния между объектами

Найти:

Y – множество кластеров

$a : X \rightarrow Y$ – алгоритм кластеризации

Степени свободы в постановке задачи

Степени свободы в постановке задачи

- Критерий качества кластеризации
- Число кластеров неизвестно заранее
- Результат кластеризации существенно зависит от метрики

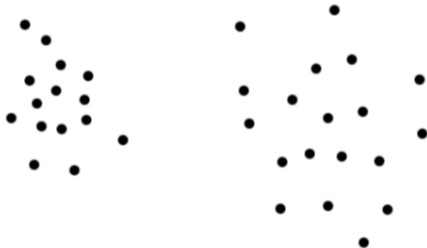
Цели кластеризации

Цели кластеризации

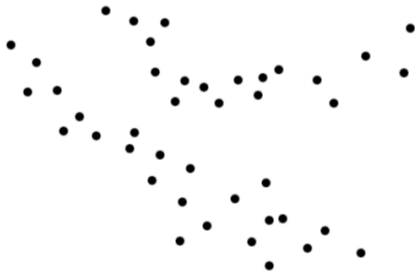
- Сократить объём хранимых данных
- Выделить нетипичные объекты
- Упростить дальнейшую обработку данных
- Построить иерархию множества объектов

Какие бывают кластеры?

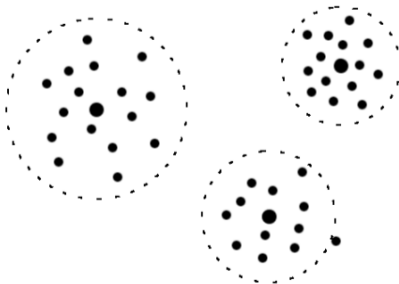
Типы кластерных структур. Сгущения



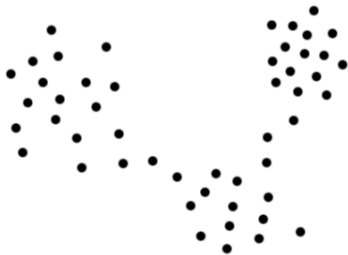
Типы кластерных структур. Ленты



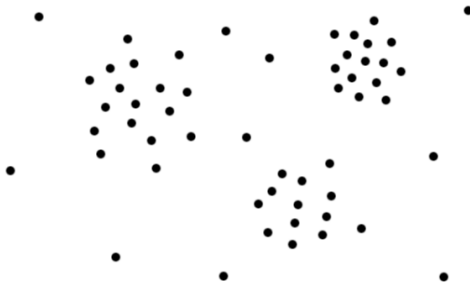
Типы кластерных структур. С центром



Типы кластерных структур. С перемычками



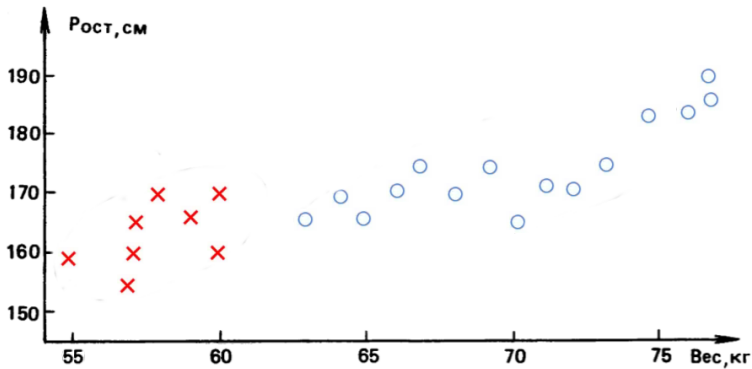
Типы кластерных структур. На фоне



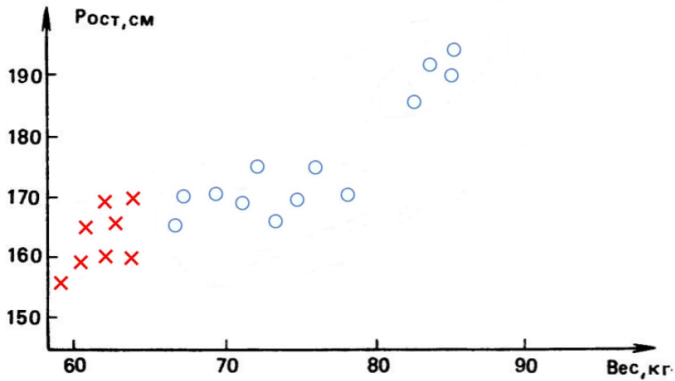
Типы кластерных структур. Перекрывающиеся



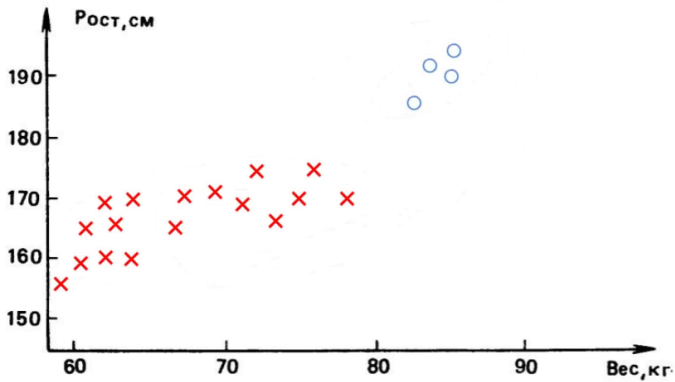
Чувствительность к выбору метрики



Чувствительность к выбору метрики



Чувствительность к выбору метрики



Оценка качества кластеризации

Есть несколько разбиений на кластеры. Как их сравнить?

Оценка качества кластеризации

- Минимизировать среднее внутрикластерное расстояние

$$\frac{\sum_{a(x_i)=a(x_j)} \rho(x_i, x_j)}{\sum_{a(x_i)=a(x_j)} 1} \rightarrow \min$$

- Максимизировать среднее межкластерное расстояние

$$\frac{\sum_{a(x_i) \neq a(x_j)} \rho(x_i, x_j)}{\sum_{a(x_i) \neq a(x_j)} 1} \rightarrow \max$$

Методы кластеризации

- Иерархические
- Графовые
- Статистические

Агломеративный алгоритм Ланса-Уильямса

Идея:

- Считаем каждую точку кластером.
- Затем объединяем ближайшие точки в новый кластер.
- Повторяем.

Алгоритм Ланса-Уильямса

```
 $C_1 = \{\{x_1\}, \{x_2\}, \dots, \{x_l\}\}$   
for  $t = 2, \dots, l$ :  
     $(U, V) = \arg \min_{U \neq V} R(U, V)$   
     $W = U \cup V$   
     $C_t = C_{t-1} \cup \{W\} \setminus \{U, V\}$   
    foreach  $S \in C_t$   
        вычислить  $R(W, S)$ 
```


Алгоритм Ланса-Уильямса

```
 $C_1 = \{\{x_1\}, \{x_2\}, \dots, \{x_l\}\}$   
for  $t = 2, \dots, l$ :  
     $(U, V) = \arg \min_{U \neq V} R(U, V)$   
     $W = U \cup V$   
     $C_t = C_{t-1} \cup \{W\} \setminus \{U, V\}$   
    foreach  $S \in C_t$   
        вычислить  $R(W, S)$ 
```

Алгоритм Ланса-Уильямса

Чего не хватает?

Формула Ланса-Уильямса

Расстояние $R(W, S)$?

$$W = \{U \cup V\}$$

Знаем:

$$R(U, S), R(V, S), R(U, V)$$

Формула Ланса-Уильямса

Расстояние $R(W, S)$?

$$W = \{U \cup V\}$$

Знаем:

$$R(U, S), R(V, S), R(U, V)$$

$$R(U \cup V, S) = \alpha_U R(U, S) + \alpha_V R(V, S) + \\ + \beta R(U, V) + \gamma |R(U, S) - R(V, S)|$$

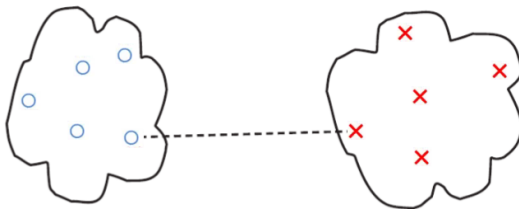
$\alpha_U, \alpha_V, \beta, \gamma$ – числовые параметры

Формула Ланса-Уильямса

Значения параметров $\alpha_U, \alpha_V, \beta, \gamma$?

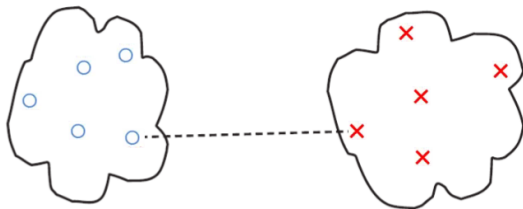
Формула Ланса-Уильямса

Расстояние ближнего соседа:



Формула Ланса-Уильямса

Расстояние ближнего соседа:



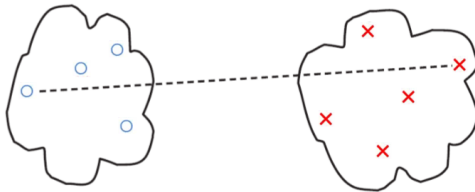
$$\alpha_U = \alpha_V = \frac{1}{2}$$

$$\beta = 0$$

$$\gamma = -\frac{1}{2}$$

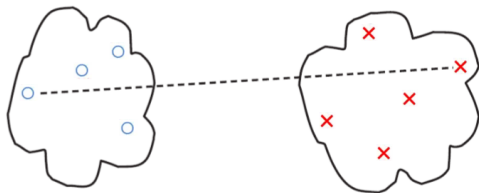
Формула Ланса-Уильямса

Расстояние дальнего соседа:



Формула Ланса-Уильямса

Расстояние дальнего соседа:



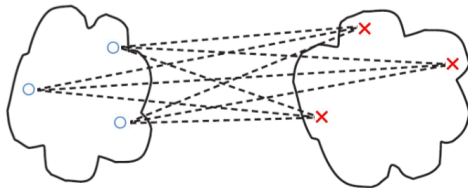
$$\alpha_U = \alpha_V = \frac{1}{2}$$

$$\beta = 0$$

$$\gamma = \frac{1}{2}$$

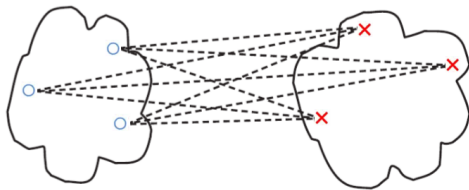
Формула Ланса-Уильямса

Групповое среднее:



Формула Ланса-Уильямса

Групповое среднее:



$$\alpha_U = \frac{|U|}{|W|}$$

$$\alpha_V = \frac{|V|}{|W|}$$

$$\beta = \gamma = 0$$

Графовые алгоритмы

Какие есть две очевидные идеи?

Графовые алгоритмы

Очевидные:

- Выделение связных компонент
- Минимальное покрывающее дерево

Выделение связанных компонент

- Рисуем полный граф с весами, равными расстоянию между объектами
- Выбираем лимит расстояния r и выкидываем все ребра длиннее r
- Компоненты связности полученного графа – наши кластеры

Выделение связанных компонент

Как искать компоненты связности?

Минимальное покрывающее дерево

Минимальное остовное дерево – дерево, содержащее все вершины графа и имеющее минимальный суммарный вес ребер.

Как найти?

Минимальное покрывающее дерево

Как использовать минимальное остовное дерево для разбиения на кластеры?

Минимальное покрывающее дерево

Строим минимальное остовное дерево, а потом выкидываем из него ребра максимального веса.

Сколько ребер выбросим – столько кластеров получим.

Алгоритм FOREL

$$U = X, C = \emptyset$$

while $U \neq \emptyset$:

 выбрать случайную точку x_0

 Повторять пока x_0 не стабилизируется:

$$c = \{x \in X \mid \rho(x, x_0) < R\}$$

$$x_0 = \frac{1}{|c|} \sum_{x \in c} x$$

$$U = U \setminus c, C = C \cup \{c\}$$

Метод k -средних

Идея:

минимизировать меру ошибки

$$E(X, C) = \sum_{i=1}^n \|x_i - \mu_i\|^2$$

μ_i – ближайший к x_i центр кластера

Метод k -средних

Инициализировать центры k кластеров

Пока c_i не перестанет меняться:

$$c_i = \arg \min_{c \in C} \rho(x_i, \mu_c) \quad i = 1, \dots, l$$

$$\mu_c = \frac{\sum_{c_i=c} f_j(x_i)}{\sum_{c_i=c} 1} \quad j = 1, \dots, n, c \in C$$

μ_c – новое положение центров кластеров

c_i – принадлежность x_i к кластеру

$\rho(x_i, \mu_c)$ – расстояние от x_i до центра кластера μ_c

Особенности метода k -средних

- Чувствительность к начальному выбору μ_c
- Необходимость задавать k

Как устранить эти недостатки?

Устранение недостатков

- Несколько случайных кластеризаций
- Постепенное наращивание числа k

На следующей лекции

- Линейные методы классификации
- Метод опорных векторов
- Выбор ядра для метода опорных векторов
- Мультиклассовый классификатор