

# **Машинное обучение без учителя**

## **Вопросы для подготовки к экзамену**

### **1. Общая характеристика задач машинного обучения без учителя. Основные типы задач обучения без учителя.**

Общая характеристика задач “Обучение без учителя” - изучение широкого класса задач обработки данных, в которых известны только описания множества объектов (обучающей выборки), и требуется обнаружить внутренние взаимосвязи, зависимости, закономерности, существующие между объектами.

Основные типы задач:

- Кластеризация – задача разделения множества объектов на группы, обладающие «похожими» свойствами;
- Задача обнаружения аномалий – задача выявления объектов, «непохожих» на остальные объекты;
- Понижение размерности – задача генерации таких новых признаков объектов, что их количество меньше, чем исходных признаков, но при этом качество решения задачи на основе сгенерированных признаков не хуже или не значительно хуже

### **2. Постановка задачи кластеризации объектов. Неоднозначность решения задачи кластеризации. Области применения. Типы кластерных структур.**

Задача кластеризации - задача разбиения заданной выборки объектов (ситуаций) на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из схожих объектов, а объекты разных кластеров существенно отличались. Число кластеров может быть известно или неизвестно.

Неоднозначность решения вытекает из не конкретной постановки задачи: не однозначное определение понятий «близкие объекты», «существенно различны».

- существует много критериев качества кластеризации;
- существует много эвристических (не являющийся гарантированно точным или оптимальным, но на практике часто дает правильные результаты) методов кластеризации;
- результат кластеризации может существенно зависеть от метрики  $\rho$  (функция расстояния между объектами), которая задается субъективно.

Области применения:

- Упрощение дальнейшей обработки данных. Разбиение множества  $X$  на группы и дальнейшая работа с каждой группой в отдельности (классификация, регрессия, прогнозирование).

- Сокращение объема данных для обработки. *Формирование выборки меньшего объема из типичных представителей каждого кластера.*
- Выделение нетипичных объектов. *Определение объектов, которые не подходят ни к одному из кластеров (задачи одноклассовой классификации).*
- Построение иерархии множества объектов. *Распределение документов по условным папкам;*

Типы кластерных структур:

- Самый простой случай - “сгусток точек” с центрами, которые могут быть отделены друг от друга с помощью окружностей (явное разделение кластеров)
- “Сгусток точек” произвольной формы - внутрикласовое расстояние в среднем меньше межкластерного (явное разделение кластеров)
- “Ленточные кластеры” - вытянутая в определенном направлении форма кластера. Внутрикласовое расстояние может быть больше межкластерного.
- “С перемычками” - кластеры, которые имеют пограничные элементы, четко отделенные от основного скопления элементов. Такие объекты трудно отнести к какому-то определенному кластеру.
- Пересекающиеся кластеры
- Кластеры, образованные по определенному неизвестному закону, а не по сходству.

Каждый алгоритм кластеризации имеет свои ограничения и выделяет кластеры лишь некоторых типов.

Все алгоритмы кластеризации работают, выявляя сходство объектов  
→ структуры, не обусловленные сходством, не будут выявлены ни одним алгоритмом.

Понятие «тип кластерной структуры» не имеет строгого определения

Жесткая и мягкая кластеризация:

- Жесткая - объект может относится только к одному кластеру
- Мягкая - объект относится к каждому кластеризацию с некоторым “показателем соответствия” (в сумме = 1, если рассчитывается доля “принадлежности”)

### **3. Функционалы качества кластеризации.**

Возможно два случая:

1. Объекты множества X не являются элементами линейного пространства: задана матрица расстояний между объектами, но нет возможности определить центры кластеров.

2. Объекты множества X являются элементами линейного пространства, и можно рассчитать координаты центров кластеров.

Для первого случая:

Две основные идеи.

- Минимизация среднего внутрикластерного расстояния

$$F_0 = \frac{\sum_{i < j} [y_i = y_j] \rho(x_i, x_j)}{\sum_{i < j} [y_i = y_j]} \rightarrow \min.$$

Суммирование по всем парам объектов из одного кластера

- Максимизация среднего межкластерного расстояния

$$F_1 = \frac{\sum_{i < j} [y_i \neq y_j] \rho(x_i, x_j)}{\sum_{i < j} [y_i \neq y_j]} \rightarrow \max.$$

Суммирование по всем парам объектов из разных кластеров

Для второго случая:

- Минимизация суммы средних внутрикластерных расстояний

$$\Phi_0 = \sum_{y \in Y} \frac{1}{|K_y|} \sum_{i: y_i = y} \rho^2(x_i, \mu_y) \rightarrow \min,$$

где  $K_y = \{x_i \in X \mid y_i = y\}$  – кластер  $y$ ,  
 $\mu_y$  – «центр масс» кластера  $y$ .

Расстояние не между двумя объектами, а от каждого объекта до центра кластера.  
Это более эффективно

- Максимизация суммы межкластерных расстояний

$$\Phi_1 = \sum_{y \in Y} \rho^2(\mu_y, \mu) \rightarrow \max,$$

где  $\mu$  – «центр масс» всей выборки.

Если  $\rho$  – евклидово расстояние, то внутренняя сумма в  $\Phi_0$  имеет физический смысл момента инерции кластера  $K_y$  (кластер – как материальное тело из точек одинаковой массы)

Недостаток рассмотренных функционалов: с помощью этих функционалов нельзя подобрать число кластеров (минимум будет достигаться при отнесении каждого объекта к своему кластеру – среднее внутрикластерное расстояние будет равно нулю)

#### 4. Коэффициент силуэта как метрика качества кластеризации.

**Коэффициент силуэта** - метрика качества, которая позволяет выбрать количество кластеров.

**Коэффициент силуэта объекта** определяется формулой

$$s = \frac{b - a}{\max\{a, b\}},$$

где  $a$  - среднее расстояние от данного объекта до других объектов из этого же кластера,  
 $b$  - среднее расстояние от данного объекта до объектов из ближайшего другого кластера.

Силуэт выборки - среднее значение силуэта по всей выборке. Показывает, насколько среднее расстояние до объектов своего класса отличается от среднего расстояния до объектов других кластеров.

Величина силуэта лежит в [-1, 1]:

- значения, близкие к -1, соответствуют плохим (разрозненным) кластеризациям;
- значения, близкие к нулю, говорят о том, что кластеры пересекаются и накладываются друг на друга;
- значения, близкие к 1, соответствуют "плотным", четко выделенным кластерам.

Для оценки качества кластеризации обычно выполняется визуализация силуэтов объектов каждого кластера; среднее значение (силуэт выборки) обозначается пунктирной линией.

Такие графики можно построить для результатов кластеризации одного и того же набора при разных значениях числа кластеров.

Критерии оценивания:

- разброс значений силуэтов объектов между кластерами;
- значения силуэтов объектов в каждом кластере по отношению к среднему по выборке (больше/меньше)

## **5. Метрики качества кластеризации при известной (частичной) разметке выборки.**

Предположим:

- для некоторого количества объектов выборки известна разметка (метки классов объектов);
- количество размеченных объектов недостаточно для обучения классификатора (обучение с учителем).

Возможный подход: использовать разметку для оценки качества кластеризации с помощью метрики “accuracy”.

Другой подход: использовать метрики, основанные на энтропии.

Энтропия - степень беспорядка в данных (ну типа), степень хаоса и тьмы, ага.

Понятие пришло из физики (термодинамики) и там звучит как “степень неопределенности системы”. Если посмотреть на визуализацию, то можно предположить следующее - чем более объекты разных классов пересекаются друг с другом (накладываются), тем большая у нас энтропия - ну типа хаос, объекты тут и там, их не разделить и хз че происходит.

Чем больше разброс данных, тем больше энтропия.

**Энтропия класса** может быть вычислена по объектам с известными метками:

$$P(c) = \frac{n_c}{n} \quad H(C) = - \sum_{c=1}^{|C|} \frac{n_c}{n} \log_2 \left( \frac{n_c}{n} \right).$$

$n_c$  – количество объектов в классе «С»  
 $n$  – общее число объектов

**Энтропия кластера:**

$$P(k) = \frac{n_k}{n} \quad H(K) = - \sum_{k=1}^{|K|} \frac{n_k}{n} \log_2 \left( \frac{n_k}{n} \right).$$

$n_k$  – количество объектов, отнесенных к кластеру «К»  
 $n$  – общее число объектов

**Энтропия класса при условии кластера:**

$$P(c|k) = \frac{n_{c,k}}{n} \quad H(C|K) = - \sum_{c=1}^{|C|} \sum_{k=1}^{|K|} \frac{n_{c,k}}{n} \log_2 \left( \frac{n_{c,k}}{n_k} \right).$$

$n_{c,k}$  – количество объектов класса «С», которые были отнесены к кластеру «К» (энтропия совместного распределения)  
 $n$  – общее число объектов

## Метрики качества, основанные на энтропии

- Однородность (*homogeneity*)

$$h = 1 - \frac{H(C|K)}{H(C)}.$$

Будет максимальной, если кластеры состоят из объектов только одного класса:  
 $H(C|K) = 0, h = 1.$

- Полнота (*completeness*)

$$c = 1 - \frac{H(K|C)}{H(K)}.$$

Будет максимальной, если объекты каждого класса принадлежат только одному кластеру:  $H(K|C) = 0, c = 1$

- *V*-мера

$$v = 2 \cdot \frac{h \cdot c}{h + c}.$$

Среднее гармоническое однородности и полноты

### RI и ARI

Предполагается, что известны метки классов объектов.

Rand Index (RI):

$$RI = \frac{2(a + b)}{N(N - 1)}$$

*a* - число пар объектов, имеющих одинаковые метки и находящихся в одном кластере,

*b* - число пар объектов, имеющих различные метки и находящихся в разных кластерах,

*N* - число размеченных объектов в выборке.

Значения в диапазоне [0, 1]

Недостаток - не гарантируется значение индекса, близкое к нулю, при случайной маркировки объектов выборки.

Чтобы индекс давал значения, близкие к нулю, для случайных кластеризаций при любом  $N$  и любом числе кластеров, необходимо нормировать его, используя «поправку на случайность»:

$$ARI = \frac{RI - E(RI)}{\max RI - E(RI)},$$

Скорректированный  
индекс Рэнда

где  $E(RI)$  - математическое ожидание  $RI$  для случайно сгенерированной кластеризации.

## 6. Алгоритм метода $k$ -средних, его различные модификации. Особенности метода $k$ -средних.

Метод  $k$ -средних - простейший метод кластеризации.

Идея - кластеризация выполняется путем отнесения каждого объекта к ближайшему центру кластера => все объекты внутри одного кластера ближе к собственному центроиду, чем к центроидам прочих кластеров.

Изначально задается  $k$  - предполагаемое количество кластеров.

Вход - выборка,  $k$

Выход - центроиды полученных кластеров, метки каждого объекта (принадлежность к кластеру)

Преддействия - по определенному правилу, из всей выборки, выбираются  $k$  объектов как первичные центроиды.

Метод итерационный, каждая итерация состоит из двух шагов:

1. Отнесение каждого объекта к ближайшему центроиду (кластеризация)

$$y_i = \arg \min_{y \in Y} \|x_i - \mu_y\|, \quad i = 1, 2, \dots, l;$$

2. пересчет координат центроидов как центров масс только что построенных кластеров

$$\mu_y = \frac{\sum_{i=1}^l [y_i = y] \cdot x_i}{\sum_{i=1}^l [y_i = y]}, \quad y \in Y$$

Итерации выполняются, пока  $y_i$  меняется (если объекты перестают менять свой кластер)

Вариации:

- Болла-Холла -> центры выбираются случайно ( $k > 2$ ) или как наиболее удаленные друг от друга объекты ( $k = 2$ )

- МакКина -> координаты центроидов пересчитываются при каждом переходе объекта в из кластера в кластер

- Mini Batch -> расчет нового центроида выполняется не по всей выборке, а по некоторой случайной подвыборке.

- k-means++ -> первый центроид выбирается случайно из равномерного распределения на выборке. Каждый следующий центроид выбирается случайно из оставшихся объектов так, чтобы вероятность выбрать каждую следующую точку была пропорциональна квадрату расстояния от нее до ближайшего центра.

Особенности:

- Алгоритм крайне чувствителен к выбору начальных приближений центров кластеров. Случайная инициализация центров на шаге 1 может приводить к плохим результатам.

- Кластеризация может оказаться неадекватной, если изначально неверно задано число кластеров. Рекомендация: выполнить кластеризацию при различных значениях  $k$  и выбрать то, при котором достигается резкое улучшение качества по выбранной метрике.

- При очень большом числе признаков рекомендуется предварительно понизить размерность признакового пространства

Кста, k-means очень сильно зависит от формы кластеров и плохо себя показывает на пересекающихся кластеров, потому что тупа не видит пересечение.

## 7. Критерий Inertia. Подбор оптимального числа кластеров с помощью Inertia.

Inertia (инерция) - функционал, который минимизируется алгоритмом k-means.

$$J(C) = \sum_{y \in Y} \sum_{i: y_i = y} \rho^2(x_i, \mu_y) \rightarrow \min_C$$

Тута все просто - минимизируем сумму квадратов расстояний каждого объекта до своего центра. На это влияет как выбор смещение центра, так и отнесение объекта к более близкому кластеру.

Чем больше кластеров (центров), тем меньше inertia.

Подбор оптимального числа кластеров:

$$D(k) = \frac{|J(C_k) - J(C_{k+1})|}{|J(C_{k-1}) - J(C_k)|} \rightarrow \min_k$$

Тута тоже просто - перебирается некое множество числа кластеров ( $k = 2 \dots n$ ), для каждого проводится кластеризация и вычисляется inertia, строится график. Оптимальное количество кластеров - "локоть"/излом, который образуется перед выходом линии на асимптоту (ось x), то есть перед тем, как линия будет практически параллельна оси x => inertia перестанет значительно изменяться.

## 8. EM-алгоритм: базовые предположения, описание алгоритма. Особенности алгоритма.

EM-алгоритм (Expectation Maximization / максимизация ожидания) - относится к семейству статистических алгоритмов.

Статистические алгоритмы основаны на предположении, что кластеры могут быть описаны как некоторое семейство (группа) вероятностных (нормальных) распределений => задача кластеризации сводится к разделению смеси распределений.

Базовые предположения (гипотезы):

1. Объекты выборки появляются случайно и независимо согласно вероятностному распределению, представляющему смесь распределений.
2. Каждый объект выборки описан d-мерным вектором числовых признаков. Каждый кластер описывается d-мерной гауссовой плотностью  $p_y(x)$  с центром “ $\mu$ ” = ( $\mu_1, \mu_2, \dots, \mu_d$ ) и диагональной корреляционной матрицей (матрицей ковариаций)

Вход: выборка X ( $x_1, x_2, \dots, x_l$ ), параметр k (количество кластеров)

Выход: параметры многомерного нормального распределения  
 $\mu_{y1}, \mu_{y2}, \dots, \mu_{yd}, \sigma_{y1}, \sigma_{y2}, \dots, \sigma_{yd}$

Идея: параметры многомерного нормального распределения подбираются с помощью метода максимального правдоподобия.

### Пояснения.

$$p(x) = \sum_{y \in Y} w_y \cdot p_y(x),$$
$$p_y(x) = \varphi(\mu_{y1}, \dots, \mu_{yd}, \sigma_{y1}, \dots, \sigma_{yd}, x),$$

Параметры, подлежащие подбору

В соответствии с методом максимального правдоподобия,

$$\mu, \sigma, w = \arg \max_{\mu, \sigma, w} \sum_{i=1}^l \ln p(x_i) = \arg \max_{\mu, \sigma, w} \sum_{i=1}^l \ln \left( \sum_{y \in Y} w_y \cdot p_y(x_i) \right).$$

## **Пояснения** (продолжение).

Решение полученной задачи весьма проблематично  
(вычислительные проблемы обращения матриц и др.)

Подход:

введение скрытых переменных для упрощения вычислений.

В ЕМ-алгоритме используются скрытые переменные

$$g_{iy} = P(y|x_i) = \frac{w_y \cdot p_y(x_i)}{\sum_{y \in Y} w_y \cdot p_y(x_i)} .$$

**Апостериорные вероятности кластеров для объекта  $x_i$ , вычисленные по формуле Байеса**

### Шаг 1.

Выбор начального приближения для всех кластеров  $y \in Y$ :

$$w_y = \frac{1}{|Y|} ;$$

$\mu_y$  – случайный объект выборки;

$$\sigma_{yj}^2 = \frac{1}{l \cdot |Y|} \sum_{i=1}^l (x_{ij} - \mu_{yj})^2 .$$

### Шаг 2. Повторять

#### 2.1 E-шаг (expectation):

$$g_{iy} = \frac{w_y \cdot p_y(x_i)}{\sum_{y \in Y} w_y \cdot p_y(x_i)} , \quad y \in Y, \quad i = 1, 2, \dots, l.$$

**Фиксация значений  $g_{iy}$**

#### 2.2 M-шаг (maximization):

$$w_y = \frac{1}{l} \sum_{i=1}^l g_{iy} , \quad y \in Y;$$

**Уточнение параметров кластеров при вычисленных  $g_{iy}$**

$$\mu_{yj} = \frac{1}{l \cdot w_{yj}} \sum_{i=1}^l g_{iy} \cdot x_{ij} ;$$

$$\sigma_{yj}^2 = \frac{1}{l \cdot w_{yj}} \sum_{i=1}^l g_{iy} \cdot (x_{ij} - \mu_{yj})^2 , \quad y \in Y, \quad j = 1, 2, \dots, d.$$



E-шаг: на этом шаге на основании параметров модели вычисляются вероятность принадлежности каждой точки данных к кластеру.

M-шаг: обновляет параметры модели в соответствии с кластерным распределением, проведенным на шаге E.

## 2.3 Отнести объекты к кластерам по байесовскому решающему правилу

$$y_i = \arg \max_{y \in Y} g_{iy}, \quad i = 1, 2, \dots, l$$

пока  $y_i$  не перестанут изменяться.

Недостаток:

EM-алгоритм, как и k-means (хотя в несколько меньшей степени), чувствителен к выбору начального приближения.

## **9. Общая характеристика алгоритмов иерархической кластеризации.**

**Алгоритм Ланса-Уильямса. Визуализация результатов работы алгоритма.**

Алгоритмы иерархической кластеризации (алгоритмы таксономии) строят не одно разбиение выборки на непересекающиеся кластеры, а систему вложенных разбиений: кластеры получаются вложенными друг в друга.

- **Агломеративный** (восходящий). Более распространен  
Сначала каждый объект помещается в свой собственный кластер, после чего происходит постепенное объединение объектов во всё более и более крупные кластеры.
- **Дивизивный** (нисходящий).  
От англ. divisive. Сначала все объекты помещаются в один кластер, который затем разбивается на все более мелкие кластеры.

### Алгоритм Ланса-Уильямса (классический вариант)

#### Шаг 1.

Инициализировать множество кластеров  $C_1$ :

$$t = 1, \quad C_t = \{\{x_1\}, \{x_2\}, \dots, \{x_l\}\}. \quad t - \text{номер итерации}$$

#### Шаг 2. Для $t = 2, 3, \dots, l$

2.1 найти в  $C_{t-1}$  два ближайших кластера:

$$(U, V) = \arg \min_{U \neq V} R(U, V),$$

$$R_t = R(U, V);$$

2.2 изъять кластеры  $U$  и  $V$ , добавить кластер  $W = U \cup V$ :

$$C_t = C_{t-1} \cup \{W\} \setminus \{U \cup V\};$$

2.3 для всех  $S \in C_t$

вычислить расстояние  $R(W, S)$ .

В зависимости от требуемого числа кластеров, необходимо обрезать полученное дерево на определенной глубине.

Визуализация - результат обычно представляется в виде дендрограммы (таксономического дерева).

### **10. Агломеративная кластеризация: способы вычисления расстояния между кластерами, свойства расстояний.**

Агломеративная (восходящая) кластеризация - каждый объект представляет свой собственный кластер из одного элемента. На каждой последующей итерации выбираются два наиболее близких кластера (расчет расстояния - выбор определенного метода) и объединяются в один.

Методы расчета расстояния между кластерами (из 2 элементов и более):

1. Расстояние между ближайшими соседями - минимальное расстояние между парами объектов из двух кластеров.
2. Расстояние между дальними соседями - максимальное расстояние между парами объектов из двух кластеров.
3. Среднее расстояние - среднее по расстояниям пар объектов из двух кластеров
4. Расстояние между центрами кластеров - в каждом кластере вычисляется "предположительный центр" (там может и не быть точки), рассчитывается расстояние между полученными точками
5. Расстояние Уорда

Свойства расстояний:

• Монотонность - функция расстояния считается монотонной, если при каждом слиянии расстояние между объединяемыми кластерами не уменьшается. Если кластеризация обладает свойством "монотонности", то дендрограмму можно построить без "пересечений" (расстояние до центра не обладает данным свойством)

• Растяжение - по мере укрупнения кластера расстояния от него до других кластеров увеличиваются. Свойство растяжения считается желательным: способствует более четкому отделению кластеров. НО: при сильном растяжении возможно найти кластеры там, где их не должно быть.  $R$ (между центрами) и  $R$ (уорда) обладают данным свойством.

• Сжатие - по мере укрупнения кластера расстояния от него до других кластеров уменьшаются. В случае сжимающего расстояния естественная кластеризация может исчезнуть. Расстояние  $R$ (ближайшие соседи) является сильно сжимающим.

• Редуктивность - Расстояние  $R$  называется редуктивным, если для любого  $\delta > 0$  и любых  $\delta$ -близких кластеров  $U$  и  $V$  ( $U, V : R(U, V) \leq \delta$ ) объединение  $\delta$ -окрестностей  $U$  и  $V$  содержит в себе  $\delta$ -окрестность кластера  $W = U \cup V$ . Если расстояние обладает свойством редуктивности, то оно обладает и свойством монотонности.

## **11. Общая характеристика графовых методов кластеризации. Кластеризация по компонентам связности. Кластеризация на основе минимального оствновного дерева.**

Ряд алгоритмов кластеризации основан на представлении выборки в виде графа  $G(V, E)$ : множество вершин графа  $V$  - это множество объектов выборки, а веса рёбер  $v_k = xi, xj$  - это парные расстояния между объектами  $\rho_{ij} = \rho xi, xj$ .

Достоинства этих алгоритмов:

- наглядность,
- относительная простота реализации,
- возможность вносить различные усовершенствования, опираясь на простые геометрические соображения.

Основные подходы:

- выделение компонент связности графа путем удаления некоторых ребер;
- построение минимального оствновного дерева графа с последующим разделением на компоненты связности

Связность графа - это свойство, заключающееся в том, что любые две вершины графа могут быть соединены цепью.

Связные компоненты - это подграфы, которые обладают свойством связности; при этом никакие вершины нельзя добавить в эти подграфы, сохранив их связность.

### **Связность графа**

**Маршрутом** в графе называется последовательность, в которой чередуются вершины и рёбра, начинающаяся и кончающаяся вершиной:

$$v_0, e_1, v_1, e_2, v_2, \dots, e_k, v_k,$$

в которой любые два соседних элемента инцидентны, причём однородные элементы (вершины, рёбра) через один смежны или совпадают.

**Если в графе нет кратных ребер, то достаточно указать только последовательность вершин**

Если в маршруте  $v_0 = v_k$ , то маршрут называется **замкнутым**; в противном случае - **открытым**.  
Если все рёбра в маршруте различны, то маршрут называется **цепью**.

В цепи  $v_0, e_1, v_1, e_2, v_2, \dots, e_k, v_k$  вершины  $v_0$  и  $v_k$  называются **концами цепи**.

Говорят, что цепь с концами  $u$  и  $v$  **соединяет вершины  $u$  и  $v$** .

Две вершины в графе называются **связанными**, если существует соединяющая их цепь.

Граф, в котором все вершины связаны, называется **связным**.

Отношение связности вершин является отношением эквивалентности.

Классы эквивалентности по отношению связности называются **компонентами связности** графа.

Связный граф имеет только одну компоненту связности.

Идея алгоритма:

- задаётся параметр  $R$ ;
- в графе удаляются все рёбра  $(x_i, x_j)$ , для которых  $\rho_{ij} > R$ , и соединёнными остаются только наиболее близкие пары объектов.

Нужно подобрать такое  $R \in [\min_{i,j} \rho_{ij}, \max_{i,j} \rho_{ij}]$ , при котором граф «развалится» на несколько связных компонент.

Найденные компоненты связности - и есть кластеры.

Недостатки подхода.

- Ограниченнная применимость.

Алгоритм наиболее подходит для выделения кластеров типа сгущений или лент; наличие разреженного фона или «узких перемычек» между кластерами приводит к неадекватной кластеризации.

- Плохая управляемость числом кластеров.

Непонятно, какое значение  $R$  позволит получить нужное значение числа кластеров  $K$ . Приходится многократно решать задачу при разных  $R$ .

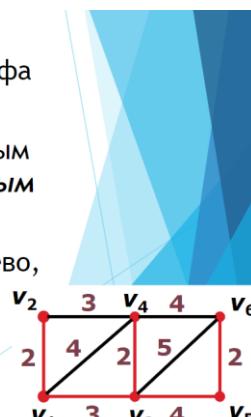
Обойти вторую из указанных проблем (число кластеров) можно с помощью подхода, основанного на построении минимального оставного дерева (например - **алгоритм кратчайшего незамкнутого пути**).

Напоминание.

Граф  $G'(V', E')$  называется **остовным подграфом** графа  $G(V, E)$ , если  $V' = V$  &  $E' \subset E$ .

Остовный подграф, который является деревом (связным графом без циклов), называется **остовом (остовным деревом, каркасом)**.

**Минимальное оставное дерево** - это оставное дерево, имеющее минимальную сумму весов входящих в него рёбер.



## Алгоритм кратчайшего незамкнутого пути (КНП)

### Шаг 1.

Первоначально считать остов пустым.

Найти пару вершин  $(x_i, x_j)$  с наименьшим значением  $\rho_{ij}$  и добавить в остов соединяющее их ребро.

### Шаг 2. Повторять

- 2.1 среди вершин, не включенных в остов, найти ближайшую к некоторой уже включенной вершине;
- 2.2 включить в остов ребро, соединяющее эти вершины пока имеются вершины, не включенные в остов.

### Шаг 3.

Удалить  $K - 1$  ребер с максимальными весами.

## Алгоритм кратчайшего незамкнутого пути (КНП)

После выполнения шага 3 граф распадается на  $K$  компонент связности (кластеров).

Отличие от предыдущего алгоритма выделения компонент связности: число кластеров  $K$  задаётся как входной параметр.

Недостатки:

- как и предыдущий алгоритм, имеет ограниченную применимость (проблемы те же);
- высокая трудоемкость: требуется  $O(l^3)$  операций.

## 12. Методы кластеризации, основанные на плотности: общая характеристика. Метод DBSCAN: описание алгоритма, его достоинства и недостатки, рекомендации к подбору параметров.

Методы, основанные на плотности точек (*density-based*), используют следующую идею.

Все объекты выборки (точки в многомерном пространстве) рассматриваются совместно с их окрестностью.

- Точка называется **основной** или **ядерной** (*core*), если в ее окрестности радиуса  $\varepsilon$  содержитя не менее, чем  $N$  других точек.  **$\varepsilon$  и  $N$  - параметры алгоритма**
- Точка называется **границной** (*border*), если в ее окрестности менее, чем  $N$  других точек, но среди них есть основная.
- В ином случае точка называется **шумовой** (*noise*).

Наиболее известный алгоритм этой группы - **DBSCAN** (*Density-Based Spatial Clustering of Applications with Noise*). Предложен в 1996 г.

Алгоритм рассматривает кластеры как области с высокой плотностью точек, разделенные областями с низкой плотностью.

Следствие такого общего представления:  
кластеры, найденные DBSCAN, могут иметь любую форму  
(в отличие от  $k$ -средних и др.).

Шаг 1.

$i = 1$  (или  $i = 0$ )      **Счетчик кластеров**

Шаг 2.

Пометить все точки выборки, как не просмотренные.

Шаг 3. Для каждой точки выборки из числа не просмотренных

3.1 создать список обхода (все точки из окрестности текущей точки)

3.2 если в списке менее, чем  $N$  других точек

3.2.1 пометить текущую точку как шумовую,

3.2.2 перейти на следующую итерацию (просмотр следующей точки);

иначе

3.3 пометить текущую точку как принадлежащую кластеру  $i$

3.4 для каждой точки из списка обхода

3.4.1 если точка была помечена как шумовая, то

3.4.1.1 пометить ее как принадлежащую кластеру  $i$

3.4.1.2 перейти на следующую итерацию      **Границная точка**

иначе

3.4.2 пометить ее как принадлежащую кластеру  $i$

3.4.3 сформировать список точек из окрестности данной точки

3.4.4 если в списке не менее, чем  $N$  других точек,  
то добавить этот список к списку обхода

3.5  $i = i + 1$

Достоинства алгоритма:

- не требует задания числа кластеров  $K$ ;
- автоматически определяет выбросы;
- позволяет выявить кластеры сложной формы.

Недостатки алгоритма:

- качество работы DBSCAN зависит от способа измерения расстояния;
- плохо работает, если кластеры имеют разную плотность (трудно подобрать оптимальные значения  $\epsilon$  и  $N$ );
- может неправильно определять число кластеров, когда кластеры расположены близко друг к другу.

### ***13. Проблемы анализа многомерных данных. Основные подходы к отбору признаков. Методы одномерного подхода. Проблемы одномерного подхода.***

#### Проблема

При решении задач распознавания объекты выборки могут быть представлены сложными многомерными данными:

- изображениями,
- набором кривых,
- текстом,
- ДНК-микрочипами, и т. д

Перед подачей на вход модели распознавания эти данные должны быть представлены в виде числового вектора.

Особенности таких векторов:

- они имеют (как правило) большую длину,
- содержащиеся в них признаки часто малоинформационны.

Наличие в описании объектов шумовых (не влияющих на значения целевых переменных) признаков создает проблемы в проведении анализа.

Еще один аспект – скорость получения прогноза с помощью модели. Чем больше признаков, тем более сложна модель → тем больше время построения прогноза. Во многих системах время получения прогноза важно или даже критично

Из вышеописанных проблем появляется задача сокращения размерности описания данных (получение относительно небольшого множества информативных признаков).

# Подходы к отбору признаков

## Одномерный подход.

Оценка связи каждого признака с целевой переменной (например, путем определения корреляции).

## Жадные методы отбора.

Перебор различных подмножеств признаков с обучением и оценкой качества модели, использующей только признаки этого подмножества.

## Отбор признаков с помощью моделей.

## Понижение размерности.

Построение новых признаков на основе старых с сохранением максимального количества информации.

Задача “одномерного подхода” - оценить предсказательную силу (информативность) каждого признака. На основе оценки информативности можно отобрать  $k$ -лучших признаков (либо признаки, у которых значение информативности больше некоторого порога)

Основные методы основаны на использовании для отбора признаков:

- корреляции;
- бинарного классификатора;
- метрик теории информации

Минусы одномерного подхода - совместное использование малоинформационных признаков может заметно повысить качество прогнозирования. Одномерный подход не позволяет это прояснить.

## **14. Жадные методы отбора признаков. Алгоритм ADD-DEL. Отбор признаков с помощью обученных моделей.**

Данные методы, по сути, являются надстройками над методами обучения моделей: они перебирают различные подмножества признаков и выбирают то подмножество, которое дает наилучшее качество определённой модели машинного обучения. При этом обучение модели считается «черным ящиком»: вход - информация о том, какие признаки можно использовать при обучении модели; выход - оценка качества обученной модели. Следовательно, получаем задачу минимизации функционала качества модели по подмножествам признаков

Основные методы:

- переборные методы;
- метод жадного добавления;
- алгоритм ADD-DEL

### Алгоритм ADD-DELL.

Жадная стратегия приводит к перебору слишком малого количества вариантов. Один из подходов к усложнению процедуры - алгоритм ADD-DEL (позволяет не только добавлять, но и удалять признаки из оптимального множества).

Идея алгоритма:

- 1) Процедура жадного добавления (множество признаков наращивается, пока удается уменьшать ошибку).
- 2) Процедура жадного удаления признаков из множества, полученного на шаге 1) - аналогична процедуре жадного добавления.
- 3) Повторение шагов 1) и 2), пока уменьшается ошибка.

### Отбор признаков с помощью моделей.

Идея: использовать обученные модели для оценки информативности признаков и их отбора.

Линейные модели.

- Если признаки масштабированы, то веса признаков в обученной модели можно интерпретировать как показатели информативности.
- Для решения задачи отбора признаков можно использовать L1-регуляризацию (чем больше коэффициент регуляризации, тем меньше признаков будет отобрано).

### □ Деревья решений.

Построение разбиения вершины: выбор признака и порога разбиения - с помощью критериев информативности  $H(X)$

- в задаче регрессии - функционал среднеквадратичной ошибки;
- в задаче классификации - критерий Джини или энтропийный.

Пусть в вершине  $m$  производится разбиение по признаку  $j$ .

Оценка важности признака в этой вершине - по величине уменьшения значения критерия информативности:

$$R_{jm} = H(X_m) - \frac{|X_l|}{|X_m|} \cdot H(X_l) - \frac{|X_r|}{|X_m|} \cdot H(X_r).$$

### □ Деревья решений (продолжение).

Пусть  $R_j$  - сумма величин  $R_{jm}$  по всем вершинам дерева, в которых для разбиения использовался признак  $j$ .

Суммарная оценка важности признака  $j$  - по величине  $R_j$  (чем больше  $R_j$ , тем более значим признак).

## □ Композиции алгоритмов.

- В случае композиции деревьев - подход, аналогичный описанному для одного дерева:  
суммарная оценка важности признака  $j$  - по величине  $R_j$ ,  
но значение  $R_j$  определяется суммированием по всем деревьям композиции.

## □ Композиции алгоритмов (продолжение).

- Для случайного леса - подход с использованием out-of-bag:
  - 1) ошибка  $Q_n$  базового дерева  $b_n$  оценивается по out-of-bag выборке;
  - 2) признак  $j$  превращается в шумовой путем перемешивания значений в столбце  $j$ ;
  - 3) оценивается ошибка  $Q'_n$  базового дерева  $b_n$  по выборке, полученной после перемешивания;
  - 4) величина  $Q'_n - Q_n$ , усредненная по всем деревьям случайного леса, характеризует значимость признака (чем больше, тем более значим признак).

## 15. Общая характеристика задачи понижения размерности. Линейный подход к понижению размерности. Метод случайных проекций.

Задача понижения размерности состоит в формировании новых признаков на основе исходных. При этом: количество признаков должно стать меньше, но новые признаки должны сохранять как можно больше информации, присутствующей в исходных (без потери информации не обойтись, но необходимо это минимизировать).

### Линейный подход

Простейший подход к понижению размерности - каждый новый признак является линейной комбинацией исходных признаков.

Значение нового  $j$ -го признака на  $i$ -м объекте линейно выражается через исходные признаки на этом же объекте:

$$z_{ij} = \sum_{k=1}^D w_{kj} \cdot x_{ik}$$

Где:

- $D$  - количество исходных признаков;
- $d$  - количество новых признаков;
- $x_{ij}$  - значение исходного  $j$ -го признака на  $i$ -м объекте выборки;
- $z_{ij}$  - значение нового  $j$ -го признака на  $i$ -м объекте выборки.

- $w_{kj}$  - вес, характеризующий вклад исходного признака  $k$  в новый признак  $j$  на каждом объекте.

### Метод случайных проекций

В методе случайных проекций веса генерируются случайно.

Математическое обоснование метода: лемма Джонсона-Линденштрауса.

#### Лемма Джонсона-Линденштрауса (нестрого):

Если в выборке немного объектов, которые описываются большим числом признаков, то выборку можно спроектировать в пространство меньшей размерности так, что расстояния между объектами практически не изменяются (сохранение топологии в новом признаковом пространстве).

Для изменения расстояний не более, чем на  $\epsilon$ , размерность нового признакового пространства должна удовлетворять условию

$$d > \frac{8\ln l}{\epsilon^2} . \quad (3.2)$$

### **16. Метод главных компонент (PCA): идея метода, постановка задачи оптимизации, применение сингулярного разложения матриц. Описание алгоритма PCA.**

Возможны разные точки зрения на PCA, но пусть будет самая простая:

- нахождение в исходном многомерном пространстве попарно ортогональных направлений – главных компонент – вдоль которых данные имеют наибольший разброс (выборочную дисперсию)

Ортогональность – обобщение понятия «перпендикулярность» для элементов линейного пространства, в котором введена операция скалярного произведения.

Качество проекции на главную компоненту оценивается количеством информации о выборке, сохранившейся после проецирования.

Формализация количества информации – с помощью дисперсии: чем больше дисперсия выборки после проецирования на прямую (главную компоненту), тем больше сохранилось информации о выборке.

Формальная постановка задачи максимизации дисперсии может иметь вид:

$$\sum_{j=1}^d w_j^T \cdot X^T \cdot X \cdot w_j \rightarrow \max_w . \quad (3.7)$$

Для описания формы вектора необходима ковариационная матрица

Ковариационная матрица является обобщением дисперсии на случай многомерных случайных величин – она также описывает форму (разброс) случайной величины, как и дисперсия

Напоминание.

**Корреляционным моментом**  $K_{xy}$  СВ  $X$  и  $Y$  называется величина, равная математическому ожиданию произведения отклонений СВ  $X$  и  $Y$ :

$$K_{xy} = M((X - M(X)) \cdot (Y - M(Y))).$$

Другое название: **ковариация**;  
обозначение -  $\text{cov}(X, Y)$

**Корреляционная матрица** системы СВ  $(X_1, X_2, \dots, X_n)$  (многомерной СВ  $X = (X_1, X_2, \dots, X_n)$ ):

$$\begin{pmatrix} K_{11} & K_{12} & \dots & K_{1n} \\ K_{21} & K_{22} & \dots & K_{2n} \\ \dots & \dots & \dots & \dots \\ K_{n1} & K_{n2} & \dots & K_{nn} \end{pmatrix}.$$

Другое название:  
ковариационная матрица  
(матрица ковариации);  
обозначение -  $\text{cov}(X)$

Эта матрица - обобщение понятия «дисперсия» на случай многомерной СВ.

$K_{ii}$  - дисперсии одномерных СВ  $X_i$

О'кей, мы получили матрицу, описывающую форму нашей случайной величины, из которой мы можем получить ее размеры, а также примерную форму на плоскости. Теперь надо найти такой вектор или вектора, при котором максимизировался бы размер (дисперсия) проекции нашей выборки на него

Формальная постановка задачи:

$$\begin{cases} \sum_{j=1}^d w_j^T \cdot X^T \cdot X \cdot w_j \rightarrow \max_w, \\ W^T \cdot W = E. \end{cases} \quad (3.8)$$

Максимизация  
дисперсии после  
проецирования  
(задача (3.7))

Ограничение,  
обеспечивающее  
единственность решения

Для решения задачи главных компонент можно использовать сингулярное разложение матрицы  $X$ , сформировав матрицу весов  $W$  из собственных векторов (столбцов матрицы  $V$ , соответствующих максимальным сингулярным числам)

**Сингулярным разложением (Singular Value Decomposition, SVD)** матрицы  $A \in \mathbb{R}^{m \times n}$  называется ее представление в виде

$$A = U \cdot D \cdot V^T,$$

где

Сингулярные числа  
матрицы  $A$

$D \in \mathbb{R}^{m \times n}$  – матрица, у которой  $d_{ii} \geq 0$ , а остальные элементы равны нулю;

$U$  и  $V$  – ортогональные матрицы порядка  $m$  и  $n$  соответственно.

Столбцы матриц  $U$  и  $V$  называются, соответственно, **левыми** и **правыми сингулярными векторами** матрицы  $A$ .

Задать  $d$  - число новых признаков.

1. Нормировать значения старых признаков (элементы столбцов матрицы  $X$ ).

2. Найти сингулярное разложение матрицы  $X$ :

$$X = U \cdot D \cdot V^T$$

3. Сформировать матрицу  $W$  из  $d$  столбцов матрицы  $V$ , соответствующих максимальным сингулярным числам.

4. Получить матрицу  $Z$  (новое, сокращенное, признаконое описание выборки):

$$Z = X \cdot W.$$

## 17. Примеры задач поиска аномалий. Природа аномальности. Постановка задачи поиска аномалий. Основные подходы к обнаружению аномалий.

На интуитивном уровне:

аномалии – объекты, которые не описываются общими правилами, закономерностями, справедливыми для представленных данных.

Природа аномальности:

- шумовые данные (образовавшиеся случайным образом – следствие ошибок, сбоев и т. п.);
- редкие или принципиально новые явления, требующие дополнительного изучения (могут выделяться в отдельные кластеры)

Поэтому подходы к обнаружению аномалий разных типов могут существенно отличаться.

Примеры задач поиска аномалий:

- Обнаружение подозрительных банковских операций.
  - Объекты - клиенты банка (в данный момент времени)
  - Признаки объектов - характеристики выполняемых транзакций, характеристики поведения в банке

о Необычное (аномальное) поведение – подозрение на мошенничество (операции с украденной картой и т. п.)

о Возможное решение банка при обнаружении – блокировка карты/выполняемой операции.

● Мониторинг сложной компьютерной системы, включающей большое число взаимосвязанных машин

о Объект - состояния системы в разные моменты времени.

о Признаки объектов - загрузка процессоров, использование памяти на каждой машине, нагрузка на сеть и т.д.

о Необычное (аномальное) поведение - отличается ли текущее состояние системы от тех состояний, про которые известно, что они «нормальные»

о Возможное решение при обнаружении – проведение диагностики системы, при необходимости - ТО

Другие примеры:

- обнаружение вторжений в компьютерную систему (IntrusionDetection);
- обнаружение нестандартных игроков на бирже (инсайдеров);
- медицинская диагностика (Medical Diagnosis);
- сейсмология

Направления в анализе данных, связанные с поиском аномалий:

- Обнаружение выбросов
- Обнаружение “новизны” - объектов, отличающихся по своим свойствам от «нормальных» объектов, но, в отличие от выбросов, отсутствующих в самой выборке

Постановка задачи:

Известна выборка - множество векторов признаков

Требуется для каждого объекта выборки определить:

- 0, если объект “нормальный”
  - 1, если объект не вписывается в определение “нормального” - аномальный
- Формально, это задача бинарной классификации. Однако, использование методов обучения с учителем невозможна, так как “аномальные” объекты могут не присутствовать в выборке (или их будет слишком мало) => классификатор будет выдавать константу (0).

### **Основные подходы к обнаружению аномалий**

Большинство подходов сводятся к построению/созданию некоторой функции “anomaly\_score”, которая для каждого объекта определяет “степень аномальности”.

Решение о присвоении маркера (аномальный “1” или нормальный “0”) выполняется при сравнении “степени аномальности” с некоторым пороговым значением.

Наиболее популярные:

- вероятностный подход (восстановление плотности распределения);
- использование методов классификации;
- изолирующий лес

## **18. Вероятностные методы выявления аномалий: базовая идея, основные подходы (перечислить). Параметрический подход**

### Основная идея

Аномалии - это объекты, которые получены из вероятностного распределения, отличного от распределения обучающей выборки (нормальных объектов). Если найти распределение, из которого получена выборка, то можно оценить вероятность принадлежности нового объекта этому распределению.

Малая вероятность - аномалия. Вероятность используется в качестве функции “anomaly\_score”.

### Основные подходы

- параметрический
- восстановление смесей
- непараметрический

### Параметрический подход

Предполагается - существует параметрическое вероятностное распределение с плотностью  $p(x|\theta)$ , описывающее объекты выборки (нормальные объекты),  $\theta$  - вектор параметров распределения.

Для оценки параметров - метод максимального правдоподобия.

Алгоритм:

- 1) найти распределение выборки  $p(x|\theta)$ ;
- 2) для нового объекта  $x$  определить вероятность порождения этого объекта данным распределением;
- 3) сравнить полученную вероятность с некоторым порогом  $t$

Для построения требуемого распределения нужно решить задачу

$$\sum_{x \in X} \ln p(x_i | \theta) \rightarrow \max_{\theta}$$

В случае, если в выборке присутствуют аномалии, то  $X$  меняется на  $X_{norm}$

Однозначных рекомендаций определить порог  $t$  нет, но есть следующие варианты:

- Использовать априорные соображения (считать все объекты аномалиями, если вероятность меньше 0.01, например)

- Если в выборке присутствуют аномалии, то подобрать такое  $t$ , чтобы присутствующие аномалии корректно считались.

### Недостатки

- на практике сложно выполнить проверку адекватности полученной модели
- сложно обосновать правильность выбранного семейства распределений: малое значение функционала может быть получено как вследствие неудачного моделирования.

### Итог

Данный подход следует использовать только с опорой на априорную информацию (известен тип распределения и прочее)

## **19. Вероятностные методы выявления аномалий: непараметрический подход (формула Парзена-Розенблatta, примеры ядер, обобщение на многомерный случай).**

Предположим: нет оснований считать, что распределение выборки можно отнести к какому-либо семейству параметрических распределений (нет оснований использовать для построения модели то или иное семейство)

Непараметрическое восстановление плотности основано на локальной аппроксимации плотности  $p(x)$  в окрестности объекта  $x$  (фактически, используется только определение плотности распределения)

### Одномерный случай

Для одномерного случая чаще всего используется формула Парзена-Розенблatta:

$$p_h(x) = \frac{1}{l \cdot h} \sum_{i=1}^l K\left(\frac{x - x_i}{h}\right), \quad (4.3)$$

где

$h > 0$  - параметр, *ширина окна*;  
 $K(r)$  - **ядро** - четная неотрицательная функция, удовлетворяющая условию нормировки

$$\int_x K(r) dr = 1, \quad \text{что, в свою очередь,}$$

$$\text{гарантирует } \int_x p_h(x) dx = 1.$$

**Основное свойство плотности  
(должно быть обеспечено)**

В качестве ядра может использоваться, в частности, функция Гаусса (плотность нормального распределения)

В этом случае:

- для каждого объекта выборки строится своя “гауссиана” (всем точкам на оси присваиваются значения плотности)
- для получения итогового распределения “гауссианы” в каждой точке суммируются

Другие ядра:

- Епанечникова
- квартическое
- треугольное
- гауссовское
- прямоугольное

### Влияние ядра

Функция ядра  $K$  практически не влияет на точность восстановления плотности. Минимальное значение функционала ошибки достигается для ядра Епанечникова; другие ядра уступают ему незначительно.

Вид ядра влияет

- на степень гладкости функции  $p_h(x)$ ;
- на эффективность вычислений

Доказано - оценка формулы Парзена-Розенблatta сходится к истинному значению плотности  $p(x)$  для широкого класса ядер при неограниченном увеличении длины выборки  $I$  и одновременном уменьшении ширины окна  $h$ .

Ширина окна  $h$  решающим образом влияет на качество восстановления плотности:

- при слишком узком окне ( $h \rightarrow 0$ ) плотность концентрируется вблизи объектов, и функция  $p_h(x)$  претерпевает резкие скачки;
- при слишком широком окне плотность чрезмерно сглаживается и при  $h \rightarrow \infty$  (бесконечность) вырождается в константу.

### Обобщение на многомерный случай

$$p_h(x) = \frac{1}{l \cdot V(h)} \sum_{i=1}^l K\left(\frac{\rho(x, x_i)}{h}\right), \quad (4.4)$$

где

$\rho(x, x')$  – метрика в пространстве признаков;

$V(h)$  – нормирующий множитель,

$$V(h) = \int_X K\left(\frac{\rho(x, x_i)}{h}\right) dx. \quad \text{Гарантирует выполнение основного свойства плотности}$$

### Замечание

Чтобы определение нормирующего множителя  $V(h)$  было корректно, значение интеграла не должно зависеть от  $x_i$  (однородность пространства  $X$ ).

Это не ограничивает применение подхода, т. к. функцию  $\rho(x, x')$  можно задать так, как удобно. В частности, пространство  $\mathbb{R}^n$  (с евклидовым расстоянием) удовлетворяет требованию однородности.

### Проклятие размерности

Если используемая метрика основана на суммировании различий по всем признакам, а число признаков очень велико, то все точки выборки могут оказаться почти одинаково удаленными друг от друга. Тогда оценки вида (4.4) неадекватны.

Выход - отбор признаков или понижение размерности

## **20. Одноклассовый метод SVM: линейный классификатор SVM, нелинейное обобщение SVM, примеры ядер, применение к задаче поиска аномалий.**

SVM - support vector machine, метод опорных векторов - относится к линейным классификаторам: строит линейную разделяющую поверхность (гиперплоскость). Метод допускает обобщение, позволяющее получать нелинейные модели.

### Линейный классификатор SVM

В случае линейно-разделимой выборки, разделяющая гиперплоскость может быть проведена различными способами.

Идея метода SVM - максимизировать ширину “полосы”, разделяющей объекты разных классов. Разделяющая гиперплоскость - посередине полосы (максимально удалена от обоих классов). После обучения классификатора, все объекты попадающие по одну сторону от построенной гиперплоскости, будут предсказываться как “первый” класс, по другую - “второй класс”.

Объекты, которые лежат на границах “полосы” и между которыми строится “полоса” называются “опорными векторами”.

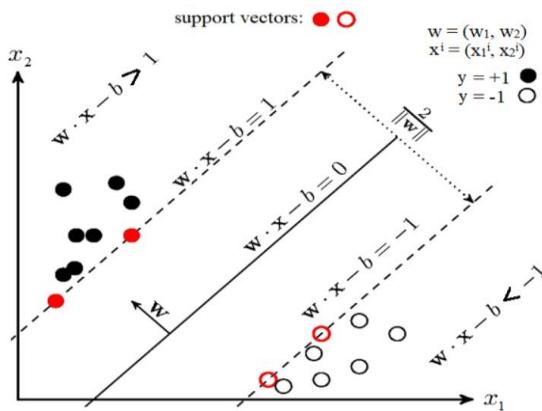
Общий вид преобразования объекта  $x$  в метку класса:  $ax = sign(\langle w, x \rangle - w_0)$ .

Вектор  $w$  - вектор нормали к разделяющей гиперплоскости.

Здесь вводится понятие “Отступа”. Отступ - характеристика уверенности классификатора в его ответе для некоторого объекта. Формула:  $(\langle w, x_i \rangle - w_0) * y_i$ .

Для нормировки задают условие, что минимальное значение  $M = 1$  и достигает только на “опорных” векторах, а для остальных - больше.  $y$  для “первого” класса = 1, для “другого” = -1

В этом случае получается следующая картина:



Ширина разделяющей полосы  $= \frac{2}{\|w\|}$ , а отсюда и задача  $= \frac{2}{\|w\|} \rightarrow \max$

Для линейно разделимой выборки получена задача

условной оптимизации:

$$\begin{cases} \|w\|^2 \rightarrow \min_{w, w_0}, \\ M_i(w, w_0) \geq 1, \quad i = 1, 2, \dots, l. \end{cases}$$

В случае, когда выборка линейно неразделима, решений нет (множество  $M_i(w, w_0) \geq 1$  пусто)

В общем случае задачу надо модифицировать путем ослабления некоторых неравенств с одновременным наложением штрафа за их невыполнение:

$$\begin{cases} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i \rightarrow \min_{w, w_0, \xi}, \\ M_i(w, w_0) \geq 1 - \xi_i, \quad i = 1, 2, \dots, l, \\ \xi_i \geq 0, \quad i = 1, 2, \dots, l. \end{cases} \quad (4.5)$$

Параметр С определяет величину штрафа

Полученный линейный классификатор имеет вид

$$a(x) = \text{sign} \left( \sum_{i=1}^l \lambda_i \cdot y_i \langle x_i, x \rangle - w_0 \right). \quad (4.7)$$

Объекты выборки входят только в виде скалярных произведений

### Нелинейное обобщение SVM

Идея - если можно каким-то образом определять скалярное произведение, не используя признаковое описание объектов, то метод может применяться, даже если признаки явно не сформированы.

И ТУТ НАЧИНАЕТСЯ ПИЗДЕЦ

Предположим: существует некая функция  $\varphi: X \rightarrow H$ , где  $H$  - пространство со скалярным произведением.  $H$  называется "спрямляющим пространством".

Функция  $K: X \times X \rightarrow \mathbb{R}$  называется ядром, если ее можно представить скалярным произведением в некотором пространстве  $H : K(x, x') = \langle \varphi(x), \varphi(x') \rangle$ .

То есть, подставляя вместо скалярного произведения функцию ядра, мы будем настраивать классификатор в произвольном признаковом пространстве. Такая

подмена получила в англоязычной литературе название kernel trick =>  $\langle xi, x \rangle \rightarrow K(xi, x)$  (замена скалярного произведения на какую-то функцию, которая может представить нам на выходе некое пространство со скалярным произведением).

Результат такой замены: если пространство  $H$  имеет достаточно высокую размерность (больше размерности  $X$ ), то выборка в этом пространстве будет линейно разделимой.

Примеры ядер:

- полиномиальное с мономами степени  $k$ :  $K(x, x') = \langle x', x \rangle^k$  (в степени  $k$ )
- полиномиальное (общий случай):  $K(x, x') = (\langle x, x' \rangle + k_0)^k$  (в степени  $k$ )
- сигмоидное:  $K(x, x') = \tanh(\gamma \langle x, x' \rangle + k_0)$
- RBF (радиальная базисная функция):  $K(x, x') = e^{-\gamma \langle x, x' \rangle^2}, \quad \gamma > 0$

### Метод SVM: поиск аномалий

Для реализации поиска аномалий методом SVM задача

(4.5) заменяется на задачу

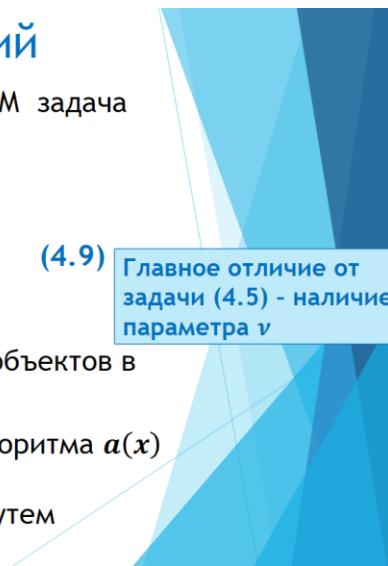
$$\begin{cases} \frac{1}{2} \|w\|^2 + \frac{1}{\nu \cdot l} \sum_{i=1}^l \xi_i - \rho \rightarrow \min_{w, \xi, \rho}, \\ \langle w, x_i \rangle \geq \rho - \xi_i, \quad i = 1, 2, \dots, l, \\ \xi_i \geq 0, \quad i = 1, 2, \dots, l, \end{cases} \quad (4.9)$$

Главное отличие от задачи (4.5) - наличие параметра  $\nu$

где  $\nu$  – верхняя оценка доли аномальных объектов в выборке (гиперпараметр).

Метод решения задачи (4.9) и построения алгоритма  $a(x)$  такой же, как для задачи (4.5).

Затем – построение нелинейной модели путем применения трюка с ядром (kernel trick).



## 21. Метод Изолирующий лес: алгоритм построения деревьев, критерии останова, обнаружение аномалий с помощью изолирующего леса.

Метод “изолирующий лес” реализует одну из вариаций идеи случайного леса.

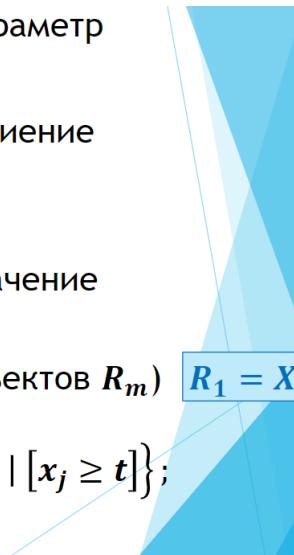
Большинство алгоритмов пытаются моделировать нормальные объекты; аномальными считаются те объекты, которые не соответствуют модели нормального объекта. Идея изолирующего леса – в явной изоляции аномальных объектов (без попыток построения модели нормальных объектов) с помощью бинарных решающих деревьев. В основе алгоритма – тенденция: аномальные объекты легче отделить от других объектов выборки по сравнению с нормальными объектами.

Алгоритм построения дерева:

Нужно построить лес из  $N$  деревьев ( $N$  - гиперпараметр алгоритма).

Построение каждого дерева - рекурсивное разбиение выборки:

- 1) если выполнен критерий останова, то выход:
- 2) выбрать случайный признак  $j$  и случайное значение порога  $t$  (из диапазона значений признака  $j$ );
- 3) разбить вершину  $m$  (содержит множество объектов  $R_m$ ) на две дочерние вершины
$$R_l = \{(x, y) \in R_m \mid [x_j < t]\}, \quad R_r = \{(x, y) \in R_m \mid [x_j \geq t]\};$$
- 4) повторить процедуру для дочерних вершин.



### Критерии останова

Текущая вершина объявляется листовой (не подлежит разбиению) в случаях:

- 1) в вершине только один объект;
- 2) все объекты в вершине имеют одинаковые признаковые описания.

В случае больших выборок имеет смысл также ограничение на глубину дерева (в противном случае случайный процесс разбиения может быть очень долгим).

### Обнаружение аномалий

Идея: при описанном способе построения деревьев аномальные объекты будут попадать в листья на ранних этапах (на небольшой глубине) => можно определять степень аномальности объекта на основании глубины листа, в котором оказался этот объект.

### Обозначения

$n$  - "номер" дерева

$hn(x)$  - оценка деревом  $n$  степени аномальности объекта  $x$

$kn(x)$  - глубина листа в который попадает объект  $x$  в дереве  $n$

$mn(x)$  - число объектов выборки в листе, в который попал объект  $x$  дерева  $n$

Если объект  $x$  - единственный в листе, то разумной оценкой может быть

$$hn(x)=kn(x)$$

Если в листе есть другие объекты (объект  $x$  еще не изолирован при достижении ограничения на глубину дерева), то требуется поправка – оценка глубины поддерева, которое могло бы вырасти из данной вершины с числом объектов  $m$  при отсутствии ограничения на глубину.

Поправка:

$$C(m) = 2H(m-1) - 2 \frac{m-1}{l},$$

где  $H(i)$  –  $i$ -е гармоническое число,  $H(i) \approx \ln i + 0,577$

и

$$h_n(x) = k_n(x) + C(m_n(x)).$$

Итоговая оценка аномальности объекта  $x$  (с учетом всех деревьев):

$$a(x) = 2^{-\frac{\frac{1}{N} \sum_{n=1}^N h_n(x)}{C(l)}}.$$

В показателе – средняя глубина листа, содержащего объект  $x$ , по всем деревьям, соотнесенная со средней глубиной всего дерева по выборке

Для «среднестатистического» объекта выборки показатель

степени будет близок к -1, и  $a(x) \approx \frac{1}{2}$ ;

чем ближе значение  $a(x)$  к 1, тем более аномальным является объект  $x$ .

## 22. Кластеризация текстовых коллекций: краткая характеристика этапов предварительной обработки текста.

Этапы предварительной обработки текста:

- Замена множественных пробелов на одинарные
- Токенизация

Токенизация – разбиение текста на более мелкие фрагменты – токены.

Виды токенов:

Токенами могут быть

- предложения;
  - слова;
  - символы
- Удаление знаков препинания, спецсимволов, “стоп-слов”
  - Перевод в нижний регистр
  - Нормализация: стемминг, лемматизация

Стемминг – это процесс нахождения основы для заданного слова.

Лемматизация – замена слова его смысловой канонической формой

- Разметка

Термы – фрагменты текста, рассматриваемые в анализе как элементы.

Термами могут быть:

- токены (слова, символы);
- $n$ -граммы – последовательности из  $n$  токенов (униграммы, биграммы, триграмм и т.д.);
- предложения

Разметка(tagging) – присвоение термам некоторых характеристик (тегов).

Разметка:

- морфологическая,
- синтаксическая,
- семантическая и др

- Подсчет вхождений

### **23. Векторизация текста на основе частотного подхода: характеристика основных методов.**

Векторизация - преобразование текста к цифровому (векторному) представлению.

Подходы:

- частотный;
- тематическое моделирование;
- дистрибутивная семантика

#### **Метод Bag of Words.**

Основные особенности подхода:

- порядок слов в документе не учитывается; коллекцию документов можно рассматривать как множество пар «документ–слово» ( $d, w$ );
- коллекция документов представляется матрицей  $T = (t)_{d,w}$ : каждая строка соответствуетциальному документу (тексту)  $d$ , а каждый столбец – определенному слову  $w$ ;
- элемент  $t_{d,w}$  равен количеству вхождений слова  $w$  в документ  $d$

#### **TF-IDF.**

Особенность подхода:

элемент  $t_{d,w}$  матрицы  $T$  равен значению функции

$$tf\text{-}idf(w, d, D).$$

$$tf\text{-}idf(t, d, D) = tf(t, d) \cdot idf(t, D),$$

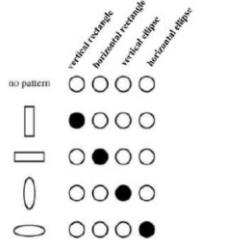
$$tf(t, d) = \frac{n_t}{\sum_k n_k}; \quad idf(t, D) = \ln \frac{|D|}{|\{d_i \in D | t \in d_i\}|}.$$

## 24. Векторизация текста на основе семантического представления слов. Подход Word2Vec.

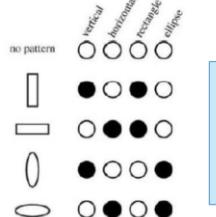
### Векторизация: Word2Vec

Иллюстрация разреженного (sparse) и распределенного (distributed) представления объектов:

sparse representations (OHE)      a distributed representation is dense



a distributed representation is dense



В строке, описывающей объект, может быть более одной единицы

Разреженное (sparse) представление:

появление нового типа объектов (new shape) приведет к увеличению размерности признакового пространства.

Распределенное (distributed) представление:

новый тип объектов можно описать в существующем признаковом пространстве:

$$\text{○} \approx \text{Vertical} + \text{Horizontal} + \text{Ellipse} = \bullet \bullet \circ \bullet$$

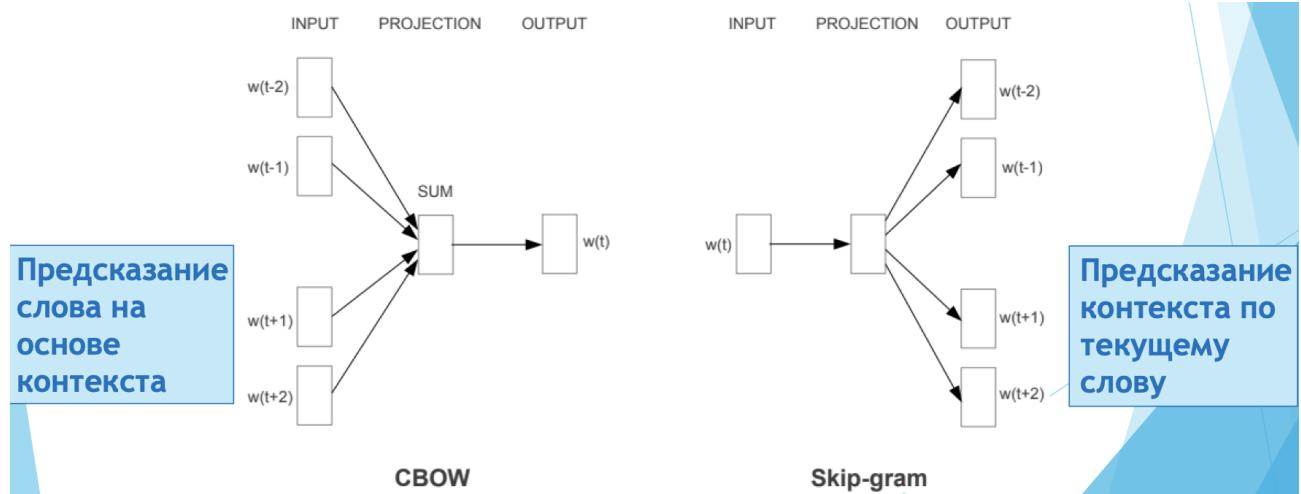
Подход Word2Vec основан на «гипотезе локальности»: слова, которые встречаются в одинаковых окружениях, имеют близкие значения.

Близость понимается как сочетаемость в одном контексте. Например: «завести будильник» – допустимо и привычно; «завести апельсин» - ???

Идея Миколова: конструировать для слов такие вектора, чтобы прогнозируемая моделью вероятность появления наблюдаемых слов в имеющемся контексте была максимальна.

Конструирование векторов слов происходит путем обучения нейронной сети.

Предложено две архитектуры ИНС:



В процессе обучения на последовательности слов

$w_1, w_2, \dots, w_T$  максимизируется средняя логарифмическая вероятность

$$\frac{1}{T} \sum_{i=1}^T \sum_{\substack{-c \leq j \leq c \\ j \neq 0}} \log p(w_{t+j} | w_t) ,$$

где  $c$  - размер обучающего контекста (может быть функцией центрального слова  $w_t$ ).

Вероятность  $p(w_{t+j} | w_t)$  определяется с помощью функции *softmax*:

$$p(w_o | w_I) = \frac{e^{\langle v'_w, v_{wI} \rangle}}{\sum_{w=1}^W e^{\langle v'_w, v_{wI} \rangle}} ,$$

где  $v_w$  и  $v'_w$  – входное и выходное векторные представления слова  $w$ ,

$W$  – число слов в словаре.

В дальнейшем применимость модели была расширена за счет использования векторного представления не только для отдельных слов, но и для словосочетаний и целых фраз.

Идея:

сначала формируется векторное представление большого количества фраз, а затем в процессе обучения фразы рассматриваются как токены.

Свойство модели *Word2Vec*:

полученные векторные представления слов кодируют разнообразные лингвистические закономерности и шаблоны.

Эти закономерности могут быть обнаружены при выполнении арифметических операций над векторами слов.

Примеры (источник):

`vec("Россия") + vec("река")` близко к `vec("Река Волга")`;

`vec("Германия") + vec("столица")` близко к `vec("Берлин")`.

## 25. Основные этапы кластеризации текстовых коллекций. Алгоритм *Affinity Propagation*.

Основные этапы кластеризации текстов

- Предварительная обработка текстов коллекции.

Извлечение признаков - векторизация; понижение

- размерности.
- Применение метода кластеризации.
- Оценка качества кластеризации с помощью метрик.

Основные методы

Популярные алгоритмы:

- Affinity Propagation,
- Agglomerative Clustering,
- K-Means.

Алгоритм *Affinity Propagation* Первая публикация об алгоритме – в 2007 г.

Особенности алгоритма

- получает на вход *матрицу сходства S (matrix of Similarities)* между объектами выборки;

- построен на основе концепции «передачи сообщений» (*passing messages*) между объектами;

предварительное задание числа кластеров не требуется.

При передаче сообщений происходит обновление двух матриц:

- «матрицы ответственности» *R(Responsibility matrix)*,

**Сообщения от объектов  $x_i$  к потенциальным образцам  $x_k$**

- «матрицы доступности» *A(Availability matrix)*.

**Сообщения от потенциальных образцов  $x_k$  остальным объектам  $x_i$**

Для каждого объекта  $x_i$  рассчитывается максимальное значение  $r_{ik} + a_{ik}$ , которое и определяет принадлежность к тому или иному кластеру; индекс  $k$ , на котором достигается максимум, определяет объект-образец – центр кластера (*exemplar*).

Шаг 1.

Инициализация:  $R = 0$ ,  $A = 0$ .

Шаг 2. Повторять

2.1 обновление матрицы *R*:

$$r_{ik} = s_{ik} - \max_{j \neq k} (a_{ij} + s_{ij}), \quad i, k = 1, 2, \dots, l;$$

2.2 обновление матрицы *A*:

$$a_{ik} = \min \left\{ 0, r_{kk} + \sum_{j \neq i, j \neq k} \max \{ 0, r_{jk} \} \right\}, \quad i \neq k;$$

$$a_{kk} = \sum_{j \neq k} \max \{ 0, r_{jk} \}, \quad i, k = 1, 2, \dots, l.$$

пока изменение матрицы *A* не меньше заданного порога.

Шаг 3. Определение центров кластеров:  $c_i = \arg \max_k (a_{ik} + r_{ik}), \quad i = 1, 2, \dots, l$ .

Класс *AffinityPropagation* из модуля *cluster*.

Матрица сходства *S* вычисляется по заданной матрице *X* (задается аргументом метода *fit()* ).



**26. Тематическое моделирование текстовых коллекций: понятие тематической модели; задачи, решаемые с помощью тематического моделирования. Вероятностное тематическое моделирование: базовые предположения, вероятностный процесс порождения текста.**

Тематическое моделирование (topic modeling) - одно из современных приложений машинного обучения к анализу текстов. Активно развивается с конца 90-х годов.

Тематическая модель (topic model) коллекции текстовых документов определяет, к каким темам относится каждый документ и какие термины (слова, словосочетания) образуют каждую тему.

Выполняется мягкая кластеризация (один документ может относиться к нескольким темам)

Задачи, решаемые с помощью тематического моделирования

#### ***Автоматический анализ текста.***

- **Классификация и категоризация документов:** требуется «разложить по папкам» большую коллекцию или поток документов.
- **Автоматическое аннотирование документов:** выделение в документе наиболее важных фраз и составление на их основе краткого обзора документа.
- **Тематическая сегментация документов:** требуется разбить длинный документ на тематически однородные фрагменты и определить тематику каждого фрагмента.

#### ***Систематизация больших объёмов информации.***

- **Семантический (разведочный) поиск информации:** поиск и понимание текстовой информации (например, в самообразовании) – структуризация общих представлений о предметной области.
- **Визуализация тематической структуры коллекции.**
- **Анализ динамики развития тем.**
- **Тематический мониторинг новых поступлений:** возможность автоматического оповещения пользователя о появлении в Интернет или в библиотеке новых документов по заданным темам.
- **Рекомендация документов пользователям** (на основе данных о прошлой активности этого и других пользователей).

#### **Области применения**

- **Поиск научной информации, трендов, фронта исследований.**
- **Подбор экспертов, рецензентов, исполнителей проектов** (пример - автоматизация деятельности экспертных советов: анализ большого количества заявок на инновационные проекты и гранты, подбор экспертов и распределение документов на экспертизу).
- **Агрегирование новостных потоков:** определение тематики каждого документа в приходящем из разных источников новостном потоке, поиск дубликатов и отслеживание каждой темы во времени.
- **Аннотирование и поиск изображений** (элементы изображений можно рассматривать в качестве терминов, а изображения - в качестве документов → возможность построения двух представлений документа: на основе графических элементов и сопровождающего текста → реализация поиска текстов по изображениям, изображений по тексту, аннотирования изображений).

- Анализ видеопоследовательностей, задачи биоинформатики (аннотация генома и др.), анализ дискретизированных биомедицинских сигналов, мониторинг состояния технических систем (применение методов тематического моделирования к анализу последовательностей различной природы).

### Пример применения

Тематизация Википедии: была построена мультиязычная модель (на русском и английском языках).

С помощью модели (без помощи экспертов) было обработано 216175 русско-английских пар статей Википедии и собрано 400 тем (на обоих языках).

Полученные темы оказались легко интерпретируемыми. Более того, модель выявляет двуязычные темы без словарей, даже когда тексты не являются точными переводами.

## **27. Вероятностное тематическое моделирование: вероятностный процесс порождения текста, постановка задачи тематического моделирования, получение частотных оценок вероятностей.**

**Вероятностная тематическая модель** описывает каждую тему дискретным распределением на множестве терминов, а каждый документ - дискретным распределением на множестве тем.

Предполагается, что коллекция документов - это последовательность терминов, выбранных случайно и независимо из смеси таких распределений. Ставится задача восстановления компонент смеси по выборке.

Далее – формализация.

Обозначения:

- $D$  - множество (коллекция) текстовых документов,
- $W$  - множество (словарь) всех употребляемых в них терминов (слов или словосочетаний).

Каждый документ  $d \in D$  представляет собой последовательность  $n_d$  терминов ( $w_1, w_2, \dots, w_{n_d}$ ) из словаря  $W$ .

Термин может повторяться в документе множество раз.

Базовые предположения

Предполагается:

существует конечное множество тем  $T$ , и каждая пара  $d, w$  (употребление термина  $w$  в документе  $d$ ) связана с некоторой темой  $t \in T$ , которая неизвестна.

Коллекция документов рассматривается как множество троек  $d, w, t$ , выбранных случайно и независимо из дискретного распределения  $p(d, w, t)$ , заданного на конечном множестве  $D \times W \times T$ .

Документы  $d \in D$  и термины  $w \in W$  являются наблюдаемыми переменными, тема  $t \in T$  – латентной (скрытой) переменной.

$p(w/t)$  – вероятность (частота) термина  $w$  в теме  $t$ ;

$p(t/d)$  – тематика документа  $d$ . Условное распределение тем

документа (вероятностная смесь тем)

**Гипотеза «мешка слов» (*bag of words*).**

Порядок терминов в документах не важен для выявления тематики: тематику документа можно узнать даже после произвольной перестановки терминов (хотя для человека такой текст теряет смысл).

**Гипотеза «мешка документов».**

Порядок документов в коллекции также не имеет значения.

Гипотеза «мешка слов» позволяет перейти к более компактному представлению документа как подмножества  $d \subset W$ , в котором каждому элементу  $w \in d$  поставлено в соответствие число  $ndw$  вхождений термина  $w$  в документ  $d$ .

**Гипотеза условной независимости.**

Вероятность слова в документах определяется только темой, а не самим документом (распределение  $p(w/t)$  общее для всех документов):

$$p(w|t, d) = p(w|t)$$

Для построения простых моделей используются дополнительные **предположения разреженности**:

- тематика документа состоит из небольшого числа тем;
- тема определяется небольшим числом терминов - лексическим ядром, которое существенно отличает эту тему от других тем.

Если документ относится к большому числу тем (энциклопедия, журнал, сборник статей), то имеет смысл разбить его на части, более однородные по тематике.

Если термин относится к большому числу тем, то, скорее всего, это общеупотребительное слово (стоп-слово), бесполезное для определения тематики.

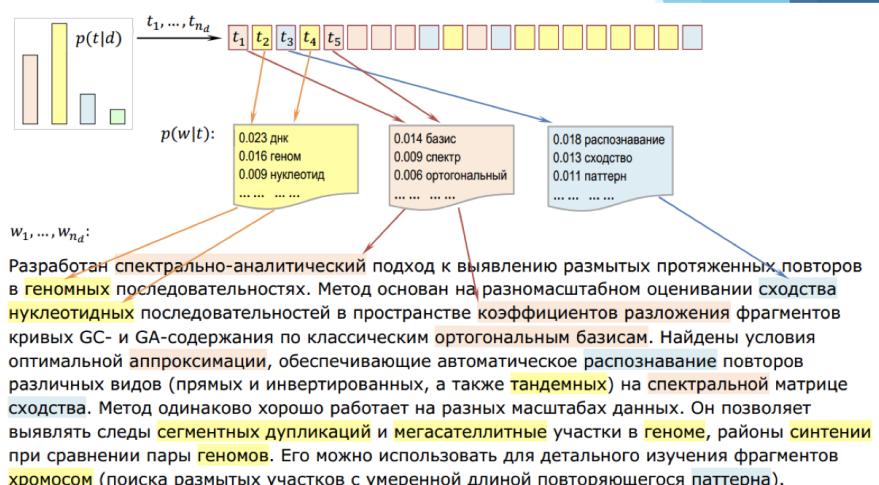
Документ  $d$  - это смесь распределений  $p(w|t)$  с весами  $p(t|d)$ :

$$p(w|d) = \sum_{t \in T} p(w|t) \cdot p(t|d). \quad (5.1)$$

Процесс порождения текста:

для каждой словопозиции

- 1) выбирается тема (в соответствии с  $p(t|d)$ );
- 2) из темы выбирается слово (в соответствии с  $p(w|t)$ ).



При этом (с учетом гипотезы «мешка слов»):

- сгенерированный таким образом текст навряд ли будет осмысленным;
- можно предполагать, что с точностью до произвольной перестановки слов, этот текст может нести в себе какую-то тематику.

Методы тематического моделирования не обеспечивают понимание смысла текста, а только позволяют выполнить кластеризацию документов по темам.

#### Постановка задачи

Построение тематической модели - это задача, обратная к задаче порождения текста.

Дана коллекция документов  $D$ ,  $d \in D$ :

для каждого документа  $d$  известна его длина  $n_d$  и количество  $n_{dw}$  вхождений каждого термина  $w$ .

Требуется восстановить распределения  $p(t|d)$  и  $p(w|t)$ , породившие  $D$ .

### Частотные оценки условных вероятностей

Вероятности, связанные с наблюдаемыми переменными  $w$  и  $d$ , можно оценить по выборке с помощью частот:

$$p^*(d, w) = \frac{n_{dw}}{n}, \quad p^*(d) = \frac{n_d}{n}, \quad p^*(w) = \frac{n_w}{n}, \quad p^*(w|d) = \frac{n_{dw}}{n_d},$$

где

$n_{dw}$  – число вхождений термина  $w$  в документ  $d$ ;

$n_d = \sum_{w \in W} n_{dw}$  – длина документа  $d$  (общее количество вхождений всех терминов в  $d$ );

$n_w = \sum_{d \in D} n_{dw}$  – общее количество вхождений термина  $w$  во все документы коллекции;

$n = \sum_{d \in D} \sum_{w \in d} n_{dw}$  – суммарное количество вхождений всех терминов во все документы коллекции.

Вероятности, связанные со скрытой переменной  $t$ , можно оценить с помощью частот, если рассматривать выборку как множество троек  $(d, w, t)$ :

$$p^*(t) = \frac{n_t}{n}, \quad p^*(w|t) = \frac{n_{wt}}{n_t}, \quad p^*(t|d) = \frac{n_{dt}}{n_d}, \quad p^*(t|d, w) = \frac{n_{dwt}}{n_{dw}},$$

где

$n_{dwt}$  – число троек, в которых термин  $w$  документа  $d$  связан с темой  $t$ ;

$n_{dt} = \sum_{w \in W} n_{dwt}$  – число троек, в которых какой-либо из терминов документа  $d$  связан с темой  $t$ ;

$n_{wt} = \sum_{d \in D} n_{dwt}$  – число троек (по всем документам), в которых термин  $w$  связан с темой  $t$ ;

$n_t = \sum_{d \in D} \sum_{w \in d} n_{dwt}$  – число троек, связанных с темой  $t$ .

**28. Вероятностный латентный семантический анализ (PLSA):  
стохастическое матричное разложение, PLSA-EM – алгоритм.**

# Стохастическое матричное разложение

Если  $|T| \ll |D|$  и  $|T| \ll |W|$ , то равенство (5.1) можно понимать как задачу приближённого представления известной матрицы частот

$$F = (p_{wd}^*)_{W \times D}, \quad p_{wd}^* = p^*(w|d) = \frac{n_{dw}}{n_d},$$

в виде произведения  $F \approx \Phi \cdot \Theta$  двух неизвестных матриц меньшего размера:

$$\Phi = (\varphi_{wt})_{W \times T}, \quad \varphi_{wt} = p(w|t),$$

$$\Theta = (\theta_{td})_{T \times D}, \quad \theta_{td} = p(t|d).$$

Матрица терминов тем

Матрица тем документов

При этом:

матрицы  $\Phi$  и  $\Theta$  должны быть *стохастическими* (столбцы неотрицательны и нормированы → могут представлять дискретное распределение).

## Итог:

задача тематического моделирования сведена к задаче матричного разложения матрицы  $F = p^*$

$wd W \times D$  на стохастические матрицы  $\Phi$  и  $\Theta$ .

Наиболее известный метод матричного разложения – на основе сингулярного разложения матрицы  $F$ :  $T$  главных компонент); решение задачи, аналогичной (3.5).

Но метод главных компонент не подходит для тематического моделирования:

- получаемые матрицы  $\Phi$  и  $\Theta$  общем случае не будут стохастическими;

- квадратичная функция потерь плохо подходит для сравнения распределений с «тяжёлыми хвостами».

Метод решения – на основе принципа максимума правдоподобия:

$$p(D; \Phi, \Theta) = C \cdot \prod_{d \in D} \prod_{w \in d} p(d, w)^{n_{dw}} \rightarrow \max_{\Phi, \Theta},$$

где

$C$  - нормировочный множитель, зависящий только от чисел  $n_{dw}$  (не влияет на положение точки максимума),

$p(d, w) = \sum_{t \in T} p(d) \cdot p(w|t) \cdot p(t|d)$  – вероятность появления пары документ-термин  $(d, w)$ .

После логарифмирования и отбрасывания констант, с учетом требований к матрицам  $\Phi$  и  $\Theta$ , получим задачу максимизации с ограничениями:

$$\begin{cases} L(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in d} n_{dw} \cdot \ln \sum_{t \in T} \varphi_{wt} \cdot \theta_{td} \rightarrow \max_{\Phi, \Theta}, \\ \sum_{w \in W} \varphi_{wt} = 1, \quad \varphi_{wt} \geq 0; \\ \sum_{t \in T} \theta_{td} = 1, \quad \theta_{td} \geq 0. \end{cases}$$

PLSA

**Вероятностный латентный семантический анализ (Probabilistic Latent Semantic Analysis, PLSA)** предложен Т. Хоффманном в 1999 г.

Probabilistic latent semantic indexing.

Для решения задачи (5.2) с ограничениями (5.3) в PLSA предлагается применять различные модификации EM- алгоритма.

PLSA-EM - алгоритм.

*E*-шаг.

Вычисление по формуле Байеса условных вероятностей

$p(t|d, w)$  для всех  $t \in T, w \in d, d \in D$ :

$$p_{tdw} = \frac{\varphi_{wt} \cdot \theta_{td}}{\sum_{s \in T} \varphi_{ws} \cdot \theta_{sd}}.$$

*M*-шаг.

Вычисление новых приближений параметров  $\varphi_{wt}$  и  $\theta_{td}$ :

$$\begin{aligned} \varphi_{wt} &= \frac{n_{wt}}{\sum_{v \in W} n_{vt}}, & n_{wt} &= \sum_{d \in D} n_{dw} \cdot p_{tdw}; \\ \theta_{td} &= \frac{n_{td}}{\sum_{s \in T} n_{sd}}, & n_{sd} &= \sum_{w \in d} n_{dw} \cdot p_{sdw}; \end{aligned}$$

Замечание.

Можно показать: если начальные значения  $\varphi_{wt}$  и  $\theta_{td}$

положительны, то они будут оставаться такими после каждой итерации (хотя ограничение неотрицательности явно не используется в ходе решения).

Имеются различные модификации алгоритма, позволяющие выполнять вычисления более эффективно; существует пакетная версия (batch algorithm), позволяющая обрабатывать большую коллекцию документов по частям (пакетами).

## 29. Аддитивная регуляризация вероятностных тематических моделей. Регуляризованный EM-алгоритм. Стратегии регуляризации.

Задача называется *корректно поставленной* (по Адамару), если ее решение существует, единственно и устойчиво.

Задача стохастического матричного разложения является некорректно поставленной: множество ее решений в общем случае бесконечно.

Общий подход к решению некорректно поставленных обратных задач – *регуляризация*.

**К основному критерию добавляется дополнительный – регуляризатор (учитывает специфику решаемой задачи и знания предметной области)**

Еще один недостаток PLSA - большая размерность пространства параметров → переобучение.

**Аддитивная регуляризация тематических моделей (ARTM)** представляет собой максимизацию линейной комбинации логарифмической функции правдоподобия и регуляризаторов  $R_i \Phi, \Theta$  с неотрицательными коэффициентами регуляризации  $\tau_i, i=1,2,\dots,k$ :

$$\sum_{d \in D} \sum_{w \in d} n_{dw} \cdot \ln \sum_{t \in T} \varphi_{wt} \cdot \theta_{td} + R(\Phi, \Theta) \rightarrow \max_{\Phi, \Theta}$$

$$R(\Phi, \Theta) = \sum_{i=1}^k \tau_i \cdot R_i(\Phi, \Theta)$$

Этот подход не т  
постановку зада

Замечание.

Модель PLSA является частным случаем подхода ARTM при  $R\Phi, \Theta = 0$ .

Для решения задачи (5.4) разработаны различные модификации регуляризованного EM-алгоритма.

Вычисления выполняются с использованием оператора **norm**, преобразующего заданный вектор  $xi, i \in I$ , в вектор вероятностей  $pi, i \in I$ , дискретного распределения:

$$p_i = \text{norm}_{i \in I}(x_i) = \frac{(x_i)_+}{\sum_{j \in I} (x_j)_+}, \quad i \in I,$$

где

$(x)_+ = \max\{0, x\}$  – операция положительной срезки.

отрицат  
элемент  
нормиро

Доказано:

если функция  $R\Phi, \Theta$  непрерывно дифференцируема, то решение задачи (5.4) с ограничениями (5.3) удовлетворяет системе уравнений со вспомогательными переменными  $p_{tdw} = p_t d_w$  (если из решения исключить нулевые столбцы матриц  $\Phi$  и  $\Theta$ ):

$$p_{tdw} = \text{norm}_{t \in T}(\varphi_{wt} \cdot \theta_{td}); \quad (5.5)$$

$$\varphi_{wt} = \text{norm}_{w \in W} \left( n_{wt} + \varphi_{wt} \cdot \frac{\partial R}{\partial \varphi_{wt}} \right), \quad n_{wt} = \sum_{d \in D} n_{dw} \cdot p_{tdw}; \quad (5.6)$$

$$\theta_{td} = \text{norm}_{t \in T} \left( n_{td} + \theta_{td} \cdot \frac{\partial R}{\partial \theta_{td}} \right), \quad n_{td} = \sum_{w \in W} n_{dw} \cdot p_{tdw}. \quad (5.7)$$

Замечание.

В результате решения системы уравнений (5.5) – (5.7) некоторые столбцы матриц  $\Phi$  и  $\Theta$  могут оказаться нулевыми. Такие столбцы не могут определять условные распределения.

Тема  $t$  называется **вырожденной**, если

$$n_{wt} + \varphi_{wt} \cdot \frac{\partial R}{\partial \varphi_{wt}} \leq 0 \quad \text{для всех } w \in W.$$

Обнуление столбца матрицы  $\Phi$  означает: регуляризатору выгодно исключить данную тему из модели.

Документ  $d$  называется **вырожденным**, если

$$n_{td} + \theta_{td} \cdot \frac{\partial R}{\partial \theta_{td}} \leq 0 \quad \text{для всех } t \in T.$$

Вырожденность документа может означать: модель не может описать этот документ (он слишком короткий либо не соответствует тематике коллекции).

Обнуление столбца матрицы  $\Theta$  означает: регуляризатору выгодно исключить данный документ из коллекции.

Если вырожденность тем/документов нежелательна, то ее можно избежать путем постепенного уменьшения коэффициентов регуляризации.

Регуляризованный EM-алгоритм

**Регуляризованный EM-алгоритм** является применением метода простых итераций для решения системы уравнений (5.5) - (5.7).

- На шаге  $E$  выполняются вычисления в соответствии с (5.5);
- на шаге  $M$  - в соответствии с (5.6) и (5.7).

Различные модификации алгоритма различаются частотой обновления параметров модели  $\varphi_{wt}$  и  $\theta_{td}$  по переменным  $nwt$  и  $ntd$ .

Стратегии регуляризации

Задача тематического моделирования обычно включает ряд требований к темам. Темы должны удовлетворять сразу нескольким условиям: интерпретируемости, различности, разреженности и др.

Кроме того: тематическая модель обычно является вспомогательным инструментом для решения различных задач анализа текста (информационного поиска, сегментации, категоризации и др.) → дополнительные требования к модели, предъявляемые решаемой задачей.

При использовании подхода ARTM все требования формализуются в виде критериев регуляризации

$R_i \Phi, \Theta$  и балансируются с помощью коэффициентов  $\tau_i$ .

Коэффициенты  $\tau_i$  подбираются в каждой задаче экспериментально (нужно обеспечить компромисс между требованиями). Более общий подход – изменять коэффициенты в ходе итераций.

**Стратегия регуляризации** - правила изменения коэффициентов регуляризации  $\tau_i$  в ходе итераций EM- алгоритма.

Могут использовать текущие значения параметров модели и метрик качества

### **30. Двухступенчатая модель вероятностного порождения текста.**

**Принцип максимума апостериорной вероятности. Модель латентного размещения Дирихле (LDA).**

В предыдущих рассуждениях предполагалось, что тексты порождаются вероятностной моделью с параметрами

$\Phi, \Theta$ , которые неизвестны и неслучайны.

Предположим теперь: параметры сами являются случайными величинами и подчиняются некоторому априорному распределению  $p(\Phi, \Theta; \gamma)$ , где  $\gamma$  – неслучайный вектор гиперпараметров.

Двухступенчатая модель порождения текста:

1) из априорного распределения порождаются столбцы матриц  $\Phi$  и  $\Theta$ ;

2) на основе матриц  $\Phi$  и  $\Theta$  порождается текстовая коллекция.

Максимизация совместного правдоподобия данных  $X = d, w | d \in D, w \in d$  и модели  $\Phi, \Theta$  приводит к

*принципу максимума апостериорной вероятности (maximum a posteriori probability - MAP):*

$$p(X, \Phi, \Theta; \gamma) = p(X|\Phi, \Theta) \cdot p(\Phi, \Theta; \gamma) = \\ = p(\Phi, \Theta; \gamma) \prod_{d \in D, w \in d} p(d, w|\Phi, \Theta) \rightarrow \max_{\Phi, \Theta, \gamma}$$

Принцип максимума апостериорной вероятности

После логарифмирования получим модификацию задачи (5.3), в которой логарифм априорного распределения является регуляризатором:

$$\sum_{d \in D} \sum_{w \in d} n_{dw} \cdot \ln \sum_{t \in T} \varphi_{wt} \cdot \theta_{td} + \underbrace{\ln p(\Phi, \Theta; \gamma)}_{R(\Phi, \Theta)} \rightarrow \max_{\Phi, \Theta, \gamma} . \quad (5.8)$$

Латентное размещение Дирихле

**Модель латентного размещения Дирихле (latent Dirichlet allocation, LDA)** предложена в 2003 г. как решение проблемы переобучения модели PLSA.

Переобучение связано с избыточной размерностью пространства параметров  
→ следует наложить дополнительные ограничения на матрицы  $\Phi$  и  $\Theta$

Latent Dirichlet Allocation.

Модель LDA

Предположение модели **LDA**:

столбцы матриц  $\Phi$  и  $\Theta$  порождаются распределениями Дирихле с параметрами  $\alpha \in RT$  и  $\beta \in RW$  (являются гиперпараметрами модели **LDA**)

$$Dir(\theta_d; \alpha) = \frac{\Gamma(\alpha_0)}{\prod_{t \in T} \Gamma(\alpha_t)} \prod_{t \in T} \theta_{td}^{\alpha_t - 1}, \quad \alpha_t > 0, \quad \alpha_0 = \sum_t \alpha_t, \quad \theta_{td} > 0, \quad \sum_t \theta_{td} = 1;$$

$$Dir(\varphi_t; \beta) = \frac{\Gamma(\beta_0)}{\prod_{w \in W} \Gamma(\beta_w)} \prod_{w \in W} \varphi_{wt}^{\beta_w - 1}, \quad \beta_w > 0, \quad \beta_0 = \sum_w \beta_w, \quad \varphi_{wt} > 0, \quad \sum_w \varphi_{wt} = 1;$$

Замечания о модели **LDA**.

1. Двухуровневая модель порождения данных формализует предположение о существовании тематических кластерных структур в текстовой коллекции:

- векторы дискретных распределений  $\varphi_{wt} = p(w|t)$  определяют центры тематических кластеров;
- каждый центр порождает тематические части документов - векторы дискретных распределений  $p(w|t, d)$ , группирующиеся вокруг центра.

Замечания о модели **LDA**.

2. Распределения Дирихле могут порождать как плотные, так и разреженные векторы дискретных распределений, что позволяет реализовать предположения разреженности.

Чем меньше  $\beta_w$ , тем более разреженный порождаемый вектор  $\varphi_{wt} = p(w|t)$ .

Согласно (5.8), модели LDA соответствует регуляризатор, с точностью до константы равный сумме логарифмов априорных распределений Дирихле:

$$R(\Phi, \Theta) = \ln \prod_{t \in T} Dir(\varphi_t; \beta) \prod_{d \in D} Dir(\theta_d; \alpha) + const = \\ = \sum_{t \in T} \sum_{w \in W} (\beta_w - 1) \ln \varphi_{wt} + \sum_{d \in D} \sum_{t \in T} (\alpha_t - 1) \ln \theta_{td} .$$

Применение (5.6) и (5.7) к этому регуляризатору приводит к формулам М-шага:

$$\varphi_{wt} = \underset{w \in W}{\text{norm}}(n_{wt} + \beta_w - 1), \quad \theta_{td} = \underset{t \in T}{\text{norm}}(n_{td} + \alpha_t - 1).$$

Модель LDA

При  $\beta w = 1, \alpha t = 1$  априорное распределение Дирихле совпадает с равномерным распределением на единичном симплексе; формулы М-шага превращаются в несмешанные частотные оценки условных вероятностей, и модель LDA превращается в PLSA.

При  $\beta w > 1, \alpha t > 1$  регуляризатор оказывает сглаживающий эффект: распределения  $\varphi_t$  и  $\theta_d$  приближаются к заданным распределениям  $\beta$  и  $\alpha$  соответственно.

При  $0 < \beta w < 1, 0 < \alpha t < 1$  регуляризатор оказывает разреживающий эффект и стремится обнулить малые вероятности.

Реализации модели LDA

Имеются реализации в библиотеках *sklearn*, *Gensim* и др. Реализация в *Gensim*: класс *LdaModel* модуля *models.Ldamodel*. Описание и примеры.

Для обучения модели необходимо создать словарь и корпус, содержащий число вхождений для каждого слова из словаря.

Общее описание, документация класса Dictionary.

### **31. Критерии качества тематических моделей.**

Оценки качества тематических моделей принято делить на две категории:

- **внутренние (intrinsic)** и
- **внешние (extrinsic)** критерии качества.

**Внутренние критерии** характеризуют качество модели по исходной текстовой коллекции.

**Наиболее распространенный - перплексия**

Внутренние и внешние критерии

**Внешние критерии** оценивают полезность модели с точки зрения решаемой задачи и конечных пользователей. Для получения оценки могут привлекаться независимые ассесоры.

Внешние критерии могут быть разнообразны (в зависимости от решаемой задачи):

- качество классификации документов;
- точность и полнота информационного поиска;
- число найденных хорошо интерпретируемых тем;
- качество сегментации текстов,

- результаты сопоставления найденных тем с концептами (специальными наборами взаимосвязанных слов).

Внутренние критерии: перплексия

Перплексия (Perplexity) - известная в компьютерной лингвистике мера качества вероятностной модели языка.

Это мера несоответствия («удивленности») модели (условного распределения  $p(w|d)$ ) термам  $w$ , встречающимся в документах  $d$  коллекции  $D$ .

Перплексия коллекции документов  $D$  для языковой модели  $p(w|d)$  определяется как

$$P(D) = \exp\left(-\frac{1}{n} \sum_{d \in D} \sum_{w \in d} n_{dw} \cdot \ln p(w|d)\right), \quad n = \sum_{d \in D} \sum_{w \in d} n_{dw}$$

Внутренние критерии: перплексия

Интерпретация перплексии:

мера неопределенности появления слов в тексте.

Максимальное значение  $P = W$  - для равномерного распределения

$pwd=1; W$

чем больше распределение слов отличается от равномерного, тем меньше перплексия.

Другой взгляд: перплексия - коэффициент ветвлений текста (количество ожидаемых в среднем различных слов после каждого слова в документе).

Перплексия может быть вычислена по той же коллекции, по которой построена тематическая модель, либо по тестовой (отложенной) коллекции  $D'$  (*hold-out perplexity*).

Эксперименты на больших коллекциях показывают: различие между перплексией на обучающей и тестовой выборках, как правило, не существенно для цели сравнения разных моделей.

На очень больших выборках рекомендуется вычислять перплексию по основным данным.

Недостатки перплексии:

- неочевидность численных значений;
- зависимость не только от качества модели, но и от размерных характеристик коллекции (некорректно сравнивать модели одной и той же коллекции на разных словарях).

Внутренние критерии:

Оценка интерпретируемости тем выполняется только с помощью экспертов (ассессоров).

Тема считается интерпретируемой, если ее можно кратко озаглавить и/или по словам темы можно сформулировать поисковый запрос и получить релевантную поисковую выдачу.

Варианты заданий экспертам:

- рассмотреть темы как последовательности слов, упорядоченные по вероятности появления слова в данной теме, и оценить интерпретируемость темы по некоторой шкале (2 или 5-балльной);
- в список топовых (имеющих большую вероятность) слов темы внедряется лишнее слово, заведомо не принадлежащее этой теме; нужно определить, какое слово из списка лишнее.

Экспертные подходы необходимы при проведении исследований методов моделирования, но затрудняют автоматическое построение моделей.

Результаты проведенных исследований: экспертные оценки хорошо коррелируют с формализованной мерой качества, называемой **когерентностью** (*coherence*).

**Когерентность (согласованность) темы** - мера, которая показывает, насколько слова, встречающиеся рядом в текстах, оказываются в топах одних и тех же тем.

Численной мерой когерентности темы  $t$  является **поточечная взаимная информация**, вычисляемая по  $k$  наиболее вероятным словам темы (*pointwise mutual information, PMI*):

$$PMI_t = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=1}^k PMI(w_i, w_j),$$

где  $w_i$  -  $i$ -ый термин в порядке убывания  $\varphi_{wt}$ ,  
 $k$  обычно полагают равным 10;

$$PMI(u, v) = \ln \frac{|D| \cdot N_{u,v}}{N_u \cdot N_v}.$$

Чем выше величина **PMI**, тем выше неслучайность того, что два слова стоят рядом.

**Когерентность модели** оценивается как средняя когерентность по всем темам.

Может оцениваться по сторонней коллекции (например, по статьям Википедии).

Реализация в Gensim:  
метод `top_topics()` класса `LdaModel`.

Возвращает список (по сформированным темам): представление и значение когерентности для каждой темы.

Документация.

Внешние критерии: задача классификации документов

Задача классификации:

на вход подается документ  $d: ndw d \in D, w \in d$  ;

известна модель классификации, построенная на этапе обучения на коллекции  $D$  ( $pwt$  и  $pc$  ).

Этапы решения:

1) определение тематики данного документа  $p t d$  ;

2) вычисление для данного документа (в соответствии с моделью) условных вероятностей каждого класса:

$$p(c|d) = \sum_{t \in T} p(c|t) \cdot p(t|d).$$

Качество модели определяется тем, насколько хорошо классифицируются документы, представленные векторами  $wt$  и  $ct$ .

Критерии качества:

- число ошибок классификации,
- точность,
- полнота,
- $F$ -мера,
- чувствительность, специфичность, • **AUC-ROC**.