**ВМК**

# Задание 2. Приложения EM-алгоритма

**Практикум 317 группы, 2018**

Начало выполнения задания: 19 марта 2018 года.
Мягкий дедлайн: **2 апреля 2018 года, 23:59.**
Жёсткий дедлайн: **16 апреля 2018 года, 23:59.**
Максимальный балл: 60 (плюс бонусные баллы).
Текст задания обновлён 19 марта 2018 г.

## Формулировка задания

Данное задание направлено на ознакомление с EM-алгоритмом. Задание состоит из двух частей, каждая оценивается в 30 баллов:

1. В первой части необходимо написать на языке Python собственную реализацию EM-алгоритма для восстановления смеси нормальных распределений. Во второй части необходимо написать на языке Python собственную реализацию модели PLSA с возможностью добавления регуляризации.

   Прототипы функций должны строго соответствовать прототипам, описанным в спецификации, и проходить все выданные тесты. Задание, не проходящее все выданные тесты, приравнивается к невыполненному. При написании необходимо пользоваться стандартными средствами языка Python, библиотеками numpy, scipy, scikit-learn и matplotlib, но запрещается пользоваться уже готовыми реализациями выше упомянутых методов.

2. Провести описанные ниже эксперименты.

3. Написать отчёт о проделанной работе (формат PDF, подготовленный в системе LaTeX, или формат html, конвертированный из jupyter-notebook).

4. Сдать в систему anytask .zip архив с отчётом и jupyter-notebook с экспериментами (если есть дополнительные модули с кодом, их тоже нужно поместить в архив). Архив должен иметь название `task2_фамилия_имя.zip`

5. Допускается сдача только одной из двух частей задания

## Часть 1: Вычитание фона

### Введение

Вычитание фона — важная прикладная задача. Часто оно является первой стадией в системах анализа видео с камер наблюдения. Необходимо классифицировать пиксели видеопоследовательности на принадлежащие фону и принадлежащие объектам переднего плана. Обычно предполагается, что камера статична. По обучающей выборке необходимо оценить модель фона, чтобы затем для каждого пикселя каждого кадра тестовой видеопоследовательности уметь предсказать вероятность того, что он является фоном. При предположении равномерного распределения принадлежности пикселей объектам переднего плана (обычно нет априорных сведений, чтобы использовать другое распределение) принятие решения для данного пикселя эквивалентно сравнению предсказанной вероятности фона с некоторым порогом.

Часто (хотя и не всегда) в качестве обучающей выборки удаётся взять один или несколько кадров, на которых отсутствуют объекты переднего плана. Но даже в этом случае задача оценки модели фона не является тривиальной из-за того, что фон никогда не бывает полностью статичен: всегда есть шум камеры, также может меняться освещение, тени, камера может дрожать, фон может быть динамическим (листва или вода в присутствии ветра), к фону могут добавляться дополнительные объекты. Вам предлагается протестировать несколько моделей разной сложности и сделать выводы о применимости моделей в разных ситуациях.

### Данные

В качестве тестовых последовательностей предлагается использовать стандартные данные с веб-сайта http://wordpress-jodoin.dmi.usherb.ca/dataset2012/. Один набор данных является обязательным:

- Baseline/pedestrians (train — $1, \ldots, 299$; test — $300, \ldots, 1099$),

## Оценка алгоритмов

Для каждого кадра тестовой части последовательности можно оценить качество алгоритма, зная его верную разметку (директория groundtruth). Зная, какие части изображения на самом деле относятся к фону (класс static) и к переднему плану (класс motion), можно оценить количество верных положительных (TP) и отрицательных (TN) обнаружений пикселей, а также ложноположительных (FP) и ложноотрицательных (FN).

Для оценки качества вычитания фона следует использовать следующие три инструмента:

- Для каждого тестового кадра посчитайте количество ошибок 1 рода (FP) и 2 рода (FN). Постройте график зависимости ошибок от номера кадра. Опишите наблюдаемые эффекты и попытайтесь проанализировать их.
- Проведите визуальную оценку. Для этого на каждом из кадров выделите области, отнесённые к переднему плану. Сделайте вывод о характере ошибок.
- Проведите оценку величин TPR и FPR (доли верноположительных и ложноположительных обнаружений, соответственно) для разных значений порога по всей тестовой последовательности (не усредняя по кадрам) и постройте графики ROC характеристики https://en.wikipedia.org/wiki/Receiver_operating_characteristic. Площадь под графиком ROC кривой (AUC ROC) является одним из способ сравнить качество разных моделей.

Для удобства к заданию прилагается python код, который содержит функцию подсветки маски переднего плана, а так же функцию, которая позволяет проигрывать видео. Данный код может быть полезным для визуальной оценки качества работы методов. Для работы функции проигрывания видео в IPython Notebook необходимо установить `JSAnimation` из репозитория https://github.com/jakevdp/JSAnimation и импортировать следующий модуль – `from JSAnimation import IPython_display`. Подробнее информацию про установку можно посмотреть здесь: https://gist.github.com/gforsyth/188c32b6efe834337d8a

## Одномерная гауссиана для оценки фона

В самом простом варианте можно моделировать распределение цвета в каждом пикселе одномерным нормальным распределением. Для этого нужно перевести видеопоследовательность в полутоновую. Это можно сделать, например, с помощью формулы из стандарта `NTSC`: $gs = 0.2126r + 0.7152g + 0.0722b$. Для каждого пикселя обучающей части последовательности необходимо оценить параметры нормального распределения, используя яркости в каждой позиции пикселя (например, для **pedestrians** будет 299 точек для оценки плотности в каждом пикселе). Обратите внимание, что некоторые гауссианы получаются вырожденными. Для регуляризации разумно ограничивать возможное значение среднеквадратичного отклонения снизу. Можно использовать $\sigma_{\min} = 5$. Далее, при классификации пикселя тестовой выборки, его нужно относить к фону тогда и только тогда, когда его яркость отклоняется от $\mu$ меньше, чем на $k\sigma$. Если установить $k = 3$, ожидается, что к переднему плану будут отнесены только 0.27% пикселей фона ("правило $3\sigma$"). Изменяя этот порог, можно регулировать соотношение между количеством ложноположительных и ложноортицательных обнаружений.

## Адаптивная гауссиана для оценки фона

Если в последовательности присутствует дрейф фона, желательно оценивать его модель по нескольким последним кадрам. Проблема заключается в том, что строить новую модель для каждого кадра может быть вычислительно затратно, а на предшествующих кадрах фон может быть загорожен объектами переднего плана. Чтобы этого избежать, можно оценивать модель в онлайн-режиме («на лету»). Итеративный алгоритм оценки параметров гауссовской модели фона был рассмотрен на 4-ой лекции курса.

## Усложнение модели фона

Предложенную модель фона можно усложнить, используя трёхмерное нормальное распределение в цветовом пространстве RGB. Решение о метке пикселя тестового кадра принимается сравнением плотности нормального распределения в данной точке с порогом. Для оценки качества модели используйте ROC кривую.

Ещё более сложная модель — смесь распределений — может быть полезна для моделирования фона, если он нестатичен, например, содержит воду или листву (то есть, когда распределение может иметь несколько мод).

## Требуемые эксперименты

1. Реализовать EM-алгоритм для восстановления смеси многомерных нормальных распределений согласно заданному прототипу. Требования по эффективности реализации: среднее время одной EM-итерации для $N = 10000$ объектов, $D = 100$ признаков и $K = 10$ компонент смеси не должно превышать одной секунды.

2. Провести тестирование реализованного EM-алгоритма на двумерных модельных данных. Для этого сгенерировать данные из смеси распределений с заданными параметрами, а затем восстановить по этим данным параметры смеси с помощью EM-алгоритма. Отобразить результат восстановления, где объекты выборки, соответствующие одинаковым компонентам смеси, показаны одинаковыми цветами. Убедиться в том, что значение правдоподобия в EM-итерациях монотонно не убывает.

3. Реализовать оценку модели фона с помощью одномерной гауссианы и протестировать на последовательности **pedestrians**. Проанализировать качество вычитания фона с помощью трёх инструментов, описанных выше.

4. Реализовать оценку модели фона с помощью многомерной гауссианы в цветовом пространстве RGB и протестировать результат на последовательности **pedestrians**. Проанализировать ошибки метода и сравнить его с остальными.

5. Запустить алгоритм разделения смеси 3 трёхмерных гауссиан для вычитания фона в последовательности **traffic**. Проанализировать ошибки метода и сравнить результат с использованием одной гауссианы.

   **Замечание.** Поскольку разделение смеси гауссиан на реальных данных вычислительно затратно, предлагается сначала отладить алгоритм на синтетических данных. Сгенерируйте выборку из смеси гауссиан и попробуйте восстановить её параметры с помощью EM-алгоритма. Постройте график изменения логарифма правдоподобия, убедитесь в его монотонном росте.

6. Наилучшее из полученных решений для алгоритма вычитания фона должно быть добавлено в отчёт (или приложено отдельным файлом) в виде анимации. Формы анимации могут быть различными: виджет в IPython notebook `JSAnimation`, отдельный файл с видео, анимированные изображения кадров и т.д.

**Бонусная часть:**

1. (до +5 баллов) Реализовать оценку модели фона на основе адаптивной одномерной гауссианы. Протестировать метод на последовательности **pedestrians** и сравнить результаты с простой одномерной гауссианой.

2. (до +5 баллов) Реализовать оценку модели фона трёхмерной гауссианой и смесью гауссиан в цветовом пространстве HSV, в котором каналы коррелируют меньше. Для перевода изображения можно воспользоваться встроенной функцией `matplotlib.colors.rgb_to_hsv`.

3. (до +5 баллов) Реализовать EM-алгоритм для восстановления параметров смеси из нормальных распределений с диагональными матрицами ковариаций. Оценить, как упрощение модели влияет на её качество и скорость работы.

# Часть 2: Тематическое моделирование в мультиязычном поиске

## Введение

Тематическое моделирование — это одно из направлений статистического анализа текстов. *Вероятностная тематическая модель* (probabilistic topic model) выявляет тематику коллекции документов, представляя каждую тему дискретным распределением вероятностей слов, а каждый документ — дискретным распределением вероятностей тем.

Тематическое моделирование похоже на кластеризацию документов. Отличие в том, что при кластеризации документ целиком относится к одному кластеру, тогда как тематическая модель осуществляет «мягкую кластеризацию» (soft clustering), разделяя документ между несколькими кластерам-темами. Тематические модели называют также моделями мягкой би-кластеризации, поскольку каждое слово также распределяется по нескольким кластерам-темам.

Тема образуется семантически связанными, часто совместно встречающимися словами, словосочетаниями или терминами. Такое определение «темы» допускает точную математическую формализацию, но может отличаться от принятых в лингвистике или литературоведении.

Основной целью тематического моделирования является понимание большой текстовой коллекции, систематизация контента, разложение каждого документа на «элементарные смыслы». Одним из приложений тематического моделирования является мультиязычный информационный поиск, поиск близких по тематике докуметов на разных языках. В этом задании вам предлагается решить модельную задачу мультиязычного поиска на небольшой коллекции.

## Данные

Вам предлагается использовать коллекцию параллельных текстов (для каждого документа известен его аналог на другом языке) на русском и английском языках небольшой новостной коллекции. Дополнительно вам выданы словари с переводами самых частотных слов двух языков. Данные необходимо разделить на обучающую и тестовую выборки в пропорции 9:1.

## Оценка алгоритма

Для оценивания качества построенной модели и контроля сходимости процесса обучения можно использовать оценку правдоподобия, но в тематическом моделировании (и компьютерной лингвистике) обычно используют перплексию ($N$ — общее число словопозиций в коллекции):

$$\mathcal{P} = \exp\Big(-\frac{\mathcal{L}}{N}\Big) = \exp\Big(-\frac{1}{N}\sum_{d\in D}\sum_{w\in d} n_{dw}\ln\big(\sum_{t\in T}\phi_{wt}\theta_{td}\big)\Big)$$

Число итераций $max\_iter$ в алгоритме обучения следует выбирать достаточным для того, чтобы перплексия перестала существенно убывать. Однако известно, что перплексия плохо отражает интерпретируемость найденных тем, поэтому помимо нее обычно используются дополнительные меры или экспертные оценки.

В данном задании дополнительным критерием качества является качество поиска перевода документа по оригиналу и по обучающей, и по тестовой выборке. Для оценивания качества используйте метрику Average Precision at n, со значениями 1, 3, 5, 10. Для документа $d$, такого что $\rho(d,d_1) \leqslant \rho(d,d_2)\dots\rho(d,d_{|D|})$, а $y_i = 1$, если $d_i$ — перевод $d$ и 0 иначе, метрика вычисляется следующим образом:

$$AP@n = \sum_{k=1}^{n}[y_k = 1]\frac{\sum_{i=1}^{k} y_i}{k}$$

## Тематическая модель PLSA

---

**ЕМ-алгоритм для регуляризованной тематической модели**

**Input:** коллекция $D$, число тем $|T|$, число итераций $i_{\max}$;
**Output:** матрицы терминов тем $\Theta$ и тем документов $\Phi$;

инициализация $\phi_{wt}$, $\theta_{td}$ для всех $d \in D,\ w \in W,\ t \in T$
**forall** *итераций* $i = 1,\dots,i_{\max}$ **do**
  $n_{wt}, n_{td} := 0$ для всех $d \in D,\ w \in W,\ t \in T$
  **forall** *документов* $d \in D$ **do**
    **forall** *слов* $w \in d$ **do**
      $n_{tdw} := n_{dw}\underset{t\in T}{\operatorname{norm}}\big(\phi_{wt}\theta_{td}\big)$ для всех $t \in T$
      $n_{wt}$ += $n_{tdw}$;
      $n_{td}$ += $n_{tdw}$ для всех $t \in T$
    $\theta_{td} := \underset{t\in T}{\operatorname{norm}}\Big(n_{td} + \theta_{td}\frac{\partial R}{\partial\theta_{td}}\Big)$ для всех $t \in T$
  $\phi_{wt} := \underset{w\in W}{\operatorname{norm}}\Big(n_{wt} + \phi_{wt}\frac{\partial R}{\partial\phi_{wt}}\Big)$ для всех $w \in W,\ t \in T$

---

Пусть $D$ — множество (коллекция) текстовых документов, $W$ — множество (словарь) всех употребляемых в них термов. Каждый документ $d \in D$ представляет собой последовательность $n_d$ термов $w_1,\dots,w_{n_d}$ из словаря $W$.

Каждое вхождение терма $w$ в документ $d$ связано с некоторой темой $t$ из заданного конечного множества $T$. Коллекция документов представляет собой последовательность троек $\Omega_n = \big\{(w_i,d_i,t_i)\ \big|\ i = 1,\dots,n\big\}$. Термы $w_i$ и документы $d_i$ являются наблюдаемыми переменными, темы $t_i$ не известны и являются *латентными* (скрытыми) переменными. Порядок термов в документах не важен для выявления тематики, то есть тематику документа можно узнать даже после произвольной перестановки термов, хотя для человека такой текст потеряет смысл. Порядок документов в коллекции также не имеет значения — это предположение называют гипотезой «мешка документов».

Появление слов в документе $d$ по теме $t$ зависит от темы, но не зависит от документа $d$, и описывается общим для всех документов распределением $p(w|t)$:

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

Согласно формуле полной вероятности и гипотезе условной независимости, распределение термов в документе $p(w\,|\,d)$ описывается *вероятностной смесью* распределений термов в темах $\phi_{wt} = p(w\,|\,t)$ с весами $\theta_{td} = p(t\,|\,d)$:

$$p(w\,|\,d) = \sum_{t\in T} p(w\,|\,t,d)\,p(t\,|\,d) = \sum_{t\in T} p(w\,|\,t)\,p(t\,|\,d) = \sum_{t\in T} \phi_{wt}\theta_{td}.$$

Восстановить параметры модели можно с помощью EM-алгоритма. Построенная модель позволяет получать оценки для тестовых документов без полного переобучения модели. Необходимо зафиксировать матрицу $\Phi$ и несколько раз запустить E-шаг модели, изменяя только матрицу $\Theta$ для новых значений.

## Регуляризованная тематическая модель

Построение тематической модели по коллекции документов является некорректно поставленной оптимизационной задачей, которая может иметь бесконечное множество решений. Согласно теории регуляризации А. Н. Тихонова, решение такой задачи возможно доопределить и сделать устойчивым, добавив к основному критерию *регуляризатор* — дополнительный критерий, учитывающий какие-либо специфические особенности прикладной задачи или знания предметной области.

В экспериментах ниже вам предлагается построить мультиязычную модель и использовать специальные регуляризаторы для мультиязычных данных. Мультиязычные данные в общем случае можно разделить на два вида: данные словарей о переводах слов и корпуса параллельных текстов. В каждом из случаев можно придумать соответствующий регуляризатор. В случае слова $w$ и его перевода $u$, регуляризация направлена на сближение $p(t|w)$ одного языка и частотной оценки $n_{tu}/n_u$ для другого языка. В случае параллельных текстов $d$ и $s$, регуляризация направлена на сближение $p(t|d)$ одного языка и частотной оценки $n_{ts}/n_s$ для другого языка.

## Мультимодальная тематическая модель

В документе могут присутствовать разные метаданные: время написания документа, авторы документа, ссылки на другие документы. В теории тематического моделирования метаданные часто называют модальностями. В частности, можно считать метаданными документа его переводы на другой язык. В этом случае каждому документу, для которого известны аналоги на других языках, будет соответствовать несколько модальностей, каждая из которых связана с определённым языком. В этом случае, для каждого такого документа представление $\theta_{td}$ будет единым на всех языках, дополнительная регуляризация по параллельным текстам будет не нужна.

## Требуемые эксперименты

1. Реализовать EM-алгоритм для восстановления параметров тематической модели PLSA с возможностью добавления регуляризации. Помните, что можно избежать хранения трёхмерного массива счётчиков при правильной организации процесса обучения.

2. Провести тестирование реализованного EM-алгоритма на сгенерированных данных. Для этого необходимо сгенерировать столбцы $\Phi$, $\Theta$ (например, из нормированного Гамма распределения) и по модели порождения коллекции сгенерировать коллекцию. Протестировать алгоритм на сгенерированных данных и убедиться, что правдоподобие не убывает на каждом шаге алгоритма.

3. Считать обучающую выборку, сохранить датасет в памяти в формате `scipy.sparse.csr_matrix`.

   **Замечание.** Для текстов проведена полная предобработка (лемматизация, выброшены стоп-слова, удалены лишние символы). Можно удалить из выборки слова, которые встречаются меньше чем в 5 документах, для снижения времени работы.

4. Построить модель на русскоязычной или англоязычной части коллекции (около 50 тем), для каждой темы вывести список её топ-слов и топ-документов. Визуально оценить интерпретируемость полученных тем.

5. Построить двуязычную регуляризованную модель с регуляризатором на основе данных о переводах слов. Профильтруйте приложенные словари, чтобы в них остались только слова, находящиеся в выборке. Подберите коэффициент регуляризации, основываясь на правдоподобии и метриках AP@n. Визуально оцените качество полученных тем. Сделайте необходимые выводы о применимости такого подхода.

6. Построить регуляризованную модель с регуляризатором на основе данных о параллельных документах. Используйте данные только из обучающей выборки. Подберите коэффициент регуляризации, основываясь на правдоподобии и метриках AP@n. Визуально оцените качество полученных тем. Сделайте необходимые выводы о применимости такого подхода. Попробуйте применить одновременно оба регуляризатора.

**Бонусная часть:**

1. (до +5 баллов) Построить мультимодальую модель с двумя модальностями: английский и русский язык. Оцените значения метрик AP@n по сравнению с предыдущими подходами. Визуально оцените качество полученных тем. Сделайте необходимые выводы о применимости такого подхода.
2. (до +5 баллов) Визуализируйте модель `gensim` с помощью библиотеки `ldavis`, ссылка
3. (до +5 баллов) Придумайте/найдите интересную визуализацию модели. Оценивается нетривиальность, красота и полезность визуализации.