Министерство образования и науки Российской Федерации
Московский физико-технический институт (государственный университет)
Факультет управления и прикладной математики
Кафедра «Интеллектуальные системы»

Воронов Сергей Олегович

# Фильтрация и тематическое моделирование коллекции научных документов

010900 — Прикладные математика и физика

Выпускная квалификационная работа бакалавра

**Научный руководитель:**
д.ф.-м.н., проф. МФТИ
Воронцов Константин Вячеславович

Долгопрудный
2014

# Содержание

**Аннотация**

В данной работе исследуется проблема жанровой классификации на два жанра: научные и ненаучные документы. Построена система признаков документов, позволяющая добиться ошибки классификации около 4%.. Проведено сравнение различных методов настройки весов линейной модели классификации, производится поиск оптимальных параметров методов. Рассматриваемые методы тестируются на коллекции текстовых документов. Предлагается метод автоматического порождения новых признаков при помощи тематической модели классификации с использованием регуляризаторов. Сделан вывод о качестве работы исследуемых классификаторов.

**Ключевые слова:** *жанровая классификация, тематическое моделирование, тематическая модель классификации, аддитивная регуляризация, автоматическое порождение признаков.*

# 1 Введение

Цель работы — создание системы фильтрации научного контента. Для этого необходимо решить задачу классификации контента (текстовых документов) на научные и ненаучные, что является частью более общей задачи жанровой классификации документов, построив состоятельную систему признаков документов.

## 1.1 Задача жанровой классификации документов

Задача жанровой классификации документов состоит в определении принадлежности документа к одному или нескольким классам из заданного множества классов (жанров). Классификацию документов можно проводить на основе некоторой метаинформации, поступающей с ними (например, источник документа, авторы, заголовок и т.д.). Однако, данная информация не всегда доступна, например, если документы получены путем обхода веб-графа. В таких случаях производится классификация документов на основе их внутреннего содержания.

В данной работе исследуется проблема классификации на два жанра: научные и ненаучные документы. Понятие научного контента часто используется в реальной жизни, но формализовать определение сложно. Поэтому, научным документом мы будем называть то, что эксперт при разметке обучающей выборки отметил как научный контент.

Как было указано в [14], жанр описывает что-то о документе, а не то, о чем написано в документе. Иными словами, документы, которые описывают одну проблему могут быть разных жанров. Например, о многих задачах естественных наук можно написать научную статью, либо в новостном формате описать некоторые результаты. Соответственно, классификация по жанрам является довольно сложной задачей. При этом жанр научных документов обычно выделяется с довольно высокой точностью.

Для классификации документов часто используется SVM [5, 7], RVM [9] или какие-либо другие линейные классификаторы. Для создания признаков используются различные методы анализа структуры текста (частота встречаемости слов, специфичные слова) [14, 12] и дополнительная метаиформация [14].

## 1.2 Классификация при помощи тематической модели

Тематическая модель коллекции текстовых документов определяет, к каким темам относится каждый документ и какие слова (термины) образуют каждую тему. Вероятностная тематическая модель описывает каждую тему дискретным распределением на множестве терминов, каждый документ – дискретным распределением на множестве тем. Предполагается, что коллекция документов — это последовательность терминов, выбранных случайно и независимо из смеси таких распределений, и ставится задача восстановления компонент смеси по выборке.

Задача заключается в том, чтобы выявить связи между классами и темами, улучшить качество тематической модели и построить алгоритм классификации новых документов, для которых метки ещё не проставлены. Сложность задачи в том, что стандартные алгоритмы классификации показывают неудовлетворительные результаты на больших текстовых коллекциях с большим числом несбалансированных, пересекающихся, взаимозависимых классов [11].

В случае классификации текста тематическая модель не задает однозначно метку класса, а лишь определяет вероятности принадлежности документа к каждому из классов. Этим данный подход выгодно отличается от т.н. методов «жесткой классификации», когда документ однозначно приписывается к одному из классов. К примеру, решается проблема омонимов: такие термины распределяются между несколькими темами. Синонимы же с высокой верояностью попадают в одну тему.

Большинство тематических моделей основаны на методе латентного размещения Дирихле LDA [2]. Примеры моделей: MedLDA [13], Labeled LDA [10], Flat-LDA [11].

В данной работе тематическая модель из [11] используется для полуавтоматической генерации словарных признаков. Здесь под словарным подразумевается такой признак, значение которого для данного документа зависит только от количеств появлений слов из заданного словаря в документе.

В качестве основных классификаторов выбраны SVM и логистическая регрессия [1].

# 2 Постановка задачи

Дана коллекция текстовых документов $D$, где $|D|$ — велико. Необходимо выделить среди них те, которые являются научными. Данная задача интерпретируется

как задача классификации на два класса: научные и ненаучные. Для её решения нужно разработать обучаемый алгоритм распознавания научных документов. Обозначим через $C = \{-1, +1\}$ множество всех классов документов. Класс $-1$ интерпретируется как ненаучный документ, а $+1$ — как научный. Тогда задача классификации заключается в построении такой функции $a : D \to C$, которая минимизирует заданную функцию ошибки. В работе классификаторы сравнивались по значению AUC.

Для проверки гипотез использовалась выборка, полученная с помощью выбора страниц с 2000 сайтов вузов. Она характеризуется несбалансированным распределением классов: число научных документов не превосходит 2% всех файлов. Кроме того, велико общее число документов ($\sim 850000$).

Для использования таких методов линейной классификации, как SVM и логистическая регрессия нужно иметь представительную размеченную выборку и систему признаков документов.

# 3 Теоретическая часть

## 3.1 Линейная модель классификации

Дан набор текстовых документов $D$. Предположим, что для каждого документа $d$ задан вектор числовых признаков $f(d) = (f_1(d), ..., f_n(d))$. Необходимо классифицировать все документы на два класса: $+1$ и $-1$.

В данной модели классификатор $a(d, w)$ представляется как

$$a(d, w) = \text{sign}\left(\sum_{j=1}^{n} w_j f_j(d) - w_0\right) = \text{sign}(\langle d, w\rangle - w_0),$$

где $d$ — классифицируемый документ, $w_0$ — порог принятия решения, $w = (w_1, ..., w_n)$ — вектор весов классификатора. Напомним, что отступом документа $d_i$ называют величину $M_i(w) = y_i \cdot (\langle d, w\rangle - w_0)$. Документ $d_i$ классифицируется правильно тогда и только тогда когда $M_i(w) > 0$. Линейные методы классификации различаются способом решения задачи минимизации суммарной ошибки:

$$Q(w) = \sum_{i=1}^{n} [M_i(w) < 0] \to \min_w .$$

Вместо решения задачи минимизации с кусочно-постоянным функционалом, индикатор неверной классификации $[M_i(w) < 0]$ заменяется на некоторую непрерывную

и невозрастающую функцию $L(M_i(w))$

$$\forall w : L(M_i(w)) \geqslant [M_i(w) < 0],$$

аппроксимирующую этот индикатор.

Соответственно, решаемая задача заменяется на другую:

$$Q(w) = \sum_{i=1}^{n} [M_i(w) < 0] \leqslant \sum_{i=1}^{n} L(M_i(w)) \to \min_w .$$

### 3.1.1 Метод опорных векторов

Метод опорных векторов (SVM) настраивает веса, решая следующую оптимизационную задачу:

$$Q(w) = \sum_{i=1}^{m} (1 - M_i(w))_+ + \frac{1}{2C}|w|^2 \to \min_w,$$

где $C$ — параметр метода, отвечающий за баланс между минимизацией суммарной ошибки (первое слагаемое) и максимизацией разделяющей полосы (второе слагаемое). Эта задача получается из принципа минимизации суммарной ошибки, если взять $L = (1 - M_i)_+$ и добавить регуляризатор $|w|^2$ с параметром $\dfrac{1}{2C}$.

Так как обычно выборка не является линейно разделимой, то возможен переход от пространства признаков к другому пространству с помощью некоторого преобразования. По существу данный переход приводит к замене скалярного произведения $\langle d, d' \rangle$ на некоторую функцию $K(d, d') = \langle g(d), g(d') \rangle$, называемую ядром, которая является скалярным произведением в некотором пространстве $H$, в которое документы переводятся преобразованием $g$.

### 3.1.2 Логистическая регрессия

Регуляризованный логистический классификатор (RLR) настраивает веса, решая следующую оптимизационную задачу:

$$Q(w) = \sum_{i=1}^{m} \ln\left(1 + \exp(-y_i \langle x_i, w \rangle)\right) + \frac{\lambda}{2}|w|^2 \to \min_w$$

где $\lambda$, — параметр метода. Эта задача получается, если взять $L = \log_2(1 + e^{-M_i})$ и добавить регуляризатор $|w|^2$ с коэффициентом регуляризации $\dfrac{\lambda}{2}$.

Для решения данной задачи применяют градиентные методы оптимизации:

- Градиентный спуск

$$w = w - \alpha \left( \frac{\partial Q(w)}{\partial w} \right),$$

где $\alpha$ — градиентный шаг. Преимущество метода заключается в скорости работы. Однако, он может медленно сходиться и/или застревать в локальных минимумах.

- Метод Ньютона

$$w = w - \alpha \left( \frac{\partial^2 Q(w)}{\partial w \partial w^T} \right)^{-1} \frac{\partial Q(w)}{\partial w}.$$

Его преимущество заключается в более высокой скорости сходимости. Однако, решение требует обращения матрицы, которая в данной задаче часто была близка к вырожденной. Поэтому получилось, что метод неприменим.

- Метод Левенберга-Марквардта

$$w_i = w_i - \alpha \left( \mu + \frac{\partial^2 Q(w)}{\partial w_i \partial w_i} \right)^{-1} \frac{\partial Q(w)}{\partial w_i}.$$

Является упрощением метода Ньютона (используются только диагональные элементы матрицы Гессе). Параметр $\mu$ отвечает за скорость сходимости при нулевом элементе матрицы. Применяется для выпуклых функций.

## 3.2 Тематическая модель классификации

### 3.2.1 Тематическая модель

Обозначим через $W$ множество (словарь) всех слов, встречающихся в документах. Предположим, что для каждого документа $d$ из коллекции $D$ и для для всех слов $w$ из словаря $W$ заданы $n_{dw}$ — количество появлений слова $w$ в документе $d$. Обозначим $N = ||n_{dw}||$.

Предполагается, что существует конечное множество тем $T$, и каждое употребление термина $w$ в каждом документе $d$ связано с некоторой темой $t \in T$. Будем полагать, что появление слов в документе $d$, относящихся к теме $t$, описывается общим для всей коллекции распределением $p(w|t)$ и не зависит от документа $d$.

Для решения задачи тематического моделирования требуется предствить $N$ в виде приближенного произведения $N \approx \Phi\Theta$, где $\Phi = ||\varphi_{wt}||_{W \times T}$ — матрица распределения слов по темам, а $\Theta = ||\theta_{td}||_{T \times D}$ — матрица распределения тем по документам.

Для это решается задача максимизации логарифма правдоподобия при ограничениях неотрицательности и нормированности столбцов матриц $\Phi$ и $\Theta$:

$$\begin{cases} L(\Phi,\Theta) = \sum\limits_{d\in D}\sum\limits_{w\in d} n_{dw}\ln\sum\limits_{t\in T}\varphi_{wt}\theta_{td} \to \max\limits_{\Phi,\Theta}, \\ \sum\limits_{w\in W}\varphi_{wt} = 1;\ \varphi_{wt} \geqslant 0, \\ \sum\limits_{t\in T}\theta_{td} = 1;\ \theta_{td} \geqslant 0. \end{cases}$$

### 3.2.2 Классификация при помощи тематической модели

При классификации каждому документу соответствует набор элементов из конечного множества классов $C$. Задача заключается в том, чтобы выявить связи между классами и темами, улучшить качество тематической модели и построить алгоритм классификации новых документов, для которых метки классов ещё не проставлены.

Будем полагать, что классы $c \in C$ описываются неизвестными условными распределениями $p(t|c)$ на множестве тем $T$. Введем вероятностное пространство $D \times W \times T \times C$, считая, что с каждым словом $w$ в каждом документе $d$ связана не только тема $t \in T$, но и класс $c \in C$. Рассмотрим тематическую модель, в которой распределение вероятностей классов документов $p(c|d)$ описывается смесью распределений классов тем $p(c|t)$ и тем документов $p(t|d)$, где новой неизвестной является матрица классов тем $\Psi = ||\psi_{cd}||$. Будем моделировать распределение классов документов $p(c|d)$ через распределение тем документов $\theta_{td} = p(t|d)$ по аналогии с основной тематической моделью $p(w|d)$ (модель Dependency-LDA, [11]):

$$p(c|d) = \sum_{t\in T} p(c|t)p(t|d) = \sum_{t\in T}\psi_{ct}\theta_{td}.$$

При этом для классов постулируется гипотеза условной независимости $p(c|t,d) = = p(c|t)$, означающая, что для классификации документа $d$ достаточно знать только его тематику. Обозначим через $m_{dc}$ бинарный индикатор того, что документ $d$ относится к классу $c$. $M = ||m_{dc}||_{D\times C}$ — матрица, содержащая всю обучающую информацию о документах.

Для построения регуляризатора будем минимизировать KL-дивергенцию между моделью классификации $p(c|d)$ и эмпирическим распределением $\hat{p}(c|d) \sim m_{dc}$:

$$R(\Psi, \Theta) = \tau \sum_{d \in D} \sum_{c \in C} m_{dc} \ln \sum_{t \in T} \psi_{ct} \theta_{td} \to \max,$$

где коэффициент регуляризации $\tau$ необходим для регулирования отношения влияния частот слов $n_{dw}$ и частот классов $m_{dc}$. Таким образом, общая оптимизационная задача:

$$\begin{cases} \sum\limits_{d \in D} \sum\limits_{w \in d} n_{dw} \ln \sum\limits_{t \in T} \varphi_{wt} \theta_{td} + \tau \sum\limits_{d \in D} \sum\limits_{c \in C} m_{dc} \ln \sum\limits_{t \in T} \psi_{ct} \theta_{td} \to \max\limits_{\Phi, \Psi, \Theta}, \\ \sum\limits_{w \in W} \varphi_{wt} = 1; \ \varphi_{wt} \geqslant 0, \\ \sum\limits_{w \in W} \psi_{ct} = 1; \ \psi_{ct} \geqslant 0, \\ \sum\limits_{t \in T} \theta_{td} = 1; \ \theta_{td} \geqslant 0. \end{cases}$$

### 3.2.3 EM-алгоритм

Для решения классической задачи тематического моделирования используются различные вариации EM-алгоритма. Так как для решения задачи классификации с помощью тематической модели вводится дополнительная матрица $\Psi$, то существующий алгоритм должен быть обобщен. Результат — Алгоритм 1.

### 3.2.4 Регуляризация

При решении задачи тематического моделирования исходные матрицы $\Phi, \Psi, \Theta$ восстанавливаются не однозначно, а с точностью до невырожденного преобразования. Это влечет за собой неустойчивость EM-алгоритма, делая искомое решение зависимым от начального приближения. Формализуем требования к результирующим матрицам $\Phi, \Psi, \Theta$ за счет использования регуляризаторов.

Во-первых, так как тематическая модель нужна для генерации словарных признаков, то наличие во многих темах одинаковых слов будет противоречить построению качественных признаков. Во-вторых, для практических целей классификации было бы полезно иметь тематическую модель с сильно разреженными матрицами $\Phi$ и $\Theta$, в которых доля велика нулевых значений. Такие разреженные матрицы более удобны в использовании в пакетах прикладных программ, занимают меньше места и их использование уменьшает время работы алгоритма. Исследования, проведенные

**Алгоритм 1** ЕМ-алгоритм, одна итерация

**для всех** $d \in D$ **выполнить**
  **для всех** $w \in d$ **выполнить**
    $Z = \sum_{t \in T} \varphi_{wt}\theta_{td} = \Phi\Theta[w,d]$
  **для всех** $t \in T$ **выполнить**
    **если** $\varphi_{wt}\theta_{td} > 0$ **то**
      $x = N_{dw}\varphi_{wt}\theta_{td}/Z$
      увеличить $n_{wt}, n_{td}, n_t$ на $x$
  **для всех** $c \in C$ **выполнить**
    $Y = \sum_{t \in T} \psi_{ct}\theta_{td} = \Psi\Theta[c,d]$
  **для всех** $t \in T$ **выполнить**
    **если** $\psi_{ct}\theta_{td} > 0$ **то**
      $x = M_{cd}\psi_{ct}\theta_{td}/Y$
      увеличить $m_{ct}, m_{td}, m_t$ на $x$
$\varphi_{wt} = n_{wt}/n_t$
$\psi_{ct} = m_{ct}/m_t$
$\theta_{td} = \dfrac{n_{td} + m_{td}}{n_t + m_t}$

в [15] показали, что до некоторого предела разреживание не ухудшает контрольную перплексию.

Первое пожелание приводит к дополнительному требованию увеличивать различность тем. В качестве меры различности тем была выбрана ковариация и использован ковариационный регуляризатор:

$$R(\Phi,\Theta) = -\frac{\tau}{2}\sum_{t\in T}\sum_{s\in T/t} cov(\varphi_t,\varphi_s) \to \max_{\Phi,\Theta},$$
$$cov(\varphi_t,\varphi_s) = \sum_{w\in w}\varphi_{wt}\varphi_{ws}$$

Второе пожелание приводит к разреживающему энтропийному регуляризатору [15]:

$$R(\Phi,\Psi,\Theta) = -\beta\sum_{w\in W}\sum_{t\in T}\ln\varphi_{wt} - \gamma\sum_{c\in C}\sum_{t\in T}\ln\psi_{ct} - \alpha\sum_{d\in D}\sum_{t\in T}\ln\theta_{td} \to \max_{\Phi,\Psi,\Theta}$$

Приведенные регуляризаторы меняют формулы для $M$-шага (шаги 14-16 в Алгоритме 1):

$$\varphi_{wt} \sim (n_{wt} - \beta - \tau\sum_{s\in T/t}\varphi_{wt}\varphi_{ws})_+,$$
$$\psi_{ct} \sim (n_{ct} - \gamma)_+,$$
$$\theta_{td} \sim (n_{dt} - \alpha)_+.$$

### 3.2.5 EM-алгоритм в матричной форме

Запишем EM-алгоритм для задачи с регуляризаторами. Обозначим через $\otimes$ поэлементное произведение элементов двух матриц, через $\oslash$ — их же поэлементное частное (отметим, что матрицы должны иметь одинаковую размерность для корректного применения операции).

Пусть также $\Omega = \sum_{t\in T}\varphi_{wt}$; $1_{k\times s}$ — матрица, состоящая из единиц размера $k \times s$.

# 4 Методика формирования выборки

На рис. 1 изображен процесс разработки алгоритма классификации. Поясним, как происходит процесс доразметки выборки.

**Алгоритм 2** EM-алгоритм для ARTM, одна итерация, матричный вид

**Вход:** $\Phi, \Psi, \Theta, \tau, \alpha, \beta, \gamma$.

1: $\Phi_{new} = \Phi \otimes \Big[N^T \oslash (\Phi\Theta)\Big]\Theta^T$

2: $\Psi_{new} = \Psi \otimes \Big[M^T \oslash (\Psi\Theta)\Big]\Theta^T$

3: $\Theta_{new} = \Theta \otimes \Phi^T\Big[N^T \oslash (\Phi\Theta)\Big] + \tau\Theta \otimes \Psi^T\Big[M^T \oslash (\Psi\Theta)\Big]$

4: Нормировка столбцов $\Phi_{new}, \Psi_{new}, \Theta_{new}$.

5: $\Omega = \sum\limits_{t \in T} \varphi_{wt}$

6: Регуляризация $\Phi$ : $\Phi_{new} = (\Phi_{new} - \beta 1_{W \times T} - \eta\Phi \otimes [\Omega 1_{1 \times T} - \Phi])_+$

7: Регуляризация $\Psi$ : $\Psi_{new} = (\Psi_{new} - \gamma 1_{C \times T})_+$

8: Регуляризация $\Theta$ : $\Theta_{new} = (\Theta_{new} - \alpha 1_{T \times D})_+$

9: Нормировка столбцов $\Phi_{new}, \Psi_{new}, \Theta_{new}$.

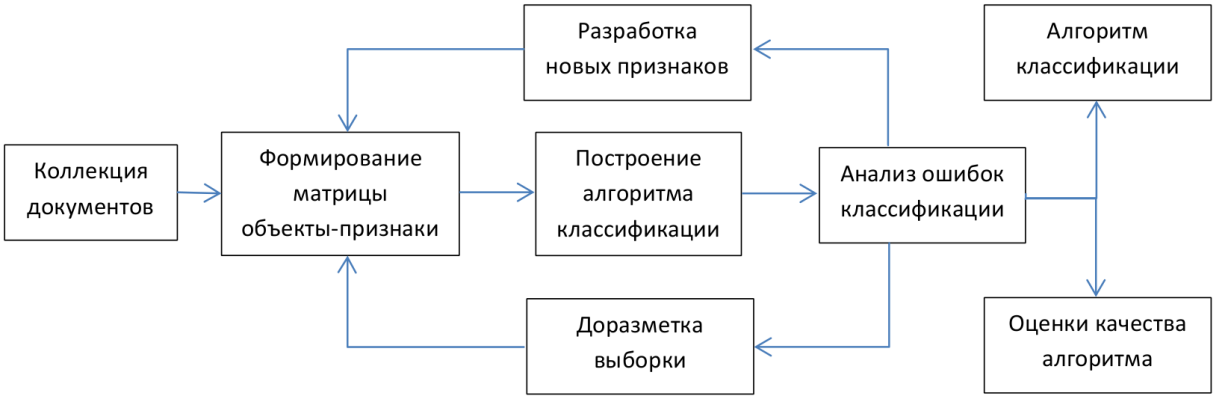



Рис. 1: Алгоритм

## 4.1 Фиксированный классификатор

Для разметки базовой выборки был придуман набор начальных признаков и построен классификатор с фиксированными весами:

$$a(d, w, w_0) = \text{sign}\left(\sum_{j=1}^{n} w_j f_j(d) - w_0\right) = \text{sign}(\langle x, w\rangle - w_0),$$

где все значения $(w_1, w_2, ..., w_n)$ задавались экспертами. В качестве признаков использовались: Греческие буквы, Большой размер, Маленький размер, Количество цифр, Математические символы (1, 2, 3, 4, 5).

## 4.2 Перевзвешивание несбалансированной выборки

Так как коллекция является сильно несбалансированной, то в случайной выборке будет слишком много ненаучных. Если обучающая выборка будет сильно несбалансированной, то существует вероятность неверной настройки (например, классифицировать большинство документов, как ненаучные). Кроме того, для настройки разделяющей гиперплоскости, хотелось бы брать побольше документов из области, близкой к это гиперплоскости. На рис. 2 можно увидеть типичный график распределения документов по расстоянию до гиперплоскости.

Для того, чтобы среди выбранных документов было достаточно близких к предполагаемой разделяющей гиперплоскости был использован Алгоритм 3.

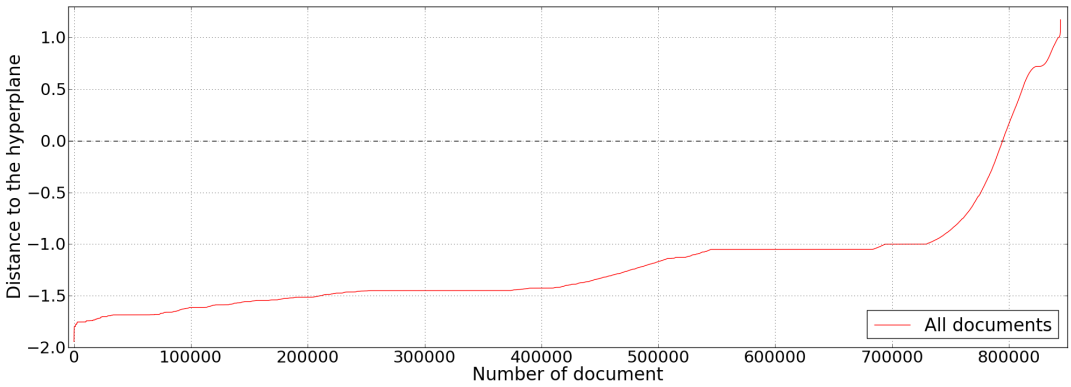



Рис. 2: Распределение документов по расстоянию до разделяющей гиперплоскости для SVM с некоторым набором признаков

**Алгоритм 3** Алгоритм выбора документов

**Вход:** $n$ – необходимое число документов

1: Зафиксировать вектор весов признаков $w$

2: Для каждого из документов определить $M_d = (\langle d, w\rangle - w_0)$

3: Из полученного распределения $\{M_d\}$ равномерно выбрать $n$ документов

Алгоритм 3 позволяет взять достаточное число как документов вблизи разделяющей гиперплоскости, так и вдали от неё. Посмотрим на рис. 2: неформально говоря, если изначально документы выбирались равномерно по оси $x$, то теперь они выбираются равномерно по оси $y$. При условии того, что фиксируемый классификатор имеет невысокий процент ошибок, то среди выбранных документов несбалансированность классов проявляется гораздо меньше. Например, для данной выборки процент научных документов вырос с $\sim 2\%$ до $40\%$ (при переходе от общей выборки к размеченным документам). Однако, заметим, что если мы хотим оценить ошибку классификации на всей выборке, то оценка ошибки только на размеченной выборке будет смещена в сторону научных документов, которые в полной выборке попадаются сравнительно редко.

Пусть $D_1$ — множество размера $n$, полученное независимым выбором документов из распределения $p(d)$; $D_2$ — из распределения $q(d)$ такого, что $\forall d\ q(d) > 0$.

$$Err(a) = \frac{1}{n}\sum_{d\in D_1} f(d), \tag{4.1}$$

$$Err'(a) = \frac{1}{n} \sum_{d \in D_2} f(d) \frac{p(d)}{q(d)}. \tag{4.2}$$

Тогда 4.2 является несмещенной оценкой 4.1. В случае, когда $p$ является равномерным распределением, оценка 4.2 называется оценкой Хансена-Гурвица [4].

Заметим, что переход от 4.1 к 4.2 эквивалентен следующему переходу: если раньше для каждого документа $d$, на котором вычислялась оценка ошибки, вклад в сумму вносился с весом 1, то теперь он считается с весом $\frac{p(d)}{q(d)}$. Поэтому достаточно лишь сохранить для каждого документа $d$ его новый вес $h(d)$ и оценку ошибки считать как:

$$Err(a) = \frac{1}{n} \sum_{d \in D_2} f(d) h(d).$$

# 5 Эксперименты

## 5.1 Фиксированный классификатор

Цель: проверить, что ненастраиваемый классификатор не обеспечивает необходимой точности классификации.

Зависимость качества классификации (доля правильно классифицированных документов) от порога принятия решения ($w_0$) можно увидеть на рис. 3.

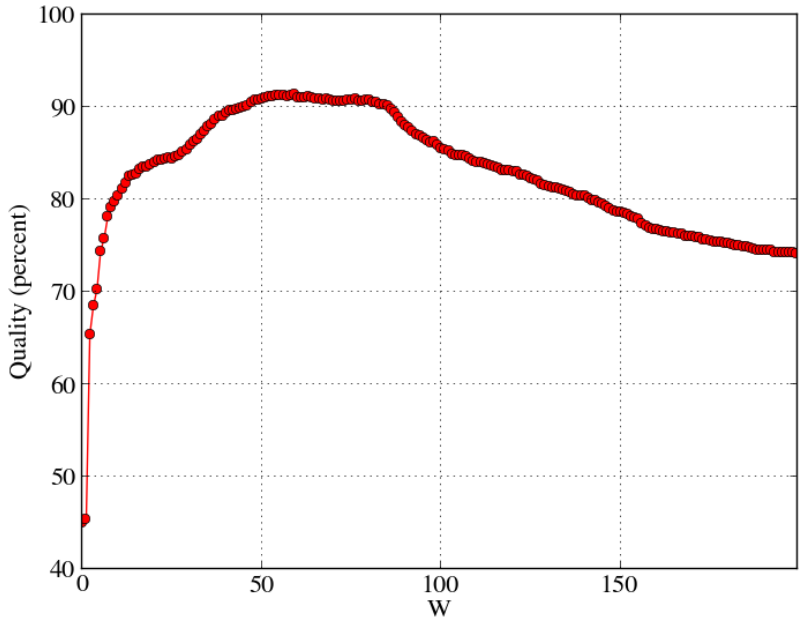



Рис. 3: Качество классификации в зависимости от $w_0$

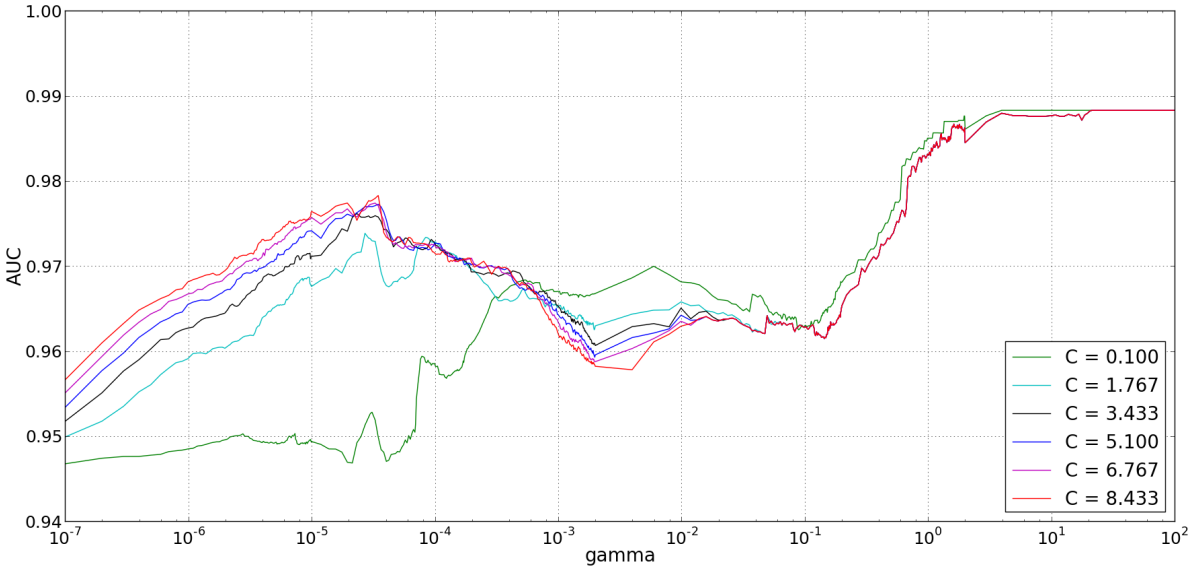



Рис. 4: Зависимость AUC от $\gamma$ для различных значений $C$ для SVM с ядром *rbf*

Из рис. 3 видно, что необучаемый классификатор не обеспечивает приемлегого качества классификации.

Для классификации использовались признаки: Греческие буквы, Большой размер, Маленький размер, Количество цифр, Математические символы (1, 2, 3, 4, 5).

## 5.2 Машина опорных векторов

Для использования SVM была взята реализация классификатора из библиотеки scikit-learn [8].

### 5.2.1 Выбор ядра

Цель: проверить скорость обучения и качество классификации для ядер *linear, poly, rbf*.

Эксперименты показали, что линейное ядро имеет недостаточное качество классификации, а полиномиальное — слишком долго обучается. При этом, *rbf* показал наивысшее качество классификации и приемлемую скорость обучения. Поэтому в дальнейшем использовалось именно это ядро.

### 5.2.2 Зависимость AUC от $C$ и $\gamma$

Цель: найти оптимальные для максимизации AUC параметры классификатора.

На рис. 4 изображена зависимость AUC от параметров классификатора. Параметры SVM, по-умолчанию используемые в [8]: $= 1, \gamma = \frac{1}{\text{количество признаков}}$.

Эксперименты показали, что оптимальные параметры: $C > 4$ и $\gamma > 10$, причем изменение $C$ (при таких $\gamma$) в предположении, что $C > 2$, почти не влияет на AUC.

## 5.3 Логистический классификатор

В данной работе использовались два различных метода настройки весов: градиентный спуск (в дальнейшем — ГС) и метод Левенберга-Марквардта (в дальнейшем — ЛМ).

### 5.3.1 Необходимое число итераций

Цель: исследовать необходимое число итераций для достижения оптимального в плане величины AUC набора весов.

Эксперименты показали, что начиная с некоторого количества итераций перестает меняться AUC для построенных классификаторов, но коэффициенты продолжают расти, что влечет за собой разреживание матрицы объекты-классы (то есть вероятности принадлежать классу $c$ становятся равными 0 или 1). Типичную картину зависимости AUC можно увидеть на рис. 5. На рис. 8 изображены типичные графики зависимости коэффициентов вектора весов от количества итераций.

Для метода ГС необходимое количество итераций не превосходит 50, а для метода ЛМ — 20.

### 5.3.2 Оптимальный метод настройки весов и стратегия получения классификатора

Цель эксперимента: сравнить два метода настройки весов, определить оптимальную стратегию для получения классификатора с максимальным AUC.

Результаты:

- Одна итерация ГС требует меньше времени, чем ЛМ
- Количество итераций, необходимых для того, чтобы AUC перестало меняться у ЛМ в среднем меньше, чем у ГС
- Итоговое значение AUC непостоянно и зависит от обучающей выборки и начального приближения; однако в среднем у ЛМ AUC получается больше

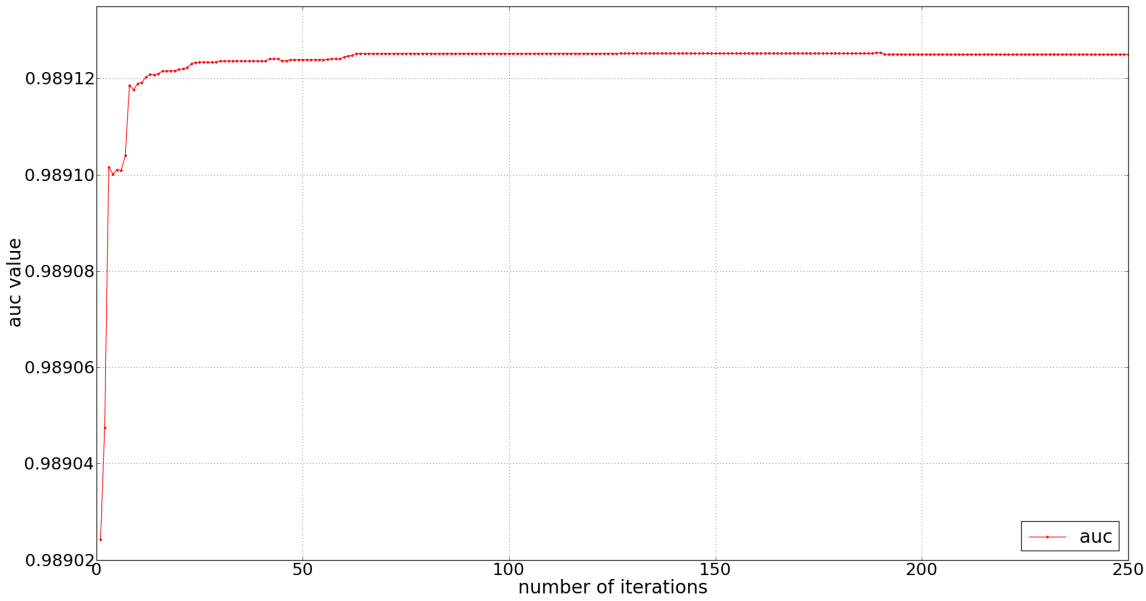



Рис. 5: Типичная зависимость AUC от числа итераций

- 95% доверительный интервал AUC: ГС — [0.981–0.991], ЛМ — [0.983–0.992].

Оптимальная стратегия получения классификатора: обучить несколько классификаторов ЛМ и выбрать из них тот, который имеет наибольшее значение AUC.

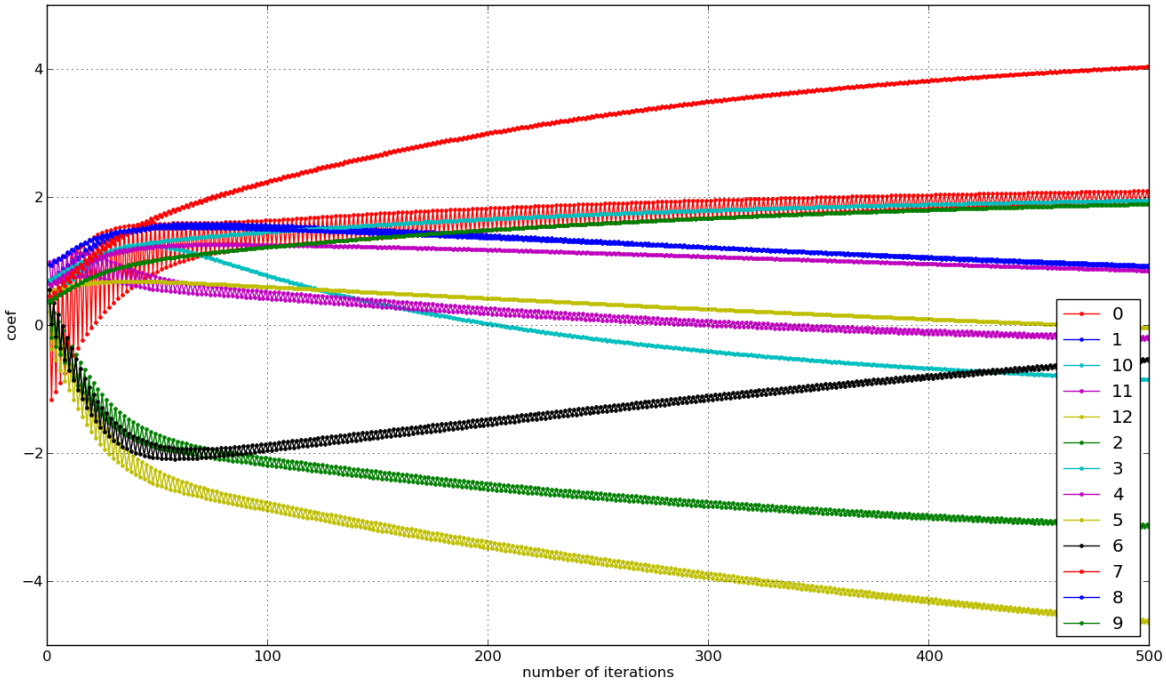
### 5.3.3 Зависимость AUC от коэффициента регуляризации $\lambda$

Цель: проверить, как зависит AUC от $\lambda$.

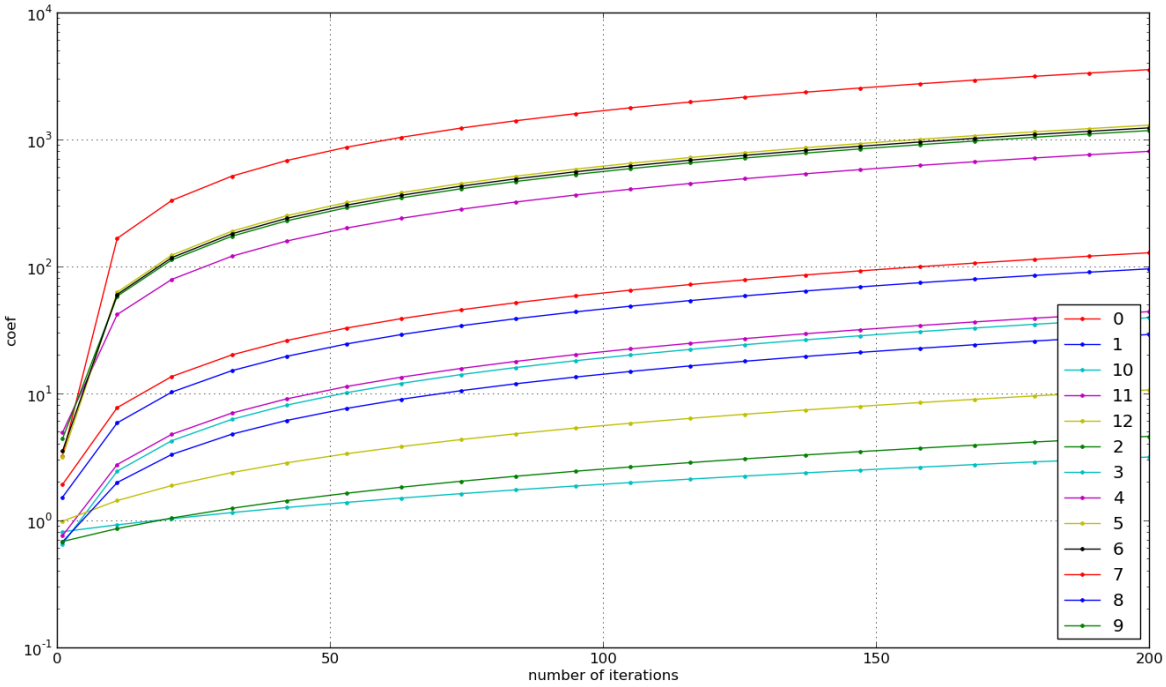
Эксперименты показали, что при одинаковой обучающей выборке значение $\lambda$ не сильно влияет на AUC, типичную картину можно увидеть на рис. 7. Однако, свободный член классификатора, отвечающий за границу между классами, не всегда настраивается корректно (то есть принимает слишком большое или слишком маленькое значение). Его следует настроить отдельно (значение свободного члена не влияет на AUC и может быть выбрано сообразно ROC-кривой).

### 5.3.4 Зависимость коэффициентов классификатора от $\lambda$ при различных $\mu$

Цель: понять, как зависят значения весов от параметров классификатора после фиксированного числа итераций.




a) настройка методом ГС (при большом $\lambda$)




b) настройка методом ЛМ (при маленьком $\lambda$ и маленьком $\mu$)

Рис. 6: Зависимость значения коэффициентов логистического классфикатора от числа итераций

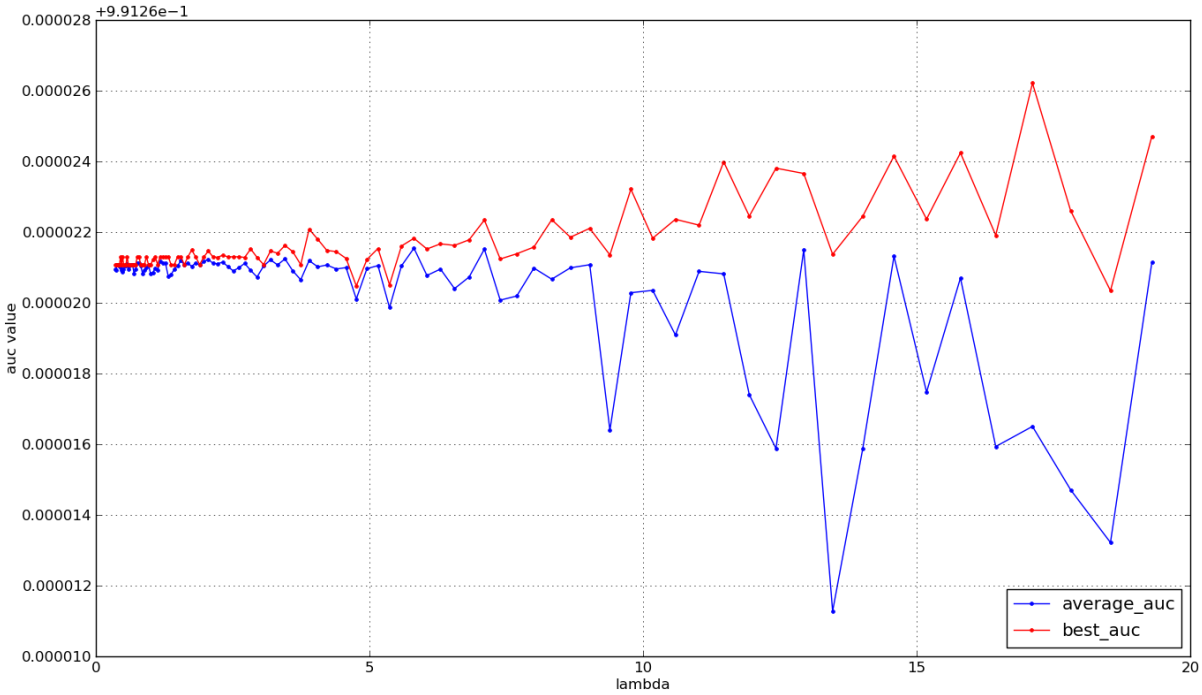
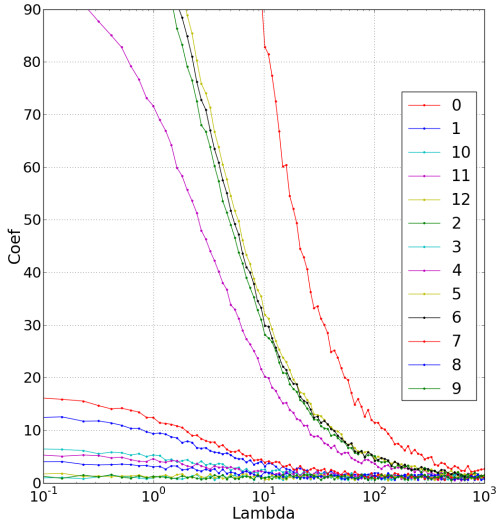
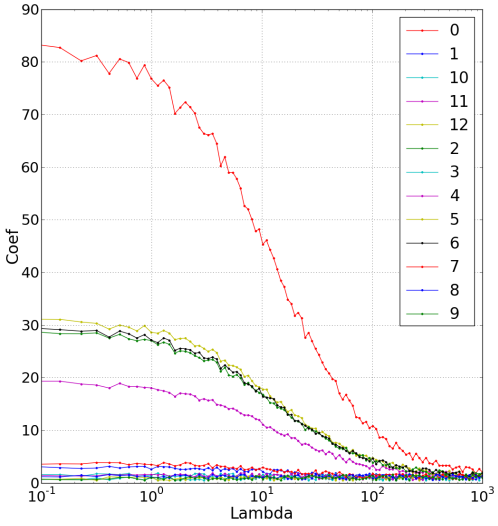


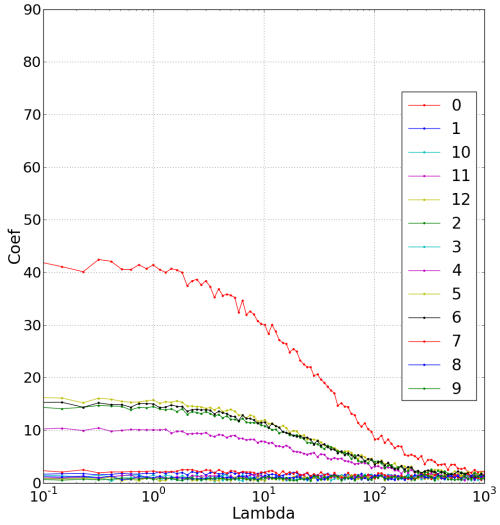

Рис. 7: Зависимость AUC от $\lambda$ (best_auc – лучший AUC среди 10 запусков алгоритма с одинаковой обучающей выборкой, но разными начальными приближениями; average_auc – средний среди них же)



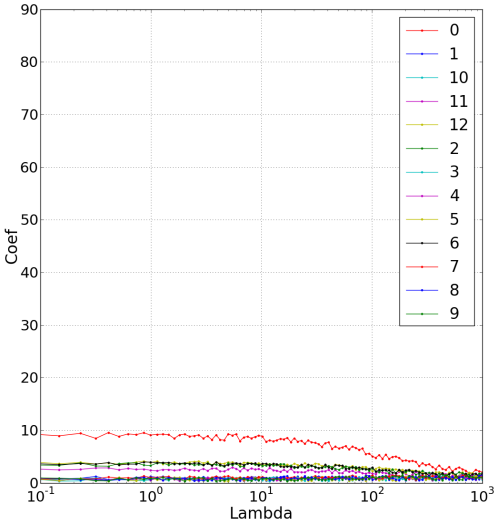

a) $\mu = 2$




b) $\mu = 10$




c) $\mu = 20$




d) $\mu = 100$

Рис. 8: Зависимость модулей весов признаков от $\lambda$ при различных $\mu$ при настройке методом Левенберга-Марквардта после 50 итераций.

Так как при небольших значениях весов лучше настраивается свободный член (увеличивая качество классификации), то сообразно выбрать такие значения коэффициентов, что итоговые веса будут не слишком большими. Результат можно видеть на рис. 8.

Стоит выбирать большое $\mu \sim 100$ и большое $\lambda \sim 0.5$.

## 5.4 Тематическая модель классификации

### 5.4.1 Классификация с помощью тематической модели

Цель: проверить, можно ли использовать тематическую модель с регуляризаторами разрежвания и рзаличия тем, как классификатор.

Эксперименты показали, что классфикатор на основе тематической модели имеет высокий процент ошибок (порядка 30-35%). Причем, этот процент не имеет тенденции к уменьшению с увеличением количества итераций. По проценту ошибок тематическая модель значительно уступает представленным выше классификаторам, что исключает её использование, как классификатора без дополнительного масштабного исследования подбора регуляризаторов и их параметров.

Кроме того, согласно тестам Алгоритм 2 быстрее, чем Алгоритм 1 в 6.5 раз при полностью заполненных матрицах $\Phi, \Psi, \Theta$ и более, чем в 10 раз быстрее при разреженных матрицах.

### 5.4.2 Получение признаков из тематической модели

Цель: получить новые словарные признаки, как результат работы тематической модели.

Результирующие матрицы $\Psi$ и $\Phi$ могут дать важную информацию о коллекции, полезную для дальнейшей классификации. В частности, матрица $\Psi$ для каждой из тем дает вероятности принадлежности темы классу. Эксперименты показали, что после некоторого числа итераций матрица $\Psi$ выглядеть также, как показано в Таблице 1.

Таблица 1: Типичный вид матрицы $\Psi$ после 50 итераций

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $c = -1$ | 0.495 | 0.544 | 0.983 | 0.932 | 1.000 | 0.117 | 1.000 | 0.605 | 0.000 | 0.115 | ... |
| $c = +1$ | 0.505 | 0.456 | 0.017 | 0.068 | 0.000 | 0.883 | 0.000 | 0.395 | 1.000 | 0.885 | ... |

Таким образом, если в поле темы $t$ с $c = +1$ стоит достаточно большое число (например, больше 0.95), то эта тема классифицирована моделью как научная. Поэтому часто встречающиеся слова из этой темы могут составить признак, аналогичный признакам 7-10. Для этого нужно, чтобы большая часть из этих слов была интерпертируема вместе экспертом. Примеры топа слов тем можно увидеть в Таблице 2. Следствие формализации этих требований является Алгоритм 4.

**Алгоритм 4** Алгоритм генерации словарных признаков

1: Запустить тематический классификатор

2: Для всех $t : p(+1|t) > 0.95$ просмотреть слова $w_t : p(w|t) > 0.008$

3: Для всех $t$ таких, что набор слов $w_t$ интерпретируем, создается признак $\text{Пр}_t$. Слову $w$ в соответствие ставим вес, равный $c_w = p(w|t)$. Значение $\text{Пр}_t$ на документе $d$: $\sum_{w \in d} c_w \log(b_w)$, где $b_w$ — количество вхождений слова $w$ в документ $d$.

Понятно, что различные признаки могут теперь иметь существенно разные значения (например: один признак принимает значения в промежутке [0, 100], а другой [0, 0.5]). Для того, чтобы этого избежать, нужна перенормировка значений признаков по всем документам так, чтобы максимальное и минимальное значения у всех признаков совпадали (если признак не является константой). Была сделана линейная перенормировка в диапазон [0, 100].

Для дальнейших тестов были отобраны 8 новых словарных признаков, полученных с помощью тематического моделирования.

## 5.5 Итоговые классификаторы

Цель: построить ROC-кривые полученных классификаторв

На рис. 9 изображены ROC-кривые классификаторов (вместе признаками, полученными в предыдущем эксперименте).

У логистического классификатора часть ненаучных документов превалирует над небольшой группой ненаучных. Однако, при этом за счет варьирования свободного члена можно добиться баланса ошибок первого/второго рода.

SVM небольшая группа научных документов классифицируются как «сильно» ненаучные, а небольшая часть ненаучных — как «сильно» научные. За счет малости этих групп достигается относительно большое значение AUC, но пропадает возможность балансировки ошибок.

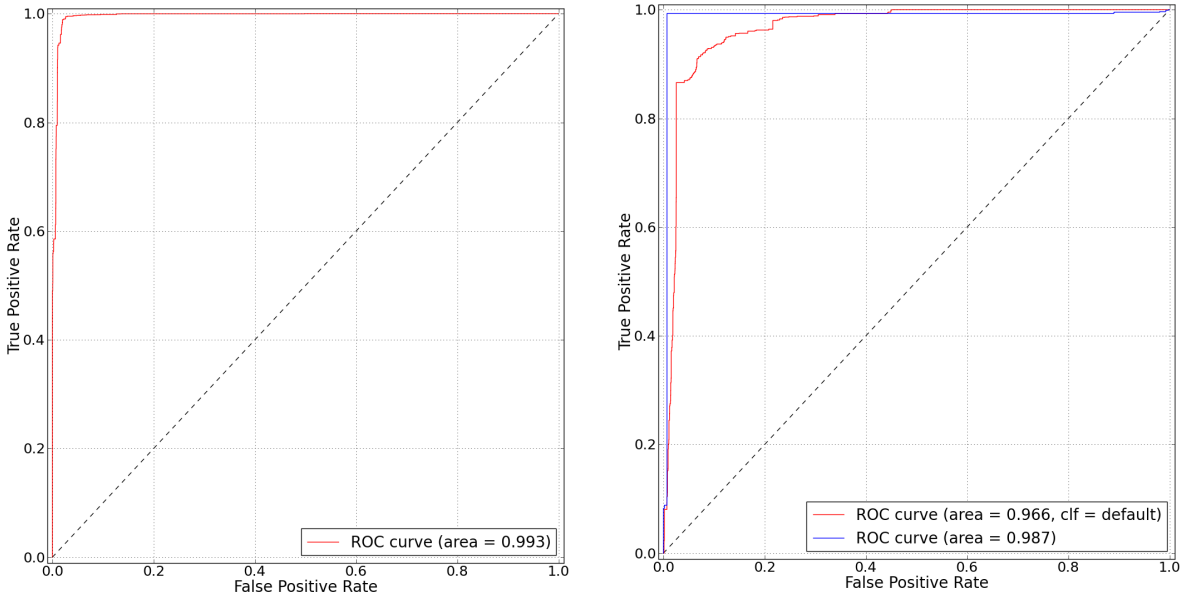
# 6 Общий список признаков

1. Греческие буквы

**Количество греческих букв в документе**

Таблица 2: Топ слов из первых семи тем тематической модели классификации. В первой строке указана $p(c = +1|t)$ (вероятность того, что тема научная)

| 0.755 | 0.000 | 0.392 | 0.099 | 0.203 | 1.000 | 0.455 |
|---|---|---|---|---|---|---|
| прямая | образование | быть | который | страна | процесс | этот |
| быть | студент | этот | ряд | россия | результат | такой |
| точка | социальный | себя | оценка | производство | модель | другой |
| значение | учебный | мочь | человек | экономика | среда | вопрос |
| множество | современный | весь | параметр | который | зависимость | первый |
| движение | университет | такой | источник | год | различный | иметь |
| состояние | российский | каждый | система | этот | структура | они |
| система | научный | условие | помощь | орган | позволять | русский |
| время | формирование | который | связь | государство | являться | любой |
| рис | образовательный | распределение | этот | проблема | поверхность | она |
| временить | международный | сила | комплекс | развитие | расчет | общество |
| тогда | организация | цена | наличие | период | технический | масса |
| исследование | проект | функция | изменение | федеральный | обработка | мочь |
| свойство | история | некоторый | труд | закон | качество | почва |
| граница | информационный | тип | мир | оно | данный | новый |
| вектор | вуз | момент | знание | хозяйство | моделирование | автор |
| уровень | кафедра | продукт | высокий | такой | сигнал | получить |
| коэффициент | личность | происходить | величина | власть | следующий | один |
| представление | субъект | модель | число | стоимость | основа | вещество |
| теория | стандарт | тома | соответствующий | весь | учет | какой |
| поле | место | определение | соответствие | условие | получение | пря |
| фаза | знанье | полный | реальный | высокий | выша | много |
| класс | сфера | система | район | общий | возможность | смысл |
| выражение | конференция | волна | весь | человек | концентрация | общественный |
| внешний | исследование | определить | лицо | крупный | центр | природа |
| возможный | учреждение | оно | еще | единый | снижение | наука |
| операция | литература | длина | данный | государственный | метод | свое |
| физический | направление | состояние | ток | действие | количество | теорема |
| вероятность | являться | здесь | должен | отрасль | вид | эксперимент |
| работа | мир | язык | частота | регулирование | эффект | функция |
| три | практический | давать | философия | национальный | давление | известный |
| очень | уровень | они | отдельный | поддержка | оптимальный | раз |




a) Линейная регрессия

b) SVM (ненастроенный и настроенный классификаторы)

Рис. 9: ROC-кривые для полученных классификаторов

Некоторая часть научных документов содержит греческие буквы, так как в формулах они часто используются как переменные.

Для $j$-й греческой буквы, встречающейся в документе, обозначим через $b_j$ количество раз, которое она встретилась в документе. Значение признака на документе: $\sum_j \log(b_j)$

2. Большой размер

**Количество символов в тексте**

Научные документы, которые содержат важные сведения, не могут быть слишком короткими.

Обозначим через $b$ количество символов в тексте, $C$ — порог признака ($\sim 15000$). Значение признака на документе:

$$\begin{cases} \log b, & \text{если } b > C, \\ 0 & \text{иначе.} \end{cases}$$

3. Маленький размер

**Индикатор размера текста**

Довольно сложно представить себе маленький научный текст (с учетом того, что аннотации не хотелось бы считать научными).

Обозначим через $b$ количество символов в тексте, $C$ — порог признака ($\sim 4000$). Значение признака на документе: $[b < C]$

4. Количество цифр

**Индикатор наличия числовых данных**

Обозначим через $b$ количество цифр в документе, $C$ — порог признака Значение признака на документе: $[b > C]$

5. Математические символы

**Количество математических символов**

Многие научные документы содержат математические символы, вроде $\int$, $\sum$.

Для $j$-го символа из $\{\sum, \Delta, \nabla, \in, \notin, \prod, \surd, \int, \iint, \oint, \subset, \not\subset, \leqslant, \geqslant, \neq, \times, ...\}$, встречающегося в документе, обозначим через $b_j$ количество раз, которое он встретился в документе. $c_j$ — важность символа. Значение признака на документе: $\sum_j c_j \log(b_j)$

6. Типичные грантовые фразы

**Наличие в тексте фраз, относящимся к грантам**

Некоторые научные статьи являются отчетными по грантам. Поэтому в них есть фразы про финансирующий грант.

Значение признака на документе: количество фраз, встреченных в документе (если во фразе переставлены слова, то она считается с меньшим весом)

7. Научные термины

**Количество научных терминов**

Многие научные документы содержат научные термины, такие как теорема, утверждение и т.д.

Для $j$-го термина, встречающегося в документе, обозначим через $b_j$ количество раз, которое оно встретилось в документе. $c_j$ — важность термина. Значение признака на документе: $\sum_j c_j \log(b_j)$

8. Имена ученых

**Имена ученых, встречающиеся в критериях/теоремах**

Во многих естественно-научным работам используются именные критерии или теоремы.

Для $j$-го имени, встречающегося в документе, Обозначим через $b_j$ количество раз, которое оно встретилось в документе. $c_j$ — важность имени. Значение признака на документе: $\sum_j c_j \log(b_j)$

9. Стоп-слова

**Количество стоп-слов**

в коллекции есть довольно большой пласт документов, связанных с образовательными документами (как-то программа подготовки студентов по курсу)

Для $j$-го стоп-слова, встречающегося в документе, Обозначим через $b_j$ количество раз, которое оно встретилось в документе. $c_j$ — важность стоп-слова (что важно, отрицательная: чем менее вероятно,что данное стоп-слово встретится в научном документе, тем меньше $c_j$). Значение признака на документе: $\sum_j c_j \log(b_j)$

10. Структура текста

**Индикаторы наличия системообразующих слов научного контента**

такие слова, как Постановка задачи или Список литературы (или их аналоги) почти наверняка есть в каждой научной статье

Для $j$-го структурного слова, встречающегося в документе, Обозначим через $b_j$ количество раз, которое оно встретилось в документе. Значение признака на документе: $\sum_j c_j[b_j > 0]$

Выделенные красным признаки образованы следующим образом: вручную выбран список слов и проставлены веса слов $c_j$, итоговое значение признака на документе равно $\sum_j c_j \log(b_j)$. Процесс генерации признаков довольно однообразен и сильно зависит от эксперта; кроме того, при большом объеме размеченной выборки, проглядеть все документы за адекватное время становится малореально. Метод полуавтоматической генерации признаков призван решить эту проблему.

# 7 Полученные результаты

В задаче есть два класса документов: научные и ненаучные. Будем различать ошибки вида научный документ классифицирован, как ненаучный и наоборот. Ошибку вида «научный документ классифицирован, как ненаучный» будем называть ошибкой первого рода, а «ненаучный документ классифицирован, как научный» — второго.

Качество построенных классификаторов и их AUC можно узнать из Таблицы 3. Разница между SVM и логистической регрессией залючается в том, что SVM дает более стабильные результаты, поэтому для него можно взять усредненные результаты. Для регрессии указан 95% доверительный интервал.

Примеры топа тем, отобранных, как новые словарные признаки в Таблице 10.

# 8 Заключение

- Разработана система признаков для линейной модели классификации документов по жанрам: научный, ненаучный

Таблица 3: Характеристики полученных классификаторов

| | Стадия работы | Ошибка | | | AUC |
|---|---|---|---|---|---|
| | | Полная | 1 рода | 2 рода | |
| | Базовый классификатор | 10.2% | - | - | - |
| SVM | Базовая версия | 9.6% | 3.2% | 6.4% | 0.91 |
| | Настройка параметров | 8.0% | 4.2% | 3.8% | 0.93 |
| | Признаки научных слов | 7.4% | 4.2% | 3.2% | 0.95 |
| | Улучшение признаков | 4.0% | 2.1% | 1.9% | 0.983 |
| | + признаки из ТМ | 3.7% | 2.3% | 1.4% | 0.985 |
| RLR | Градиентный спуск | 5.0% (3.2–6.3) | 2.6% | 2.4% | 0.981–0.991 |
| | Метод Левенберга-Марквардта | 4.6% (2.7–6.6) | 2.3% | 2.3% | 0.983–0.992 |
| | + признаки из ТМ (ГС) | 5.2% (3.1–6.4) | 2.7% | 2.5 % | 0.976–0.994 |
| | + признаки из ТМ (ЛМ) | 3.9% (2.2–5.0) | 2.0% | 1.9% | 0.985–0.996 |

Рис. 10: Примеры признаков, полученных из тематической модели

| $p_w$ | w |
|---|---|
| 0.0575 | система |
| 0.0450 | рис |
| 0.0305 | модель |
| 0.0299 | функция |
| 0.0286 | значение |
| 0.0284 | параметр |
| 0.0235 | характеристика |
| 0.0232 | уравнение |
| 0.0230 | процесс |
| ⋮ | ⋮ |

| $p_w$ | w |
|---|---|
| 0.0398 | результат |
| 0.0342 | анализ |
| 0.0296 | вид |
| 0.0296 | исследование |
| 0.0286 | решение |
| 0.0279 | метод |
| 0.0245 | задача |
| 0.0217 | использование |
| 0.0201 | фактор |
| ⋮ | ⋮ |

- Предложен метод формирования словарных признаков на основе регуляризованной тематической модели
- Выполнена программная реализация и проведены численные эксперименты показавшие, что использование данных признаков улучшает качество классификации

# Список литературы

[1] Christopher M Bishop et al. *Pattern recognition and machine learning*, volume 1. springer New York, 2006.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] O. Dekel and O. Shamir. Multiclass-multilabel classification with more classes than examples. *Journal of Machine Learning Research – Proceedings Track*, 2010.

[4] Morris H Hansen, William N Hurwitz, and William G Madow. Sampling survey methods and theory. *Vol I, John Wiley and Son Inc., New York*, 1953.

[5] Thorsten Joachims. *Text categorization with support vector machines: Learning with many relevant features*. Springer, 1998.

[6] T.-Y. Liu, Y. Yang, H. Wan, H.-J. Zeng, Z. Chen, and W.-Y. & Ma. Support vector machines classification with a very large-scale taxonomy. *SIGKDD Explorations Newsletter*, 7(1):36–43, 2005.

[7] Larry M Manevitz and Malik Yousef. One-class svms for document classification. *the Journal of machine Learning research*, 2:139–154, 2002.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[9] Muhammad Rafi and Mohammad Shahid Shaikh. A comparison of svm and rvm for document classification. *arXiv preprint arXiv:1301.2785*, 2013.

[10] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 conference on empirical methods in natural language processing*, 2009.

[11] T N Rubin, A Chambers, P Smyth, and M Steyvers. Statistical topic models for multi-label document classification. *Machine Learning.*, 88(1-2):157–208, 2012.

[12] Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. Text genre detection using common word frequencies. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*, pages 808–814. Association for Computational Linguistics, 2000.

[13] J. Zhu, A. Ahmed, and E. P. Xing. Medlda: Maximum margin supervised topic models for regression and classification. In *Proceedings of the 26th annual international conference on machine learning*, 2009.

[14] Sven Meyer zu Eißen and Benno Stein. Genre classification of web pages. In *KI 2004: Advances in Artificial Intelligence*, pages 256–269. Springer, 2004.

[15] КВ Воронцов and АА Потапенко. Регуляризация, робастность и разреженность вероятностных тематических моделей. *Компьютерные исследования и моделирование*, 4(4):693–706, 2012.