INTERNET STUDIES LAB

HIGHER SCHOOL OF ECONOMICS
NATIONAL RESEARCH UNIVERSITY
SAINT PETERSBURG

Особенности тематического моделирования

Sergei Koltcov

http://linis.hse.ru/en/

Содержимое

Различия в подходах к теории вероятностей

Случайная величина — это величина, которая принимает в результате опыта одно из множества значений, причём появление того или иного значения этой величины до её измерения нельзя точно предсказать.

1. В частотном подходе (классический подход) предполагается, что случайность есть объективная неопределенность. Вероятность рассчитывается из серии экспериментов и является мерой случайности как эмпирической данности. Исторически частотный подход возник из практической задачи: анализа азартных игр — области, в которой понятие серии испытаний имеет простой и ясный смысл.
2. В байесовском подходе предполагается, что случайность характеризует наше незнания. Например, случайность при бросании кости связана с незнанием динамических характеристик игральной кости, сопротивления воздуха и так далее.

Многие задачи частотным методом решить невозможно (точнее, вероятность искомого события строго равна нулю). В то же время интерпретация вероятности как меры нашего незнания позволяет получить отличный от нуля осмысленный ответ.

Понятие вероятности

Вероятность события — Вероятностью события А называют отношение числа благоприятствующих этому событию исходов к общему числу всех равновозможных несовместных элементарных исходов. Например. Вероятность того, что на кубике выпадет четное число, равна следующему отношению P=3/6=1/2.

Понятие условной вероятности

Условной вероятностью события **А** при условии, что произошло событие **В**, называется число **P(A|B)=P(B, A)/ P(B)**,
P(B, A) – произведение вероятностей, **P(B)** – полная вероятность события **В.**

Например. В урне 3 белых и 3 черных шара. Из урны дважды вынимают по одному шару, не возвращая их обратно. Найти вероятность появления белого шара при втором испытании (**событие А**), если при первом испытании был извлечен черный шар (**событие В**).

Решение задачи:
Событие В – это вытаскивание первого шага (а именно черного). Вероятность события B=3/6=1/2 – вер. вытащить черный шар.
События А – это вытаскивание второго шара (а именно белого), так как в урне осталось 5 шаров, то вероятность этого события A=3/5
Таким образом, совместная вероятность событий А и В это произведение вероятностей этих событий P(B, A) =(3/6)*(3/5)=9/30
Полная вероятность события B=1/2

Итоговый результат: {3/6*3/5}/(1/2)=3/5

Формула Байеса

Байесовская вероятность — это интерпретация понятия вероятности, используемое в байесовской теории. Вероятность определяется как степень уверенности в истинности суждения.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

P(A) — априорная вероятность гипотезы *A (заранее известная вероятность)*;

P (A|B) — вероятность гипотезы *A* при наступлении события *B* **(апостериорная вероятность)**;

P(B|A) — вероятность наступления события *B* при истинности гипотезы *A*;

P(B) — полная вероятность наступления события *B*.

P(A|B)) — вероятность наступления события *A* при истинности гипотезы *B*;

Формула Байеса позволяет «переставить причину и следствие»: по известному факту события вычислить вероятность того, что оно было вызвано данной причиной. Таким образом, формула Байеса может быть использована для разработки алгоритмов классификации.

Априорные и апостериорные суждения

1. Предположим, мы хотим узнать значение некоторой неизвестной величины.
2. У нас имеются некоторые знания, полученные до (a priori) наблюдений/эксперимента. Это может быть опыт прошлых наблюдений, какие-то модельные гипотезы, ожидания.
3. В процессе наблюдений эти знания подвергаются постепенному уточнению. После (a posteriori) наблюдений/эксперимента у нас формируются новые знания о явлении.
4.Будем считать, что мы пытаемся оценить неизвестное значение величины P(A|B) посредством наблюдений некоторых ее косвенных характеристик (гипотез).

Формула Байеса (1763 г.) устанавливает правила, по которым происходит преобразование знаний в процессе наблюдений.

Пример оценки надежности компании

Пусть нам нужно оценить надежность компании. Мы предполагаем, что у нас есть три гипотезы о надежности ($Pr(\theta_{i:1,2,3})$). 1. Средняя надежность. 2. Высокая надежность. 3. Низкая надежность.

Априорные значения

Номер гипотезы i	Средняя надежность (Pr1)	Высокая надежность (Pr2)	Низкая надежность (Pr3)
$Pr(\theta_i)$ (число компаний имеющих разные уровни надежности)	0.5 (50%)	0.3 (30%)	0.2 (20%)
Число компаний имеющие прибыл $Pr(y_1; \theta_i)$	0.4 (40%)	0.8 (80%)	0.3 (30%)
Число компаний, осуществляющие своевременный расчет с гос. $Pr(y_2; \theta_i)$	0.7 (70%)	0.9 (90%)	0(0%)

Вопрос, как будут меняться вероятности гипотез (Pr1, Pr2,P3) если мы наблюдаем какую либо величину? Расчет вероятности гипотез ведется при помощи формулы Байеса.

$$\Pr(\theta_j|y) = \frac{\Pr(y|\theta_j)\Pr(\theta_j)}{\Pr(y)} = \frac{\Pr(y|\theta_j)\Pr(\theta_j)}{\sum_{i=1}^{m}\Pr(\theta_i)\Pr(y|\theta_i)}.$$

Пример оценки надежности компании

Пусть мы наблюдаем компанию у которой есть прибыль. Тогда гипотеза (апостериорное значение) того, что данная компания относится к типу средней надежности будет рассчитываться следующим образом.

$$Pr1 = \frac{0.4 * 0.5}{0.4 * 0.5 + 0.8 * 0.3 + 0.3 * 0.2} = 0.4 \text{ (было 0.5)}$$

Вероятность гипотезы о высокой надежности:

$$Pr2 = \frac{0.8*0.3}{0.4*0.5+0.8*0.3+0.3*0.2} = 0.48 \text{ (было 0.3)}$$

Вероятность гипотезы о низкой надежности:

$$Pr3 = \frac{0.3 * 0.2}{0.4 * 0.5 + 0.8 * 0.3 + 0.3 * 0.2} = 0.12 \text{(было 0.2)}$$

Таким образом мы получили апостериорные оценки, которые потом можно использовать как априорные.

Пример оценки надежности компании

Предположим, что фирма, которая имеет прибыль, еще и платит своевременно долги.

Номер гипотезы i	Средняя надежность	Высокая надежность	Низкая надежность
$Pr(\theta_i)$ (число компаний имеющих разные уровни надежности)	0.4	0.48	0.12
Число компаний имеющие прибыл $Pr(y_1; \theta_i)$	0.4	0.48	0.12
Число компаний, осуществляющие своевременный расчет с гос. $Pr(y_2; \theta_i)$	0.7	0.9	0

Тогда новые вероятности гипотез рассчитываются на основании предыдущих расчетов.

$$Pr1 = \frac{0.4 * 0.7}{0.7 * 0.4 + 0.48 * 0.9 + 0 * 0.12} = 0.39 \text{ (было 0.4)}$$

$$Pr2 = \frac{0.48 * 0.9}{0.7 * 0.4 + 0.48 * 0.9 + 0 * 0.12} = 0.607 \text{(было 0.48)}$$

$$Pr3 = \frac{0.12 * 0}{0.7 * 0.4 + 0.48 * 0.9 + 0 * 0.12} = 0 \text{(было 0.12)}$$

Вероятностная постановка задачи классификации

Пусть имеется множество объектов X и конечное множество классов Y . Требуется построить алгоритм способный классифицировать произвольный объект X в рамках заданного множества Y.
Апостериорная вероятность принадлежности объекта X классу Y по формуле Байеса:

$$P(X \mid Y) = \frac{p(X,Y)}{P(X)} = \frac{p(X)P(Y \mid X)}{P(X)}$$

$P(X \mid Y)$ - Апостериорная вероятность

$p(X,Y)$ - Априорная вероятность

Задача классификации заключается в расчете (оценке) апостериорной информации на основании априорной информации. Такая оценка может быть реализована при помощи формулы Байеса. Однако существует проблема оценивания априорной величины p(x,y)

Задача восстановления априорного распределения

p(x,y)

Оценка функции ***p(x,y)*** может быть реализован при помощи трех методов.

1. **Непараметрическое восстановление плотности** основано на локальной аппроксимации плотности p(x) в окрестности классифицируемого объекта $x \in X$. Пример, Алгоритм Парзена-Розенблатта (метод парзеновского окна).
2. **Параметрическое восстановление плотности** основано на предположении, что плотность распределения известна с точностью до параметра, ***p(x,y)*** = $\phi(x; \theta)$, где ϕ фиксированная функция. Пример. Регрессионный анализ, метод наименьших квадратов. Нормальный дискриминантный анализ.
3. **Восстановление смеси плотностей**. Если функцию плотности ***p(x,y)*** не удаётся смоделировать параметрическим распределением, можно попытаться описать её смесью нескольких распределений:

Собственно именно третий метод является основой LDA

$$p(x) = \sum_{j=1}^{k} w_j \varphi(x; \theta_j), \quad \sum_{j=1}^{k} w_j = 1,$$

Постановка задачи о восстановлении плотности распределения: Непараметрическая задача

Непараметрическая задача: Задача восстановления плотности распределения формулируется следующим образом. Задано множество точек X = {x (1), . . . , x(m)} — реализация однородной выборки из неизвестного распределения с плотностью p(x), требуется по выборке X найти некоторое приближение плотности p^(x) ≈ p(x).

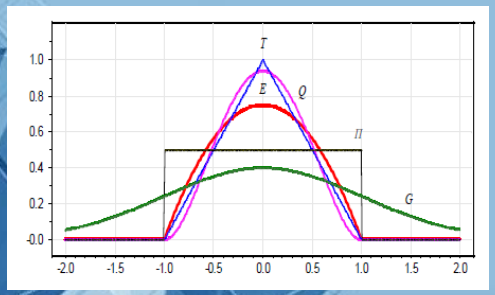
1. **Метод парзеновского окна**;

Оценка плотности Парзена-Розенблатта в одномерном случае имеет вид:

$$\hat{p}_h(x) = \frac{1}{mh}\sum_{i=1}^{m} K\left(\frac{x - x^{(i)}}{h}\right)$$

Многомерный случай:

$$\hat{p}_h(x) = \frac{1}{m}\sum_{i=1}^{m}\prod_{j=1}^{n}\frac{1}{h_j}K\left(\frac{x_j - x_j^{(i)}}{h_j}\right)$$



Непараметрическое восстановление плотности основано на локальной аппроксимации плотности p(x) в окрестности классифицируемого объекта x ∈ X.
K – вид сглаживающей функции, h – размер окна.
Часто используется в Оже и масс спектрометрии

ww.machinelearning.ru/wiki/index.php?title=Метод_Парзеновского_окна_(пример)

Параметрическое восстановление плотности.

Параметрическое восстановление плотности основано на предположении, что плотность распределения известна с точностью до параметра, $p(x,y) = \phi(x; \theta)$, где ϕ фиксированная функция. То есть нам задан вид функций $\phi(x; \theta)$, но не известна величина параметра.

1. **Примером такого подхода служит метод наименьших квадратов**, который позволяет аппроксимировать исходные данные линейной, квадратичной (и так далее) функцией. В ходе решения по методу МНК находятся неизвестные параметры функций.
2. **Принцип максимума правдоподобия.** При вероятностной постановке задачи вместо модели алгоритмов $g(x, \theta)$, аппроксимирующей неизвестную зависимость $y^*(x)$, задаётся модель совместной плотности распределения объектов и ответов $\phi(x, y, \theta)$, аппроксимирующая неизвестную плотность $p(x, y)$. Затем определяется значение параметра θ, при котором выборка данных X_ℓ максимально правдоподобна, то есть наилучшим образом согласуется с моделью плотности. Если наблюдения в выборке X_ℓ независимы, то совместная плотность распределения всех наблюдений равна произведению плотностей $p(x, y)$ в каждом наблюдении: $p(X\ell) = p(x_1, y_1), \ldots, (x_\ell, y_\ell) = p(x_1, y_1) \cdot \ldots \cdot p(x_\ell, y_\ell)$. Подставляя вместо $p(x, y)$ модель плотности $\phi(x, y, \theta)$, получаем функцию правдоподобия (likelihood):

$$L(\theta, X^\ell) = \prod_{i=1}^{\ell} \varphi(x_i, y_i, \theta).$$

Чем выше значение правдоподобия, тем лучше выборка согласуется с моделью. Значит, нужно искать значение параметра θ, при котором значение $L(\theta, X_\ell)$ максимально. В математической статистике это называется принципом максимума правдоподобия.

Математическое ожидание и сопряженные функции.

Математическим ожиданием дискретной случайной величины называется сумма произведений ее возможных значений на соответствующие им вероятности:

$$M(X) = x_1p_1 + x_2p_2 + \ldots + x_np_n$$

$$m_X = \mathrm{M}[X] = \begin{cases} \sum_{i=1}^{N} x_i \cdot p_i \\ \int_{-\infty}^{\infty} x \cdot f(x)dx \end{cases}$$

Математическое ожидание непрерывной величины характеризует среднее значение случайной величины:

Определение сопряженного априорного распределения:

Теорема Байеса, позволяет связать апостериорное распределение исходя из априорного распределения. Пусть у нас есть некая наблюдаемая случайная величина X, которая имеет плотность вероятности p(y|θ) с параметром θ. Апостериорная функция, рассчитывается при помощи формулы Байеса.

$$P(\theta \mid X) = \frac{P(\mathrm{X} \mid \theta) \cdot P(\theta)}{\int P(\mathrm{X} \mid \theta) \cdot P(\theta) d\theta}$$

Если функция $P(\theta \mid X)$ принадлежит тому же классу функций что и функция $P(\theta)$, но с другими параметрами, то такие функции называются сопряженными.

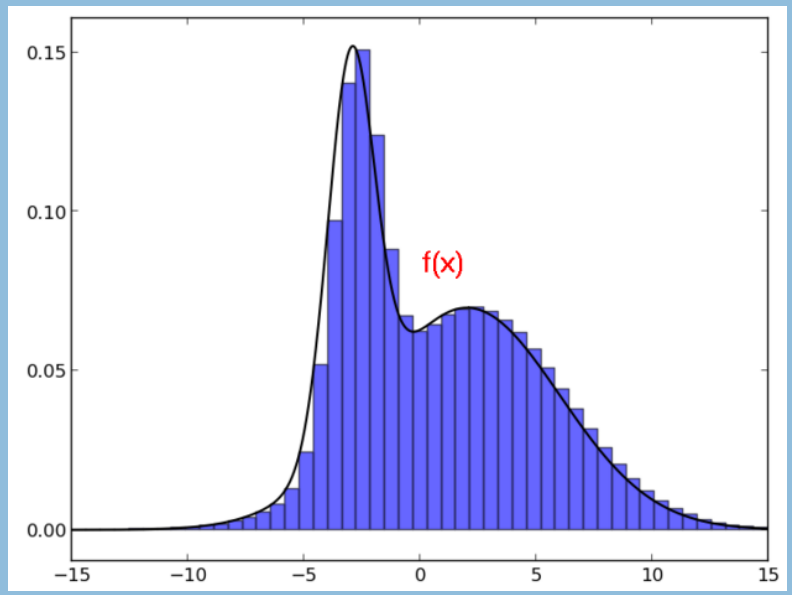
Особенности алгоритмов восстановления плотности распределения.

Для того что бы рассчитать среднее значение надо вычислять вот такой интеграл.

$$M(X) = \int_{-\infty}^{+\infty} xf(x)dx.$$

Это можно сделать следующим образом:

1. Строим график подинтегральной функции.
2. Разбиваем все область на столбики.
3. Будем случайно выбирать из этой области точки. Каждая X-координата будет встречаться с вероятностью, пропорциональной высоте своего столбика.

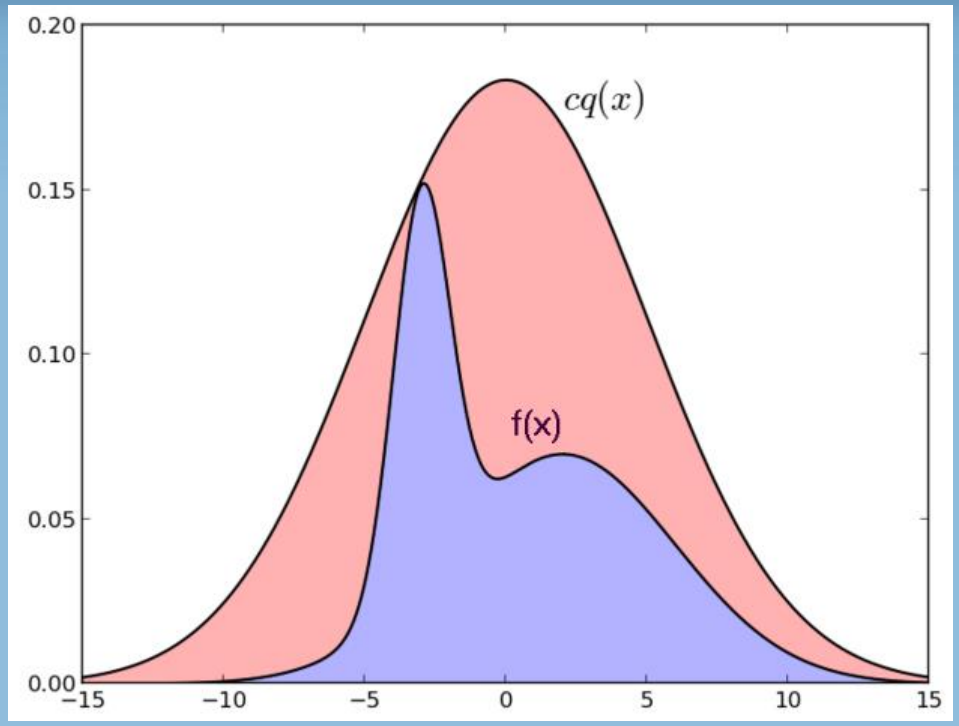


Осталось сообразить как выбирать эти случайные точки.

Алгоритм с отклонением

Мы можем взять другую функцию, такую что наша подинтегральная функция лежит внутри новой функции. Более того мы можем сказать c*q(x)>f(x).

Тогда мы сумеем сэмплировать f(x) таким образом:
берём сэмпл **x** по аспределению $q(\mathbf{x})$;
берём случайное число u равномерно из интервала [0;cq(x)];
вычисляем $f^*(\mathbf{x})$; если оно больше u, то **x** добавляется в сэмплы, а если меньше (т.е. если u не попало под график плотности f^*), то **x** отклоняется.



Если точка под графиком cq^* попадает под график f^*, т.е. в синюю зону, мы её берём; а если над ним, т.е. в красную зону, — не берём. Но для того, чтобы метод работал, нужно, чтобы cq действительно достаточно хорошо приближало f.

Алгоритм Метрополиса-Гастингса

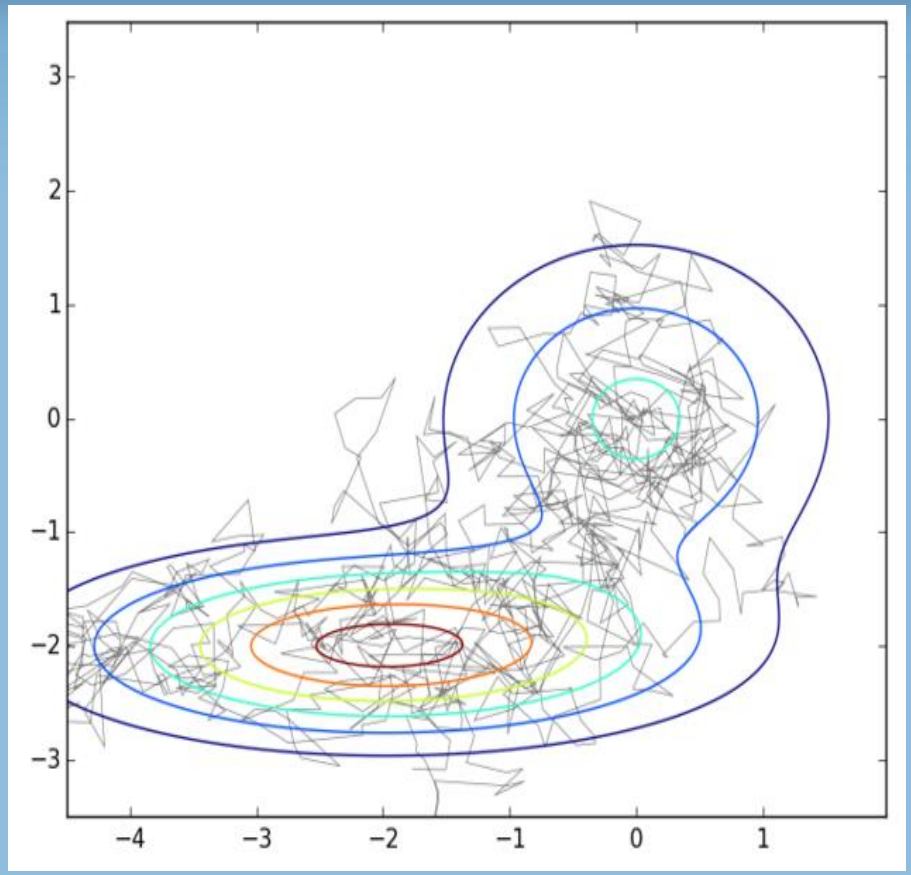
Суть этого алгоритма основана на той же идее. Только вместо накрытия под-интегральной функцией колпаком будем случайно бродит. Такое случайное блуждание является марковской цепью (т.е. его следующая точка зависит только от предыдущей, а памяти никакой нету). Только теперь будем принимать смещение или отвергать. Отвержение и прием шага зависит от некоторой простой функции q(x) которая характеризует вероятность, например нормальное распределение.
На каждом шаге рассчитывается следующие величины:

$$a_1 = \frac{P(x')}{P(x^t)}$$

Отношение функции, на новом и предыдущих шагах

$$a_2 = \frac{Q(x^t|x')}{Q(x'|x^t)}$$

Отношение вероятности, на новом и предыдущих шагах

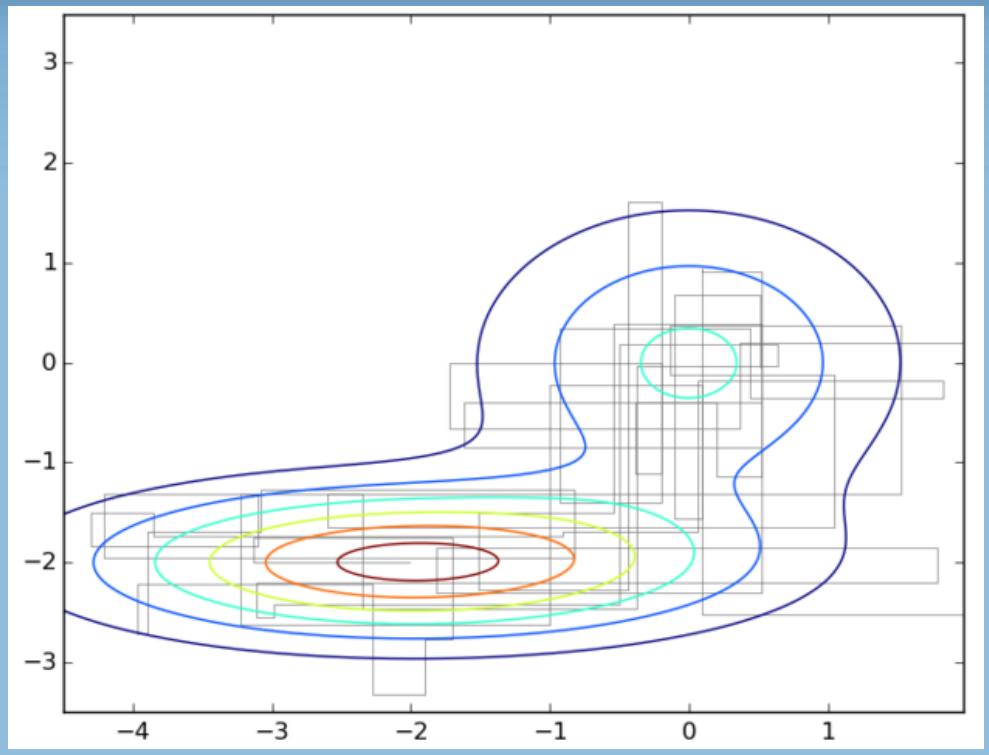


Если a1*a2>1 то принимаем новое значение

мы всегда принимаем шаг, если в эту сторону функция плотности увеличивается, и иногда отвергаем, если уменьшается

Алгоритм сэмплирования по Гиббса

Идея сэмплирования по Гиббсу совсем простая: предположим, что мы находимся в очень большой размерности, вектор **x** очень большой, и нам сложно выбирать весь сэмпл сразу (то есть по всем осям), не получается. Давайте попробуем выбирать сэмпл не весь сразу, а покомпонентно. Это особенно удобно, если каждая компонента не зависит друг от друга, то есть многомерный вектор это произведение множества одномерных функций.



$$p(x_i|x_1^{t+1},\ldots,x_{i-1}^{t+1},x_{i+1}^{t},\ldots,x_n^{t})$$

Для сэмплирования по Гиббсу не нужно никаких особенных предположений или знаний.

Вывод формул LDA.

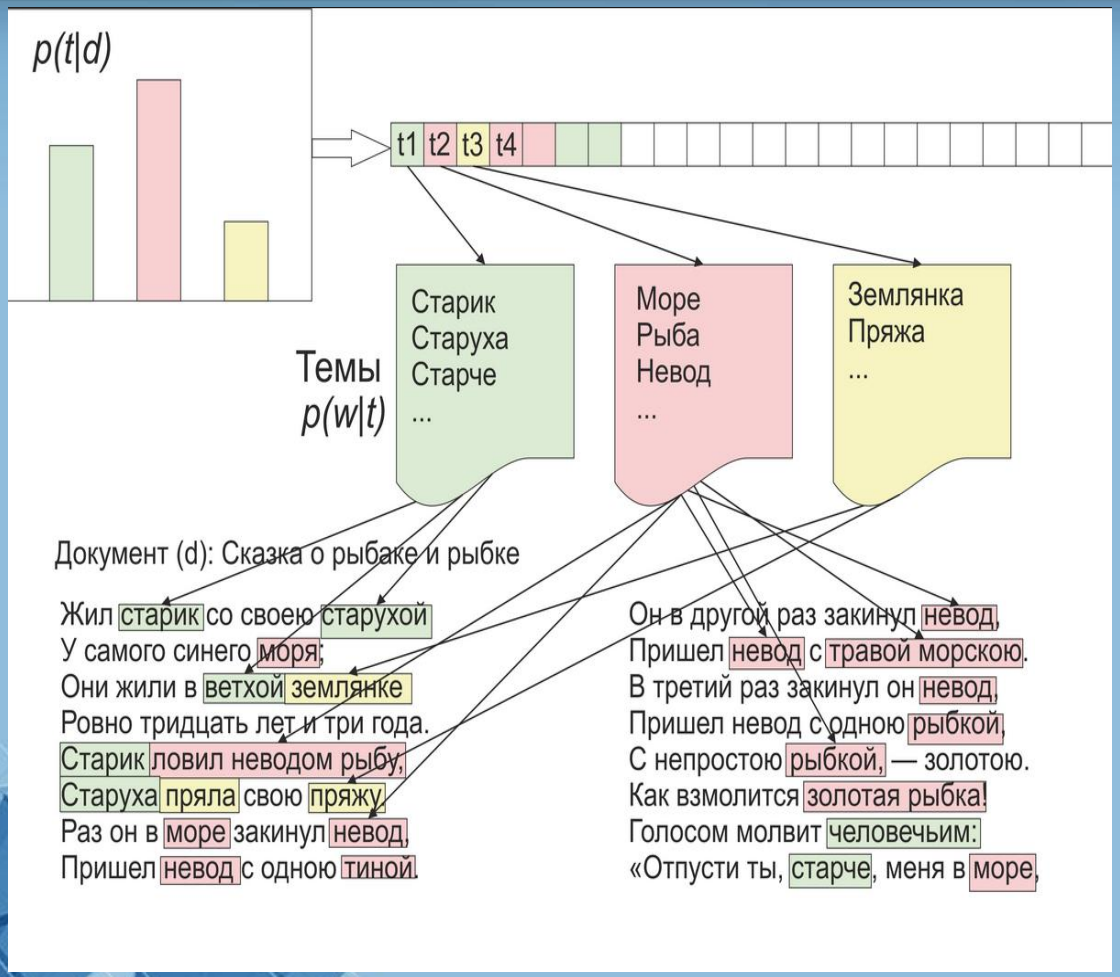
Основное предположение тематической модели Latent Dirichlet Allocation состоит в том, что каждый документ с некоторой вероятностью может принадлежать множеству тематик. Тема эта совокупность слов, где каждое слово имеет некоторую вероятность принадлежности к данной тематике.

Формально тема определяется как дискретное (мультиномиальное) вероятностное распределение в пространстве слов заданного словаря.

Тематическим моделированием называется решение задачи, обратной классификации. Каждый документ в корпусе текстов рассматривается как наблюдаемая случайная независимая выборка слов (мешок слов), порождённая некоторым, скрытым (латентным) множеством тем. По этим данным требуется восстановить вероятностные распределения всех тем в корпусе и определить, каким именно подмножеством тем порождён каждый документ.

Тематическое моделирование основано на применении формулы Байеса, в которой распределение слов и тем выражено в виде смеси плотностей распределений слов и документов.

Вывод формул LDA.



$$p(z, w \mid \alpha, \beta) = p(w \mid z, \beta) \cdot p(z \mid \alpha)$$

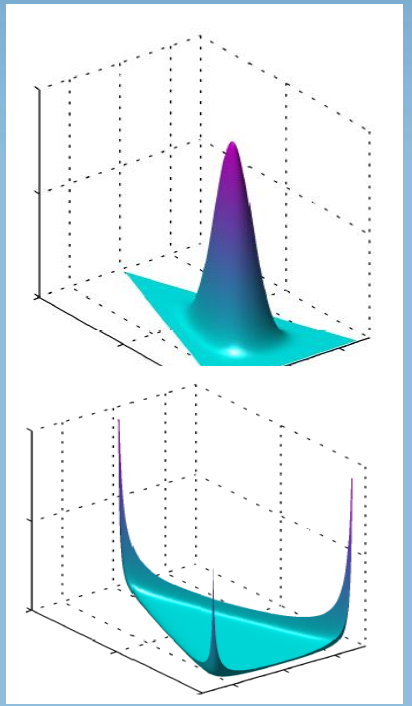
Тематическая модель (topic model) — модель коллекции текстовых документов, которая определяет, к каким темам относится каждый документ коллекции. Алгоритм построения тематической модели получает на входе коллекцию текстовых документов. На выходе для каждого документа выдаётся числовой вектор, составленный из оценок степени принадлежности данного документа каждой из тем.

Вывод формул LDA.

Мультиномиальное распределение имеет следующий вид (совместное распределение вероятностей случайных величин):

$$P(\mathrm{x} \mid \mathrm{p}) = \frac{n!}{\Pi_{i=1}^{K} x_i !} \Pi_{i=1}^{K} p_i^{x_i}$$

где x_i независимые одинаково распределенные случайные величины, p_i - функции распределений случайных величин.



Распределение Дирихле имеет следующий вид:

$$P(\mathrm{p};\alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^{K} p_i^{\alpha_i - 1}$$

Это означает, что если априорное распределение P является распределением Дирихле Dir(p;α), и x сгенерировано мультиномиальным распределением, то апостериорное распределение p(p x, α) также является распределением Дирихле:

$$P(\mathrm{p} \mid \mathrm{x}, \alpha) = Dir(\mathrm{p} \mid \mathrm{x} + \alpha) = \frac{1}{B(\mathrm{x} + \alpha)} \prod_{i=1}^{\alpha} p_i^{x_i + \alpha_i - 1}$$

Соответственно вычисление интегралов упрощается.

Вывод формул LDA.

Термины: **D** – пространство документов, **W** – пространство слов, **Z** – пространство тем. Темы являются скрытыми параметрами, которые должны быть найдены. Причем оценка основана на двух вещах: 1. Оценка Θ, Φ производится как математическое ожидание. 2. В качестве функций используются мультиномиальные функции и функции Дирихле.

$$p(w \mid z, \beta) = \int p(w \mid z, \Phi)\, p(\Phi \mid \beta)\, d\Phi \qquad p(z \mid \alpha) = \int p(z \mid \Theta)\, p(\Theta \mid \alpha)\, d\Theta$$

Θ, Φ : матрица слова – темы и документы – темы.
Эти матрицы могут быть найдены двумя способами.

Вариационный вывод

$$L(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln \sum_{t \in T} \phi_{wt} \theta_{td} \to \max$$

Свернутое семплирование по Гиббсу

$$P(z_i = j \mid w_i = m, z_{-i}, w_{-i}) \approx \frac{C^{WT}_{m,j} + \beta}{\sum_{m'} C^{WT}_{m',j} + V\beta} \cdot \frac{C^{DT}_{d,j} + \alpha}{C^{DT}_{d,j'} + \alpha T}$$

Модель LDA представляет собой произведение матриц.

$$F[documents \times words] = \Theta[documents \times topics] \cdot \Phi[topics \times words]$$

Свернутое семплирование по Гиббсу.

$$p(z, w \mid \alpha, \beta) = p(w \mid z, \beta) \cdot p(z \mid \alpha) = P(z_i = j \mid w_i = m, z_{-i}, w_{-i}) \approx \frac{C^{WT}_{m,j} + \beta}{\sum_{m'} C^{WT}_{m',j} + V\beta} \cdot \frac{C^{DT}_{d,j} + \alpha}{C^{DT}_{d,j'} + \alpha T}$$

$C^{WT}_{m,j}$ - Матрица; в каждой ячейке находится число сколько раз слово **w** было связанно с темой **t**,

$C^{DT}_{d,j}$ - Матрица; в каждой ячейке находится число сколько раз слово **w** в документе **d** связанно с темой **t**.

$\sum_{m'} C^{WT}_{m',j} = n_t$ - Вектор; в каждой ячейке находится общее число слов
- связанно с темой **t**,

$C^{DT}_{d,j'} = n_d$ Дина документа **d** словах.

Результат моделирования:

1. Распределение слов по темам.

$$\theta_{dj} = \frac{C^{DT}_{d,j} + \alpha}{C^{DT}_{d,j'} + T\alpha}$$

2. Распределение документов по темам.

$$\phi_{m,j} = \frac{C^{WT}_{m,j} + \beta}{\sum_{m'} C^{WT}_{m',j} + V\beta}$$

Алгоритм сэмплироания

На входе: коллекция документов D, число тем |T|, Число итераций;
Initialization: φ(w,t), θ(t,d) для всех документов и слов d ∈ D, w ∈ W , t ∈ T;

Внешний цикл по документам (i). Длинна цикла=числу документов

Внутренний цикл. Длина цикла = количество слов в текущем документе.

1. Берем документ i.
2. Выбираем слово **k** из документа **i**.
3. Вычисляем номер темы **t** для слова **k.**

3.1. Вычисляем величину P(z) для текущего слова и для каждой темы P(1…T).

$$P(z_i = j \mid w_i = m, z_{-i}, w_{-i}) \approx \frac{C^{WT}_{m,j} + \beta}{\sum_{m'} C^{WT}_{m',j} + V\beta} \cdot \frac{C^{DT}_{d,j} + \alpha}{C^{DT}_{d,j'} + \alpha T}$$

3.2. генерируем случайное число U. Сравниваем U с каждой величиной P(1…T). Например, если U< P(5), то текущему слову присваивается номер темы 5.

Конец внутреннего цикла.

Апдейтинг всех счетчиков

Конец внешнего цикла.

Расчет матриц φ(w,t), θ(t,d) но основании счетчиков.

$$\theta_{dj} = \frac{C^{DT}_{d,j} + \alpha}{C^{DT}_{d,j'} + T\alpha}$$

$$\phi_{m,j} = \frac{C^{WT}_{m,j} + \beta}{\sum_{m'} C^{WT}_{m',j} + V\beta}$$

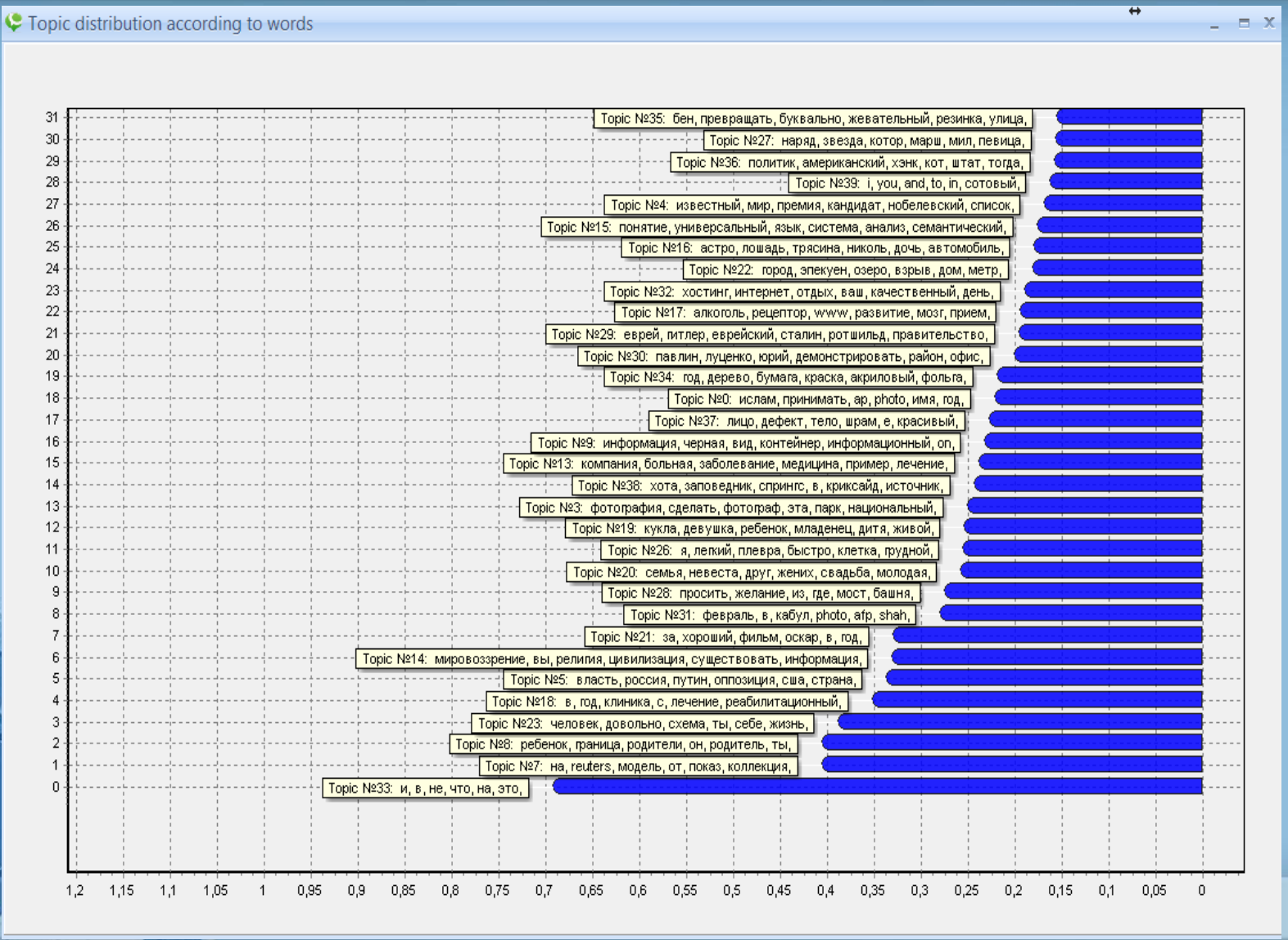
Результат: распределение слов по темам

Words with high probability

	1	2	3	4	5	6	7	8
1	ислам: 0,014347	образование: 0,003430	социальный: 0,006741	фотография: 0,018203	известный: 0,007305	власть: 0,019949	медведь: 0,007718	на: 0,039698
2	принимать: 0,013508	выступление: 0,003430	ru: 0,004531	сделать: 0,009500	мир: 0,006276	россия: 0,018902	алгоритм: 0,003370	reuters: 0,025063
3	ар: 0,007635	черт: 0,002323	гибельный: 0,002321	фотограф: 0,008775	премия: 0,005247	путин: 0,011571	лес: 0,003370	модель: 0,024453
4	photo: 0,007635	захват: 0,002323	белая: 0,002321	эта: 0,007325	кандидат: 0,005247	оппозиция: 0,011571	король: 0,002283	от: 0,018355
5	имя: 0,006796	жаловаться: 0,002323	обязательство: 0,002321	парк: 0,006599	нобелевский: 0,005247	сша: 0,011048	пикник: 0,002283	показ: 0,017135
6	год: 0,005957	дестабилизация: 0,002323	ваш: 0,002321	национальный: 0,005874	список: 0,004219	страна: 0,009477	какой-либо: 0,002283	коллекция: 0,0146
7	ты: 0,005118	nstarikov: 0,002323	счастие: 0,002321	заповедник: 0,005149	включать: 0,004219	демократия: 0,007906	слово: 0,002283	упасть: 0,011647
8	взять: 0,004279	физиологический: 0,002323	бессмысленный: 0,002321	африка: 0,005149	столица: 0,003190	под: 0,007383	отдыхать: 0,002283	инвалид: 0,011037
9	группа: 0,003440	родин: 0,002323	обама: 0,001216	рассказывать: 0,004424	номинантов: 0,003190	революция: 0,007383	несколько: 0,002283	мода: 0,010427
10	арт: 0,003440	развиваться: 0,002323	директива: 0,001216	отдыхать: 0,003699	лонг: 0,003190	война: 0,007383	книга: 0,002283	в: 0,009818
11	дейв: 0,003440	пусть: 0,002323	приближаться: 0,001216	южный: 0,003699	лист: 0,003190	голосовать: 0,006859	начало: 0,002283	февраль: 0,009818
12	блэйки: 0,003440	временить: 0,002323	привлекательный: 0,001217	индий: 0,003699	мэннинг: 0,002161	против: 0,006859	спокойно: 0,002283	полицейский: 0,009
13	стоун: 0,003440	что: 0,002323	неудобно: 0,001216	тупик: 0,003699	брэдли: 0,002161	этап: 0,006859	семейный: 0,002283	неделя: 0,009818
14	африка: 0,003440	лексический: 0,001217	зацеплять: 0,001216	сова: 0,002973	юлий: 0,002161	видео: 0,006335	вернуться: 0,002283	ла-пас: 0,008598
15	член: 0,003440	ржавый: 0,001217	маммограмму: 0,001216	рак: 0,002973	скрывать: 0,002161	народ: 0,005812	голодный: 0,002283	фотография: 0,007
16	q: 0,002601	баглан: 0,001217	лента: 0,001216	national: 0,002973	русский: 0,002161	оранжевый: 0,005812	на: 0,002283	боливия: 0,007378
17	tip: 0,002601	алсиндор: 0,001217	полуостров: 0,001216	рысь: 0,002973	тимошенко: 0,002161	общество: 0,005812	кровавый: 0,002283	во: 0,006769
18	нация: 0,002601	жать: 0,001217	дамба: 0,001216	малыш: 0,002973	великий: 0,002161	чего: 0,004241	ролик: 0,002283	время: 0,006769
19	али: 0,002601	камал: 0,001217	джами: 0,001216	побережье: 0,002973	отмечать: 0,002161	новый: 0,004241	мы: 0,002283	david: 0,006769
20	включать: 0,002601	зодиакальный: 0,001217	julie: 0,001216	род: 0,002973	его: 0,002161	манипуляция: 0,003717	изначально: 0,001196	подиум: 0,006159
21	мохаммед: 0,002601	контракт: 0,001217	жилье: 0,001216	птица: 0,002973	себе: 0,002161	снайпер: 0,003717	abbyy: 0,001196	mercado: 0,006159
22	x: 0,002601	демонстрантка: 0,001217	багор: 0,001216	park: 0,002973	падение: 0,002161	интернет: 0,003717	comas: 0,001196	путь: 0,004939
23	сейчас: 0,002601	нестабильный: 0,001217	бессрочный: 0,001216	запечатлевать: 0,002973	помещение: 0,002161	гражданский: 0,003717	дополнять: 0,001196	творение: 0,004939
24	коран: 0,002601	бесплодие: 0,001217	зерно: 0,001216	серенгети: 0,002973	комитет: 0,002161	выбор: 0,003717	гаммакурта: 0,001196	набрасываться: 0,0
25	верить: 0,002601	халена: 0,001217	невинный: 0,001216	под: 0,002973	буда: 0,002161	интерес: 0,003194	фашистский: 0,001196	падение: 0,004330

Каждая колонка это распределение слов. Соответственно просматривая эти колонки можно выбрать нужны темы для анализа

Результат: распределение слов по темам



Отсортированный список тем по весу, где вес темы это сумма всех вероятностей в данной теме. В примере расчет производился по первым 100 словам.

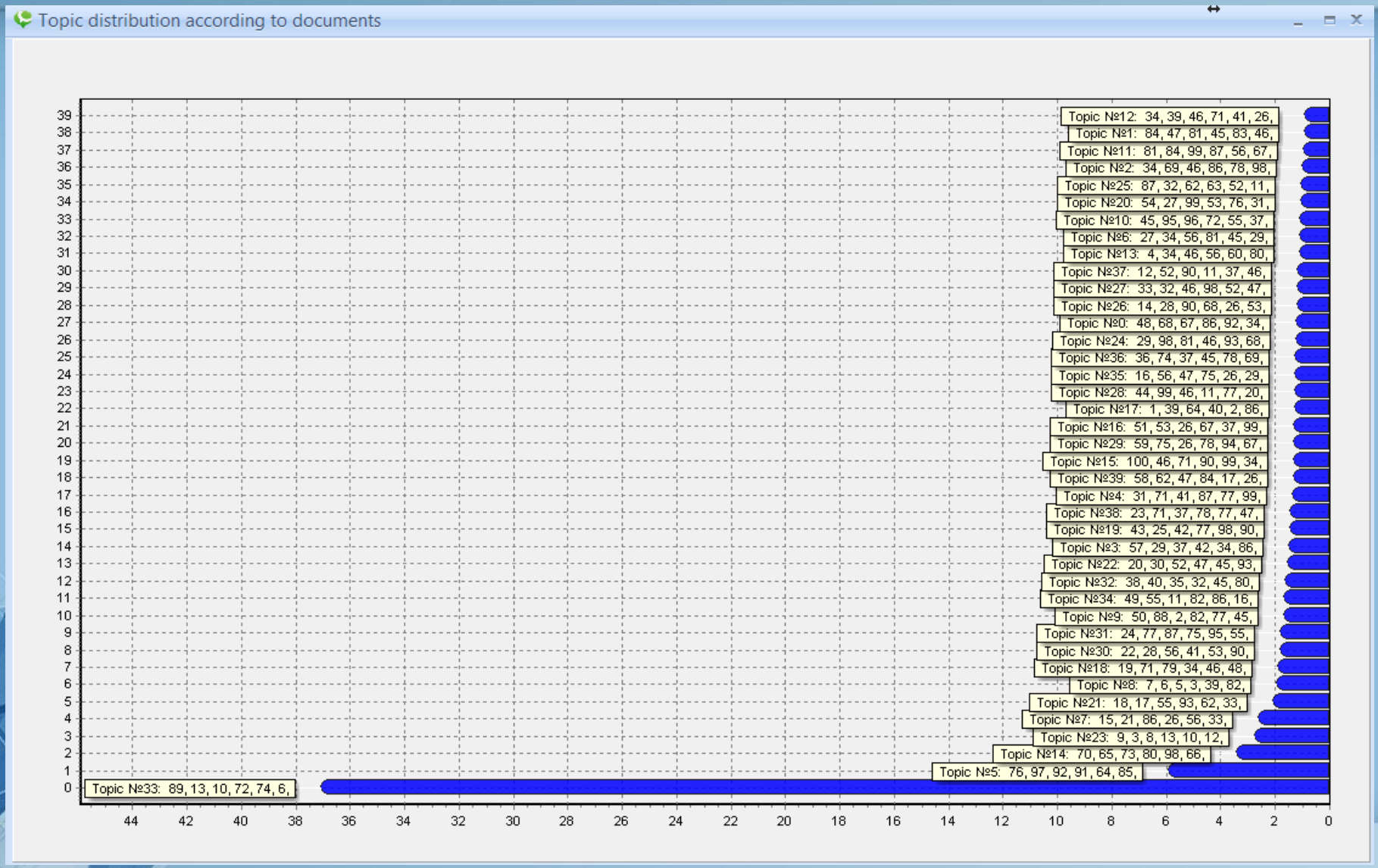
Результат: распределение документов по темам

Documents with high probability

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	48: 0,421482	84: 0,146154	34: 0,106557	57: 0,596774	31: 0,452247	76: 0,435268	27: 0,236979	15: 0,906034	7: 0,391587	50: 0,563670	45: 0,140244	81: 0,077778	34: 0,139344	4: 0,350147
2	68: 0,064103	47: 0,042763	69: 0,080508	29: 0,145320	71: 0,087209	97: 0,323529	34: 0,090164	21: 0,748333	6: 0,339714	88: 0,218333	95: 0,083929	84: 0,069231	39: 0,054245	34: 0,057377
3	67: 0,054124	81: 0,033333	46: 0,060345	37: 0,077320	41: 0,069588	92: 0,306250	56: 0,087963	86: 0,100000	5: 0,325306	2: 0,206537	96: 0,062319	99: 0,058442	46: 0,043103	46: 0,043103
4	86: 0,047368	45: 0,030488	86: 0,047368	42: 0,066832	87: 0,059677	91: 0,292526	81: 0,055556	26: 0,087838	3: 0,051165	82: 0,048387	72: 0,053867	87: 0,043548	71: 0,040698	56: 0,041667
5	92: 0,043750	83: 0,030201	78: 0,041985	34: 0,040984	77: 0,055556	64: 0,284483	45: 0,042683	56: 0,078704	39: 0,049528	77: 0,043210	55: 0,053030	56: 0,041667	41: 0,038660	60: 0,034545
6	34: 0,040984	46: 0,025862	98: 0,041667	86: 0,036842	99: 0,045455	85: 0,253817	29: 0,036946	33: 0,049043	82: 0,048387	45: 0,042683	37: 0,046392	67: 0,038660	26: 0,033784	80: 0,028846
7	79: 0,033333	56: 0,023148	55: 0,037879	81: 0,033333	66: 0,034247	96: 0,242029	95: 0,030357	34: 0,040984	25: 0,044479	83: 0,030201	93: 0,039474	72: 0,031768	99: 0,032468	27: 0,028646
8	47: 0,029605	97: 0,022876	37: 0,036082	27: 0,023438	81: 0,033333	60: 0,241818	35: 0,026814	47: 0,036184	9: 0,043534	71: 0,029070	86: 0,036842	79: 0,023810	84: 0,023077	47: 0,023026
9	11: 0,026786	80: 0,022436	40: 0,035865	55: 0,022727	74: 0,023504	61: 0,240809	89: 0,026699	81: 0,033333	77: 0,043210	11: 0,026786	26: 0,033784	52: 0,022124	3: 0,022879	39: 0,021226
10	46: 0,025862	39: 0,021226	56: 0,023148	45: 0,018293	47: 0,023026	63: 0,224138	53: 0,019481	92: 0,031250	42: 0,042079	47: 0,023026	67: 0,028351	50: 0,020599	80: 0,022436	11: 0,020833
11	26: 0,020270	53: 0,019481	89: 0,021845	71: 0,017442	52: 0,022124	93: 0,223684	30: 0,017949	35: 0,023659	66: 0,029680	74: 0,019231	90: 0,027778	64: 0,020115	53: 0,019481	50: 0,020599
12	99: 0,019481	71: 0,017442	77: 0,018519	63: 0,017241	58: 0,022013	83: 0,218121	63: 0,017241	69: 0,021186	4: 0,029087	36: 0,018519	32: 0,027273	96: 0,018841	63: 0,017241	43: 0,018719
13	63: 0,017241	63: 0,017241	45: 0,018293	90: 0,016667	35: 0,020505	95: 0,205357	82: 0,016129	45: 0,018293	78: 0,026718	27: 0,018229	63: 0,017241	92: 0,018750	49: 0,016432	63: 0,017241
14	82: 0,016129	90: 0,016667	71: 0,017442	82: 0,016129	63: 0,017241	79: 0,195238	74: 0,014957	41: 0,018041	46: 0,025862	30: 0,017949	82: 0,016129	71: 0,017442	82: 0,016129	82: 0,016129
15	58: 0,015723	60: 0,016364	63: 0,017241	98: 0,013889	50: 0,016854	86: 0,194737	98: 0,013889	71: 0,017442	84: 0,023077	42: 0,017327	76: 0,015625	63: 0,017241	95: 0,016071	87: 0,014516
16	98: 0,013889	82: 0,016129	60: 0,016364	56: 0,013889	82: 0,016129	94: 0,183019	93: 0,013158	63: 0,017241	99: 0,019481	63: 0,017241	64: 0,014368	29: 0,017241	98: 0,013889	64: 0,014368
17	93: 0,013158	9: 0,015948	82: 0,016129	52: 0,013274	28: 0,015823	75: 0,159898	68: 0,012821	61: 0,016544	53: 0,019481	65: 0,016883	79: 0,014286	48: 0,016960	56: 0,013889	98: 0,013889
18	89: 0,012136	86: 0,015789	8: 0,013980	20: 0,013235	98: 0,013889	69: 0,148305	42: 0,012376	100: 0,016484	92: 0,018750	40: 0,014768	98: 0,013889	82: 0,016129	62: 0,013393	93: 0,013158
19	91: 0,011598	11: 0,014881	25: 0,013804	93: 0,013158	62: 0,013393	67: 0,146907	66: 0,011416	82: 0,016129	45: 0,018293	79: 0,014286	56: 0,013889	38: 0,015487	52: 0,013274	68: 0,012821
20	76: 0,011161	64: 0,014368	20: 0,013235	68: 0,012821	93: 0,013158	66: 0,139269	76: 0,011161	11: 0,014881	63: 0,017241	98: 0,013889	62: 0,013393	91: 0,014175	93: 0,013158	72: 0,012431
21	81: 0,011111	98: 0,013889	93: 0,013158	69: 0,012712	27: 0,013021	72: 0,122928	40: 0,010549	64: 0,014368	43: 0,016223	93: 0,013158	52: 0,013274	31: 0,014045	27: 0,013021	78: 0,011450
22	74: 0,010684	52: 0,013274	68: 0,012821	10: 0,012521	68: 0,012821	78: 0,110687	96: 0,010145	79: 0,014286	33: 0,015550	68: 0,012821	41: 0,012887	98: 0,013889	68: 0,012821	81: 0,011111

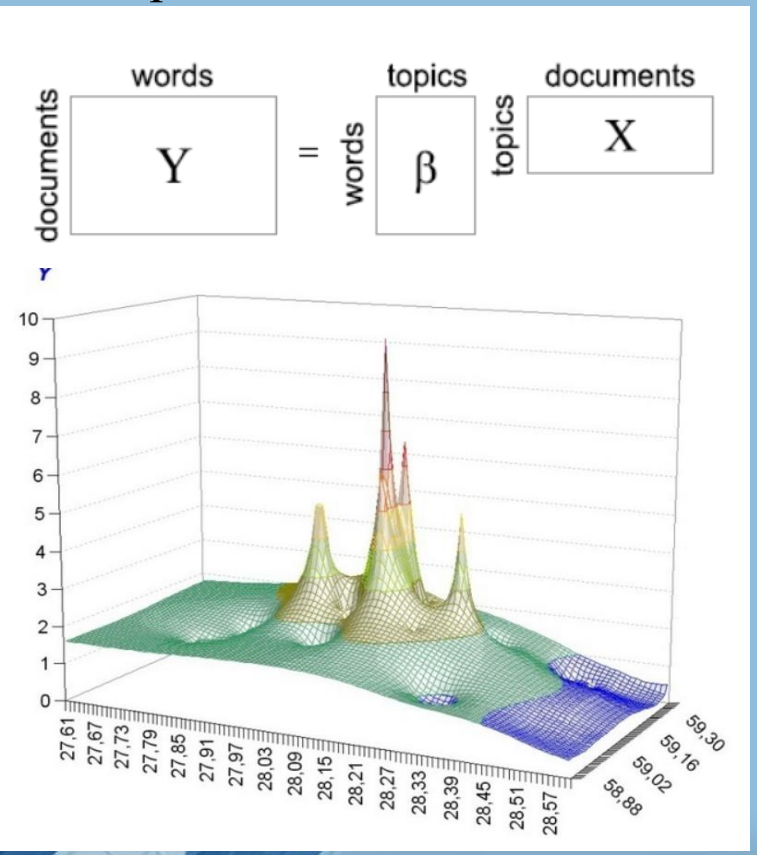
Каждая колонка это распределение документов. Соответственно просматривая эти колонки можно выбрать нужны темы для анализа

Результат: распределение документов по темам



Mathematical vision of LDA

Матрица **F** представляет исходные данные. Исходный датасет можно представить в виде произведение двух матриц меньшего размера. Но данная аппроксимация может быть сделанна разными способами.



$$F = \Theta \cdot \Phi = (\Theta \cdot R) \cdot (R^{-1}\Phi) = \Theta^{'} \cdot \Phi^{'}$$

Это значит, что при одних и тех же размерах матриц, содержимое матриц будет различным. То есть одному и тому же датасету будут соответствовать одинаковые по размеру матрицы, но их наполненность будет разная.

1. Проблема оценки сходства решений между собой решается за счет использований метрик Kullback-Leibler divergence и Jaccard coefficient.
2. Решение проблемы множества решений можно искать при помощи идеологии регуляризации.

Evaluating LDA quality with Kullback–Leibler divergence and Jaccard coefficient

The **Kullback-Leibler divergence** (**K**) является мерой сходства двух распределений. Нормализованная мера **K** может быть расчитана по следующим формулам.

$$Kn = (1 - \frac{K}{Max}) \cdot 100\% \quad \text{where} \quad Kn = 0.5 \sum_{k=1}^{W} \Omega_k^1 \log(\frac{\Omega_k^1}{\Omega_k^2}) + 0.5 \sum_{k=1}^{W} \Omega_k^2 \log(\frac{\Omega_k^2}{\Omega_k^1})$$

Если Kn=100%, две темы идентичны. Если K=0 темы полностью различны.

Jaccard coefficient: Jc=a/(a+b-k).

где**a** – число слов в тему 1, **b** – число слов в теме 2, **k** – число одинаковых слов в двух тема.

Jc = 1, если две темы идентичны, если **Jc = 0** тогда темы различны.

Пример сходства и различия тем

Level 90 - 93% (and more) means that first 50 words are almost identical.

Similarity 0.935			
USA	0.04734	USA	0.03567
American	0.02406	American	0.01804
Syria	0.02082	Syria	0.01758
Obama	0.01374	country	0.01495
weapon	0.01343	war	0.01361
war	0.01309	military	0.01246
president	0.01169	weapon	0.01084
UN	0.01018	Russia	0.01004
military	0.01014	Obama	0.00996
country	0.01005	president	0.0096
chemical	0.00944	UN	0.00869
Syrian	0.00851	international	0.00769

Level about 85%: topics are completely different.

Similarity 0.854			
USA	0.04734	water	0.01758
American	0.02406	help	0.01296
Syria	0.02082	city	0.01262
Obama	0.01374	far	0.01199
weapon	0.01343	house	0.01064
war	0.01309	east	0.0104
president	0.01169	region	0.00945
UN	0.01018	dam	0.0091
military	0.01014	flood	0.00904
country	0.01005	resident	0.00839
chemical	0.00944	injured	0.00714
Syrian	0.00851	FRS	0.00698

Слова в темах отсортированны по вероятности.

Word ratio и document ratio

Следует отметить, что в качестве исходного распределения слов и документов используется равномерное распределение. Это значит, что вероятности всех документов изначально (при первой итерации) равно следующей величине **1/K**, где **K – число тем**. Соответственно сумма вероятностей одного документа по всем темам равна 1.

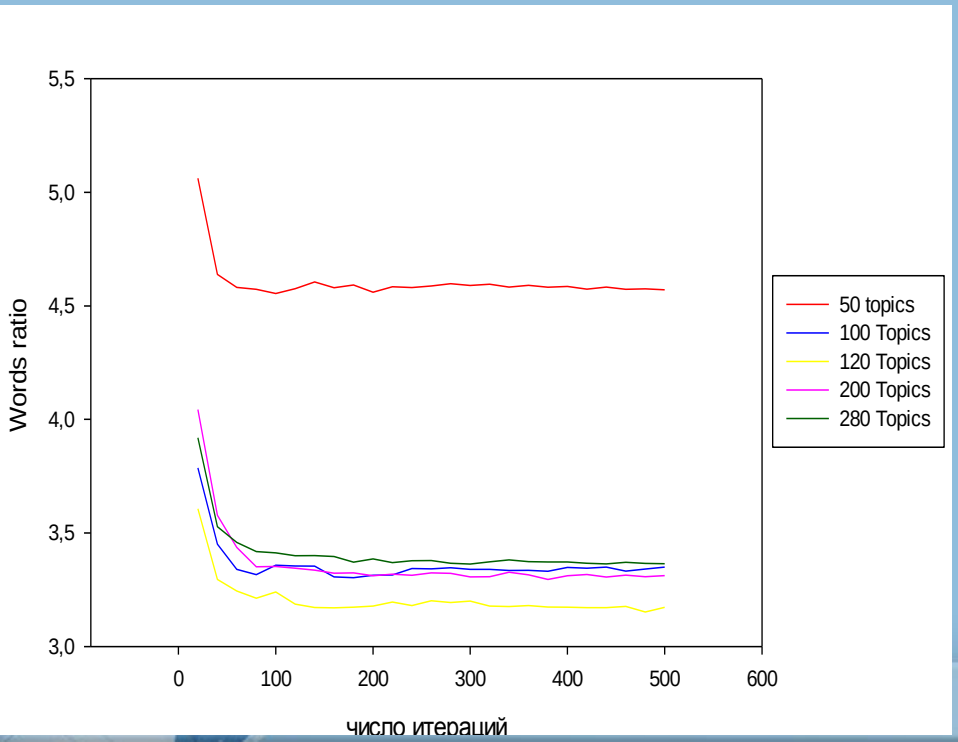
Аналогично, равномерное распределение слов означает, что вероятность слова в теме равна **1/V**, где **V – размер словаря**. Таким образом, вероятность принадлежности всех слов одной теме (то есть сумма вероятностей всех слов внутри одной темы) тоже равна 1.

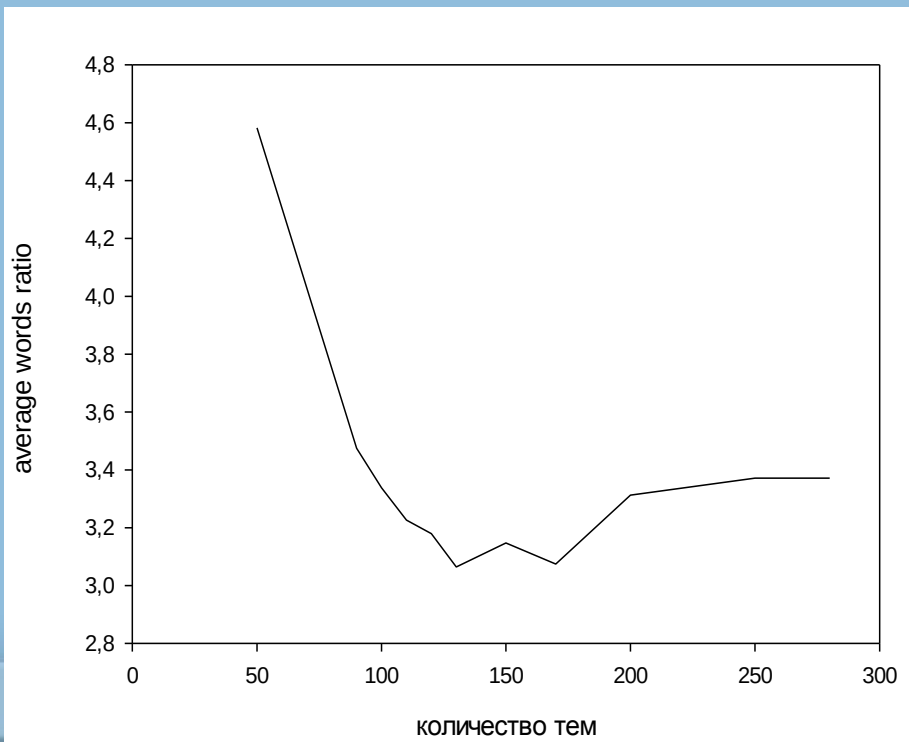
В ходе тематического моделирования производится пересчет вероятностей слов и документов по темам, однако **сумма вероятностей одного документа по темам всегда равна единице**. **Сумма вероятностей всех слов внутри одной теме также всегда равна единице.** Это означает, что вероятности документа по темам перераспределяются таким образом, что часть вероятностей становится больше величины **1/K**, а часть меньше этой величины.

Word ratio и document ratio

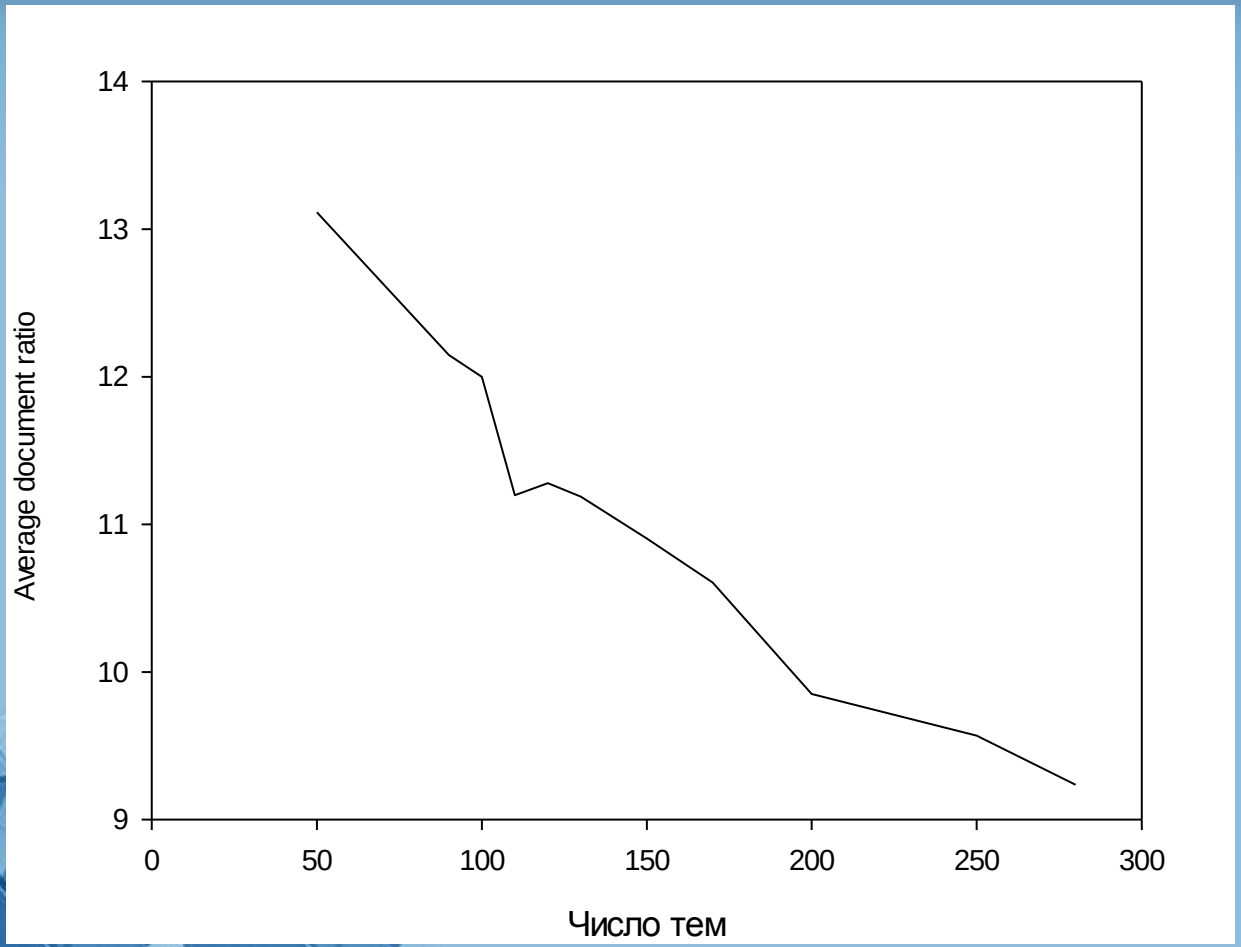
‘Documents ratio’ это параметр, который характеризует отношение числа документов во всех темах, чьи вероятности выше равномерной величины (**1/K**), к общему числу документов во всех темах.

‘Words ratio’ – это величина равная отношению числа слов во всех темах, чьи вероятности выше значения 1/V, к общему числу слов во всех темах.





Word ratio и document ratio



Можно применить теорию скачков к оценке оптимального числа кластеров. В приведенном примере, оптимальным является число 120 тем, так как наблюдаются скачки в document и word ration именно для 120 тем.

Регуляризация тематического моделирования

Задача тематического моделирования может иметь много решений. Неединственность решения влечёт неустойчивость алгоритма. В силу того, что алгоритм стартует из различных начальных точек, то он может сходится к различным решениям, к разным локальным минимумам.

Задачи, решение которых не единственно или неустойчиво, называются некорректно поставленными. Особенностью подобных задач является неустойчивость их решения: малые изменения исходных данных могут вызвать произвольно большие изменения решений. Общий подход к решению некорректно поставленных задач называется регуляризацией.

Суть регуляризации заключается либо в до определении априорной информации либо в сужении класса функций. Применении дополнительной априорной информации играет ключевую роль в теории регуляризации, чем большей априорной информацией мы обладаем, тем более устойчивые алгоритмы могут быть использованы при решении некорректно поставленной задачи. Применение процедуры регуляризации к тематическому моделированию заключается в разумном введении ограничений на матрицы.

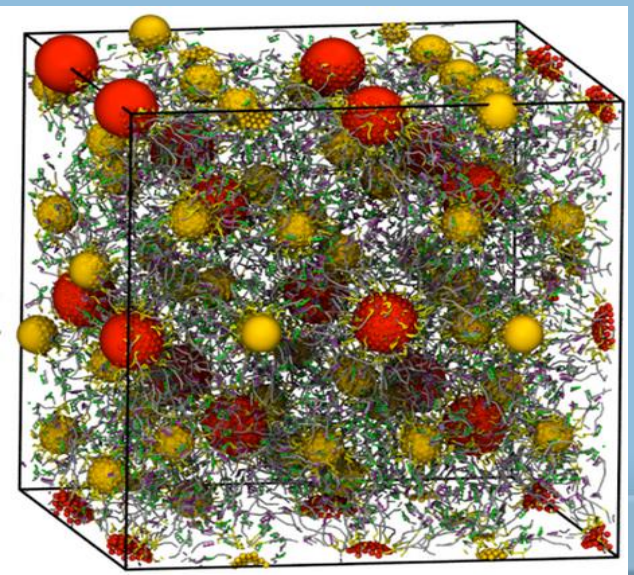
Semi-Supervised Latent Dirichlet Allocation (Gibbs sampling)

Next level of regularization is based on the following idea. If we have initial distribution of words (anchor words) over topics, then we are able to fix or glue words to topics. Therefore, when the algorithm faces an anchor word during sampling, it does not change the connection between the topic and the word. But the other words are sampled according to the standard procedure.

$$p(\mathrm{z}, \mathrm{w}, \alpha, \beta) \propto \begin{cases} z = t \\ q(\mathrm{z}, \mathrm{w}, \alpha, \beta) \end{cases}$$

Initial anchor words distribution

Standard Gibbs sampling



The SLDA modeling behaves as a process of crystallization, where anchor words are centers of crystals. The words that often co-occur with anchor words stick together during simulation and form the body of topics.
Therefore the fixed lists of anchor words assigned to topics lead to stabilization of topic modeling.

But SLDA works good if the list of words is known beforehand which is not always the case in social science.

GRANULATED LDA (based on Gibbs sampling)

Granulated LDA based on idea that each document from a collection can be regarded as a granulated surface – a notion we borrow from physics. Here a granule is a set of words located near each other. Therefore we can arrange sampling by granules.

All words within one granule belong to one topic. Therefore scanning documents and assigning granules to topics, the algorithm favors the words located near each other.

DOCUMENT

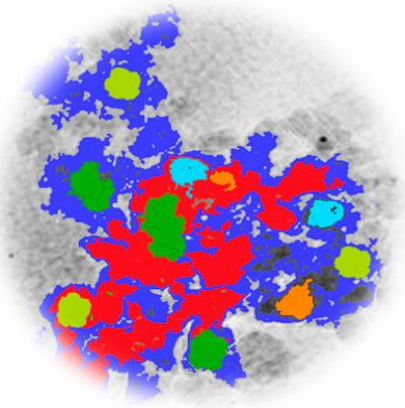
The central theme of ethnic nationalism is that «nations are defined by a shared heritage, which usually includes a common language, a common faith, and a common ethnic ancestry».[2] It also includes ideas of a culture shared between members of the group, and with their ancestors, and usually a shared language; however it is different from purely cultural definitions of «the nation» (which allow people to become members of a nation by cultural accimilation) and a purely linguistic definitions (which see «the nation» as all speakers of a specific language). Herodotus is the first who stated the main characteristic of ethnicity, with his famous account of what defines Greek identity, where he lists kinship language, cults and customs.

The central political tenet of ethnic nationalism is that ethnic groups can be identified unambiguously, and that each such group is entitled to self-determination.

The outcome of this right to self-determination may vary, from calls for self-regulated administrative bodies within an already-established society, to an autonomous entity separate from that society, to a sovereign state removed from that society. In international relations, it also leads to policies and movements for irredentism to claim a common nation based upon ethnicity.

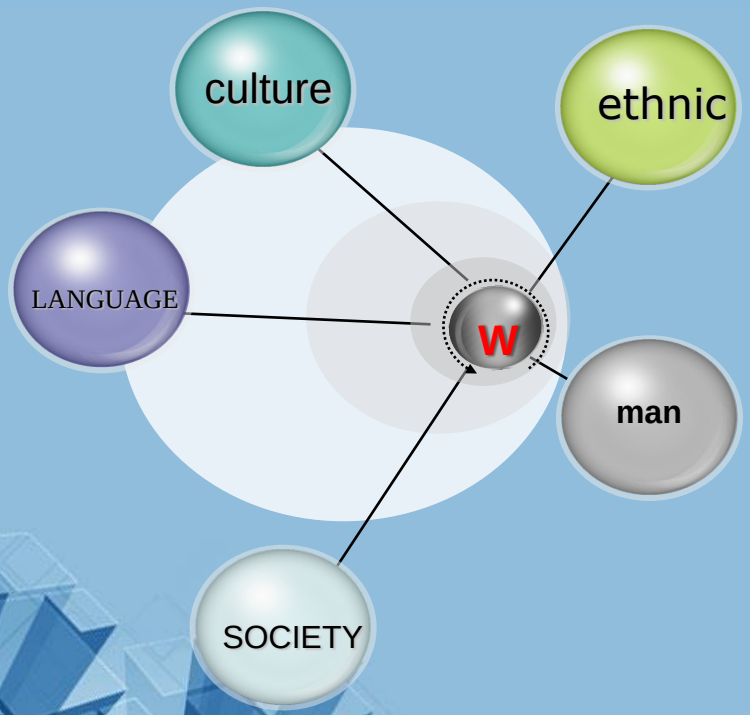
KEYWORDS

ethnic	7
common language	3
culture	2
self-determination	2
society	3



GRANULATED LDA: Algorithm

Entrance: collection of documents D, number of topics |T|,
number of iterations, size of granules L;
Initialization: φ(w,t), θ(t,d) for all documents and topics d ∈ D, w ∈ W , t ∈ T;



Run external cycle along all documents (i)
Run internal cycle. Length of cycle is number words in documents i.
1. Generation of random number k. Max value of k is number of words in documents i.
2. Choosing word **k** from document **i**.
3. Calculating topic number **t** for word **k**.
4. Defining words that are around word k in document i.
5. Assigning topic **t** to all words which are within granule L.
End of internal cycle.
Updating the following matrices:
End of external cycle

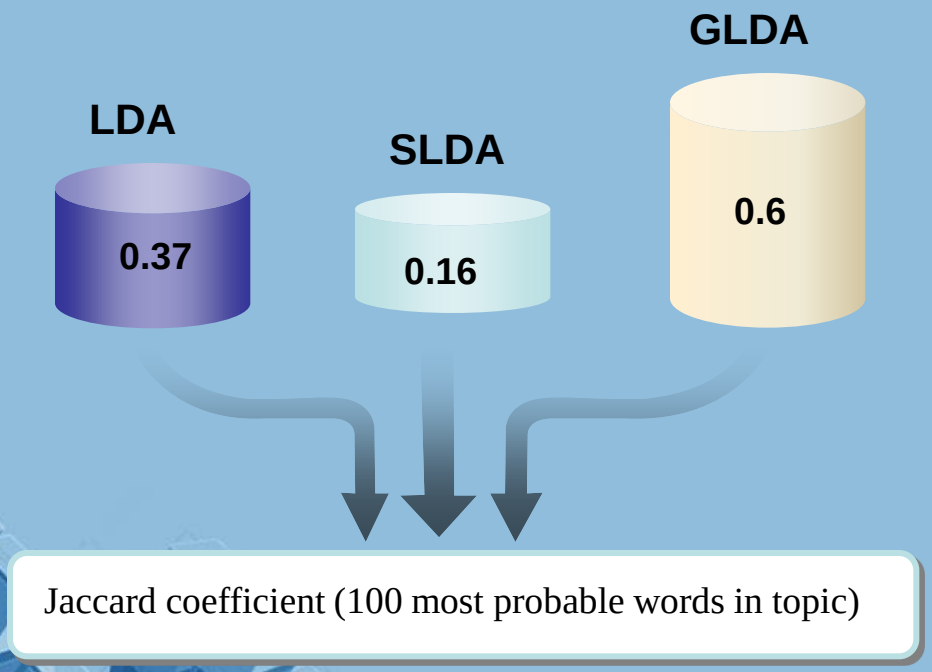
$$\theta_{dj} = \frac{C_{d,j}^{DT} + \alpha}{C_{d,j'}^{DT} + T\alpha}$$

$$\phi_{m,j} = \frac{C_{m,j}^{WT} + \beta}{\sum_{m'} C_{m',j}^{WT} + V\beta}$$

Results of simulations

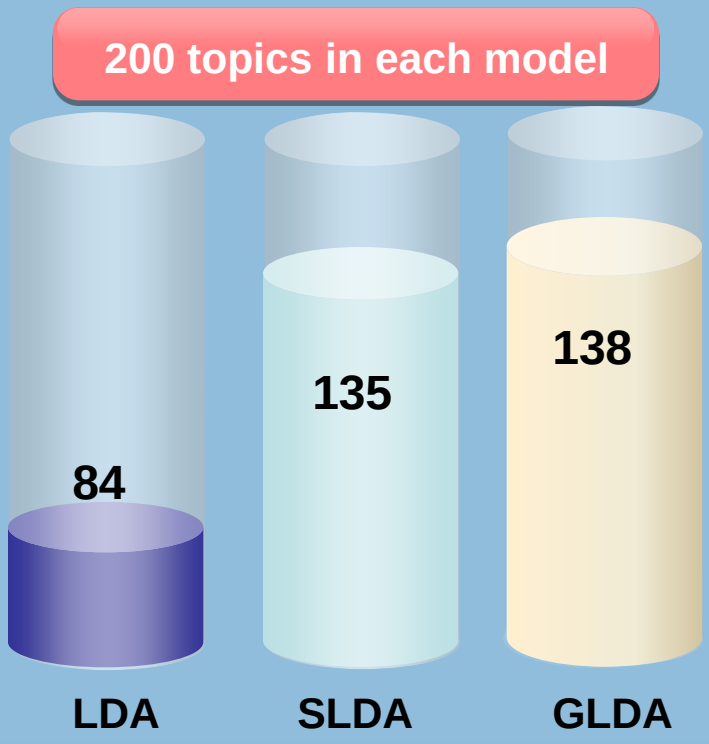
Dataset

101,481 posts from the Russian LiveJournal.
172,939,000 tokens (unique words).



Number of runs: 5 times

Number of stable topics
According to Kullback-Leibler



Возможные направления

1. Многомодальные модели. В данных моделях мешок слов дополняется словами из метаданных. Например, дополнить список слов списком регионов, городов и более мелких населенных пунктов. Основной список слов – одна модальность, список населенных пунктов и регионов – вторая модальность, образование пользователей – третья модальность, время поста – четверая модальность и так далее. Затем проводить сэмплирование по такому расширенному списке. В итоге получим, что в теме будут присутствовать слова из разных модальностей. Соответственно, производя сортировку по весу слов из разных модальностей можно получать различные типы решений.

2. Отказ от функций Дирихле. В качестве функций распределений можно например использовать другие функции, например функции Леви.

3. Отказ от модели марковской цепи и замене ее моделью с памятью. Тут можно варьировать глубину памяти. При глубине памяти =1, получаем обычную марковскую цепь.