



Московский государственный университет имени М.В. Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра Математических Методов Прогнозирования

Думбай Алексей Дмитриевич

Методы поиска максимальных совместных подсистем системы линейных неравенств и их сравнение

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Научный руководитель:

д.ф.-м.н., профессор

В.В. Рязанов

Москва, 2018

Содержание

1	Введение	3
1.1	Обзор проблемы	3
1.2	Постановка задачи и основные понятия	3
1.3	Задача выбора наилучшего алгоритма классификации по прецедентам	4
2	Описание методов решения задачи МСП	5
2.1	Комбинаторный метод	5
2.2	Релаксационный метод	7
2.3	Метод аппроксимации сигмоидами	8
2.4	Модификация метода аппроксимации сигмоидами	9
3	Эксперименты	10
3.1	Дизайн экспериментов	10
3.2	Эксперименты на данных малой размерности	11
3.3	Эксперименты на данных малой размерности	13
3.4	Анализ параметризации аппроксимирующего метода	16
3.5	Анализ предлагаемой модификации алгоритма	18
4	Заключение	19
	Список литературы	21

Аннотация

В настоящей работе рассмотрена задача выделения максимальной совместной подсистемы системы линейных неравенств. Для данной задачи рассмотрены три метода, решающих ее: точный комбинаторный алгоритм, релаксационный алгоритм, и аппроксимационный. Все методы, кроме первого дают приближенный ответ. Аппроксимационный алгоритм основан на замене исходной оптимизационной задачи линейного программирования на выражения через сигмоиды. Также предложена модификация последнего метода, позволяющая находить максимальную совместную подсистему большей мощности.

Целью данной работы является реализация, исследование четырех методов - точного комбинаторного алгоритма, релаксационного алгоритма, аппроксимационного, предложенной модификации аппроксимационного и сравнение времени и точности их работы на синтетических данных, а также более детальное изучение аппроксимационного метода и его модификации.

Предложенный метод позволяет улучшить результаты, полученные с помощью метода аппроксимации сигмоидами, что показывают эксперименты.

1 Введение

1.1 Обзор проблемы

В работе будут рассмотрены методы решения задачи поиска максимальной совместной подсистемы системы линейных неравенств. Задача состоит в нахождении совместной подсистемы данной системы неравенств, в общем случае несовместной, с максимальным числом входящих в нее неравенств. Данная задача может использоваться как простой самостоятельный алгоритм бинарной классификации по прецедентам, так же она возникает в ряде задач машинного обучения и линейного программирования как вспомогательная.

Существуют несколько подходов для точного решения данной задачи: методы, основанные на сведении задачи к поиску минимального покрытия множества всех несократимых несовместных подсистем, динамическое генерирование несократимых несовместных подсистем, метод свертки системы линейных неравенств, методы, основанные на переборе узловых подсистем и др.

Стоит отметить, что задача выделения максимальной совместной подсистемы из несовместной системы линейных неравенств является NP-трудной, что приводит к неэффективности прямых решения, дающего точный результат при больших значениях по числу неравенств или переменных. Сейчас все больше и больше задач машинного обучения, для которых есть большие объемы данных. Для таких задач становятся критически важными не только точность решения, но так же и скорость работы алгоритма. По этой причине вызывает интерес изучение и сравнение методов приближенного решения данной задачи [1].

В данной работе будут рассмотрены следующие подходы к решению задачи нахождения максимальной совместной подсистемы: релаксационный метод [2], комбинаторный метод, аппроксимация неравенств сигмоидами. Также будет предложена модификация последнего алгоритма.

1.2 Постановка задачи и основные понятия

Пусть задана система линейных неравенств вида:

$$\begin{cases} f_1(x) = a_{11}x_1 + a_{12}x_2 + \dots + a_{1m}x_m + b_1 \geq 0, \\ f_2(x) = a_{21}x_1 + a_{22}x_2 + \dots + a_{2m}x_m + b_2 \geq 0, \\ \dots \\ f_n(x) = a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nm}x_m + b_n \geq 0. \end{cases}$$

Здесь

$$A \in \mathbb{R}^{n \times m}$$

$$b \in \mathbb{R}^n$$

В общем случае нет информации о совместности данной системы.

Определение 1.1. *Задача МСП — задача нахождения максимальной по мощности совместной подсистемы данной системы неравенств и нахождения как минимум одного решения данной системы.*

1.3 Задача выбора наилучшего алгоритма классификации по прецедентам

Пусть $\{A(\mathbf{w}, \mathbf{z}), \mathbf{w} \in \mathbb{R}^n\}$ — линейное параметрическое множество алгоритмов распознавания [3]. \mathbf{z} - выборка объектов размера M . Стоит задача l бинарных классификаций, т.е. ответов вида $z_i \in K_j, j = 1, \dots, l; i = 1, \dots, M$. Задача поиска наилучшего алгоритма в данном параметрическом семействе состоит в оптимизации функционала качества классификации. Пусть дана обучающая выборка - выборка объектов для которых известны ответы, т.е. принадлежность классам.

$$\|\alpha_{ij}\|_{i=1, \dots, M}^{j=1, \dots, l}; \alpha_{ij} = \begin{cases} 1, & \mathbf{z}_i \in K_j, \\ 0, & \mathbf{z}_i \notin K_j. \end{cases}$$

Каждый алгоритм выдает нам матрицу ответов такого же вида.

Определение 1.2. *Стандартный функционал качества — функционал, задаваемый формулой*

$$L(A) = \frac{1}{Ml} \sum_{i=1}^M \sum_{j=1}^l |\alpha_{ij} - \alpha_{ij}^A|$$

Задача состоит в поиске алгоритма A из множества $A(\mathbf{w}, \mathbf{z})$, который дает максимальное число правильных ответов. тем самым максимизируя стандартный функционал качества.

Тогда для линейных алгоритмов условие правильного распознавания некоторого объекта записывается как выполнение неравенства исходной системы неравенств. К поиску МСП системы линейных неравенств и близким задачам сводятся задачи обучения алгоритмов модели АВО в подпространствах «веса признаков», «веса объектов обучения», «параметры решающего правила», модели «линейная машина» и др.

2 Описание методов решения задачи МСП

2.1 Комбинаторный метод

Комбинаторный метод является точным методом решения задачи МСП. При этом, в силу NP-трудности данной задачи, реализуем только для задач с малой размерностью. Основная идея данного алгоритма - сокращение числа перебираемых подсистем. Задачу МСП можно свести к задаче поиска максимального верхнего нуля некоторой монотонной булевой функции. Для решения этой задачи в разработан алгоритм, оптимальный в смысле Шеннона. Рассмотрим этот алгоритм более детально.

Определим булеву функцию $g(y_1, y_2, \dots, y_n)$ таким образом: занумеруем неравенства системы числами $1, 2, \dots, n$. Каждой переменной y_i поставим в соответствие неравенство с тем же номером i . Произвольный булев вектор $a = (a_1, a_2, \dots, y_m)$ определяет подсистему \mathbf{F}_a исходной системы как бинарная маска. Положим $g(a)$ как индикатор совместности подсистемы \mathbf{F}_a . Определённая таким образом булева функция g будет монотонной, так как любая подсистема совместной системы совместна, а любое расширение несовместной – несовместно. По определению нули функции g соответствуют совместным подсистемам исходной системы, тогда её верхние нули будут соответствовать тупиковым совместным подсистемам, любое расширение которых несовместно, а максимальный верхний нуль – максимальной по мощности совместной подсистеме.

Перепишем исходную систему в следующем виде:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1m}x_m \geq b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2m}x_m \geq b_2, \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nm}x_m \geq b_n. \end{cases}$$

Мы хотим найти МСП данной системы S , что будет эквивалентно нахождению МСП исходной системы. Будем считать, что данная система не является совместной, в противном случае решение тривиально.

Определение 2.1. *Нерасширяемая совместная подсистема* — такая совместная подсистема системы S , что присоединение к ней любого, не входящего в нее, неравенства из системы S делает ее несовместной.

Определение 2.2. *r -подсистема* — подсистема ранга r и мощности r .

Определение 2.3. *Нерасширяемая совместная подсистема* — такая совместная подсистема системы S , что присоединение к ней любого, не входящего в нее, неравенства из системы S делает ее несовместной.

Будем считать, то рассматривается система P , являющаяся совместной и имеющая ранг r отличный от нуля.

Определение 2.4. *Узловое решение системы P* — такое решение системы P , что оно обращает в равенства какие-нибудь r ее неравенств с линейно-независимыми левыми частями.

Определение 2.5. *Узловая подсистема системы P* — такая подсистема системы P , что ее ранг равен числу неравенств в ней и все ее узловые решения удовлетворяют системе P .

В [4] доказана следующая теорема

Теорема 2.1. Каждая совместная система линейных неравенств вида S отличного от нуля ранга r имеет хотя бы одну узловую подсистему, а значит, хотя бы одно узловое решение. При этом каждая узловая подсистема является r -подсистемой.

В случае же несовместных систем верно следующее:

Лемма 2.1. Пусть несовместная система S имеет ранг $r > 0$. Тогда ранг любой нерасширяемой совместной подсистемы системы S равен r .

Лемма 2.2. Пусть система S имеет ранг $r > 0$. Тогда каждая нерасширяемая совместная подсистема системы S имеет хотя бы одну узловую r -подсистему.

Следовательно, нам необходимо осуществлять не полный перебор, а перебор всех подсистем мощности и ранга r . Для каждой подсистемы необходимо найти узловое решение. Узловое решение находится при замене всех знаков неравенства на равенства и решении стандартной задачи СЛАУ. Найденное узловое решение подсистемы надо подставить во все неравенства системы S и выделить из нее те неравенства, которым это решение удовлетворяет. Эти неравенства образуют совместную подсистему. Перебрав таким образом все r -подсистемы, получим некоторое множество совместных подсистем, среди которых будут все нерасширяемые. Максимальная по мощности среди выделенных подсистем — это МСП системы S . Описанный метод сочетает выделение МСП с одновременным нахождением одного ее решения, что и является конечной целью задачи оптимизации. Как показано, этот метод является точным, что приводит к тому, что его можно применять только к задачам малой размерности, в силу NP-трудности данной задачи. В других случаях перебор будет слишком долгим.

2.2 Релаксационный метод

Рассмотрим теперь приближенный метод поиска МСП в исходной системе. Нормализуем матрицу A , т.е. $\|a^i\| = 1$. Предположим, что исходная система совместна. Обозначим за $I(\mathbf{x}^k) = \{i : f_i(\mathbf{x}^k) < 0, i = 1, 2, \dots, m\}$. Пусть также есть фиксированный параметр $0 < \theta < 2$. Так же обозначим за M множество решений системы. Зададим произвольное начальное приближение x^1 и будем строить последовательность $x^1, x^2, \dots, x^k, \dots$ следующим образом:

$$x^{k+1} = x^k + \theta d(x^k) \frac{w(x^k)}{\|w(x^k)\|^2}, I(x_k) \neq \emptyset$$

где

$$d(x^k) = \sum_{i \in I(x^k)} (f_i(x^k))^2, w(x^k) = \sum_{i \in I(x^k)} |f_i(x^k)| a^i$$

В случае, когда система совместна, релаксационный процесс обладает свойствами того, что:

- $\|x^{k+1} - x\| < \|x^k - x\|, \forall x \in M$
- $\rho(x^{k+1}, M) < \sigma \rho(x^k, M)$, где $0 < \sigma < 1$, ρ - некоторая метрика.

Рассмотрим также другой вариант выбора шага. Мы хотим выбрать *оптимальный гарантированный шаг* [5]. Если зафиксировать $\theta = 1$ и представить

$$d(x^k) = \frac{\sum_{i \in J(x^k)} \lambda_i |f_i(x^k)|}{\|w(x^k)\|}, w(x^k) = \sum_{i \in J(x^k)} \lambda_i a^i$$

Где $J(\mathbf{x}^k) = \{i : f_i(\mathbf{x}^k) \leq 0, i = 1, 2, \dots, n\}$, а $\lambda_i \geq 0, i = 1, 2, \dots, n$ — фиксированные, при этом есть хотя бы один не нулевой. Тогда мы можем получить улучшенную оценку на сходимость.

$$\|x^{k+1} - x\|^2 \leq \|x^k - x\|^2 - d^2(x^k), \forall x \in M$$

Приведенные выше оценки показывают сходимость в случае совместной системы. Если же система не является совместной, то релаксационный процесс не сойдется. Как показано в [6], чем больше мера несовместности системы, тем больше область блуждания релаксационного процесса. В данном случае можно поступить следующим образом - начинать убирать неравенства из системы, наиболее далекие от предполагаемого решения [7].

Положим для $i = 1, 2, \dots, n$, и S — заданного параметра:

$$T^i = \sum_{k=1}^S \mathbb{I}[i \in I(x^k)]$$

В случае, если $I(x^k) = \emptyset$, то заканчиваем процедуру, вином - отбрасываем неравенство с наибольшим значением T^i . Отметим, что можно брать другие формулы для T , имеющие смысл «наиболее далеких от решения» неравенств.

Релаксационный метод очень прост идейно, но является достаточно затратным по ресурсам на практике.

2.3 Метод аппроксимации сигмоидами

Представим исходную задачу в другом виде. Зададим функцию Хевисайда

$$\theta(x) = \mathbb{I}[x \geq 0] = \begin{cases} 1, & x \geq 0, \\ 0, & x < 0 \end{cases}$$

Тогда мы можем выписать функционал, который мы максимизируем решая задачу нахождения МСП.

$$L(x) = \sum_{i=1}^m \theta(f_i(x)) \rightarrow \max_x$$

Идея данного метода состоит в аппроксимации этого функционала дифференцируемыми функциями. Функция Хевисайда представляет собой индикатор - разрывную в 0 функцию. Мы хотим приблизить ее с помощью сигмоиды.

$$\hat{\theta}(x) = \frac{1}{1 + \exp(-\alpha x)}$$

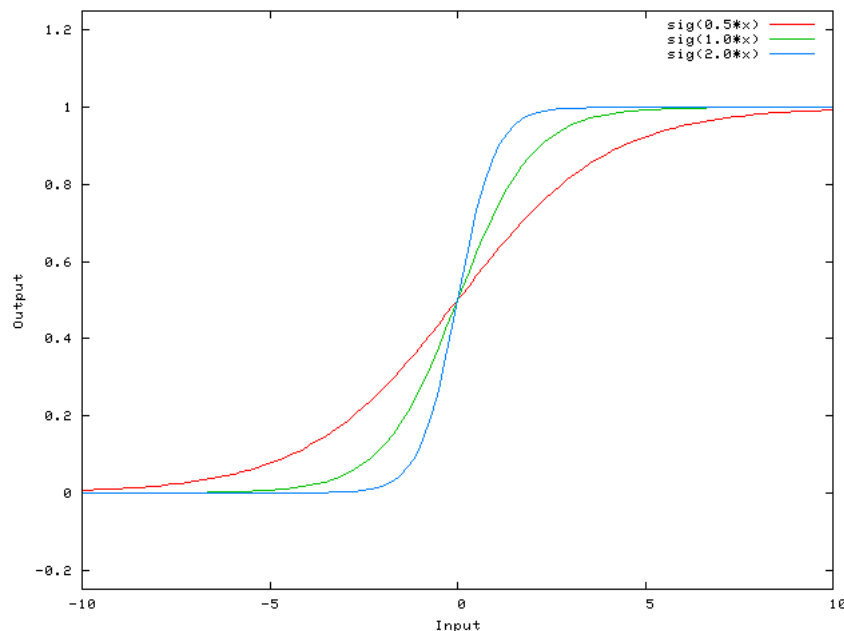


Рис. 1: Графики функции сигмоиды в зависимости от параметра

Где α - внешний параметр модели. Сигмоида, в отличие от индикатора, обладает гладкостью, что позволяет производить оптимизацию. Параметр α позволяет регулировать степень сглаженности, при малом параметре численная оптимизация быстрее сходится, но при этом исходный функционал аппроксимируется неточно, при больших значениях индикатор аппроксимируется точнее, но сходимость хуже. Поэтому важным вопросом является использование и изменение этого параметра.

Тогда аппроксимированный функционал будет иметь следующий вид:

$$\hat{L}_\alpha(x) = \sum_{i=1}^m \hat{\theta}(f_i(x)) \rightarrow \max_x$$

Для максимизации этого функционала уже можно пользоваться методами, основанными на градиентах, так как он дифференцируем. Но при сильном сглаживании функция будет с большей вероятностью уходить в локальный, а не глобальный максимум.

2.4 Модификация метода аппроксимации сигмоидами

При использовании предыдущего метода мы задаем гиперпараметр системы - α . Хотелось бы так же использовать то, что этот параметр отвечает за сглаженность исходного функционала. Предложим следующую модификацию исходного метода, основанную на эвристике.

Запустим алгоритм поиска МСП с помощью метода аппроксимации сигмоидами с разными параметрами α . Будем запускать численную максимизацию $\hat{L}_\alpha(x)$, по шагам,

$$x_i = \operatorname{argmax}[\hat{L}_{\alpha_i}(x)]$$

с начальным приближением x_{i-1} , с набором $(\alpha_0, \alpha_1, \dots, \alpha_n)$. Максимальное из найденных значений будет результатом работы алгоритма. Идея эвристики в том, что сначала мы находим оптимальное значение с большим значением параметра α , после же уменьшаем его, позволяя в окрестности найденного x находить более оптимальное решение. Для построения набора α в ходе экспериментов показала себя эффективной следующая процедура. Найдем максимальное значение мощности системы при запуске алгоритма поиска МСП с помощью метода аппроксимации сигмоидами и различными θ . Обозначим $\hat{\theta}$ - соответствующее максимальной мощности. Тогда возьмем набор для предлагаемого метода -

$$\left(\hat{\theta}, \frac{\hat{\theta}}{b}, \frac{\hat{\theta}}{b^2}, \frac{\hat{\theta}}{b^3}\right)$$

- уменьшение по геометрической прогрессии с показателем $\frac{1}{b}$. В качестве b возьмем 3.

3 Эксперименты

3.1 Дизайн экспериментов

Для задачи поиска МСП актуальным является вопрос того, как проводить исследование. В работе предложено несколько подходов для генерации данных, которые позволяют сравнивать приведенные выше алгоритмы.

Первый подход применим для малых размерностей. Зададим матрицу A , вектор b , как заполненные случайными числами. При малом числе m является возможным применение комбинаторного алгоритма поиска МСП, что является точным решением системы. В данном случае у нас есть точное решение и методы сравнимы не только между собой, но и с точным ответом. В случае, если комбинаторный алгоритм применять нельзя из-за ограничений по времени - данный подход тоже можно будет использовать. Однако мы получим только сравнительные оценки работы алгоритмов и не будем знать, насколько они далеки от точного ответа.

Второй подход заключается в том, что можно сгенерировать случайную матрицу A , после этого получить $b_i a_{i1} x_1 + a_{i2} x_2 + \dots + a_{im} x_m$. В этом случае исходная система будет совместной и содержать в точности одно решение. После чего можно выбрать долю $0 < p < 1$, и у случайной части неравенств уменьшить значение b_i как $\hat{b}_i = b_i - \varepsilon$, $\varepsilon > 0$ - случайное маленькое число. В данном случае нет гарантии, что МСП не может содержать больше, чем $(1 - p)n$ неравенств, но при этом она точно не меньше, чем это число.

Третий подход заключается в том, что бы сгенерировать матрицу таким же образом, как и во втором подходе. После этого необходимо зафиксировать параметр $\delta > 0$. Обозначим $k = \delta n$. Тогда сгенерируем дополнительно случайную матрицу $C \in \mathbb{R}^{k \times m}$, случайный вектор $\hat{b} \in \mathbb{R}^k$ и случайный вектор $\varepsilon_i > 0, i = 1, 2, \dots, n$. После чего запишем следующую систему неравенств.

$$\left\{ \begin{array}{l} f_1(x) = a_{11}x_1 + a_{12}x_2 + \dots + a_{1m}x_m + b_1 + \varepsilon_1 \geq 0, \\ f_2(x) = a_{21}x_1 + a_{22}x_2 + \dots + a_{2m}x_m + b_2 + \varepsilon_2 \geq 0, \\ \dots \\ f_n(x) = a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nm}x_m + b_n + \varepsilon_n \geq 0. \\ f_{n+1}(x) = c_{11}x_1 + c_{12}x_2 + \dots + c_{1m}x_m + \hat{b}_1 \geq 0 \\ f_{n+2}(x) = c_{21}x_1 + c_{22}x_2 + \dots + c_{2m}x_m + \hat{b}_2 \geq 0 \\ \dots \\ f_{n+k}(x) = c_{k1}x_1 + c_{k2}x_2 + \dots + c_{km}x_m + \hat{b}_k \geq 0 \end{array} \right.$$

МСП данной системы содержит как минимум n неравенств, при этом задается область, а не точка, как в предыдущем подходе. При этом можно изменять параметр ε , меняя меру несовместности системы.

В дальнейшем будем проводить эксперименты и получать оценки на время выполнения и размер определяемой подсистемы через методы Монте-Карло.

3.2 Эксперименты на данных малой размерности

Проведем сравнение методов на данных малой размерности. Зафиксируем $m = 3$. Будем варьировать n от 25 до 200. В первом эксперименте будем учитывать комбинаторный метод, для сравнения по времени работы. Будем пользоваться третьим подходом построения системы с $\varepsilon = 0.3$. Приведем сравнение времени работы.

N	Комбинаторный метод	Релаксационный метод	Метод аппроксимации сигмоидами
25	2.85	0.005	0.005
50	53.74	0.012	0.006
75	268.78	0.019	0.008
100		0.027	0.009
150		0.047	0.009
200		0.058	0.010

Таблица 1: Время работы алгоритмов в с.

Из таблицы видно что соотношение времени работы методов существенно отличается. Комбинаторный метод слабо применим при $N = 100$. В дальнейшем на данных малой размерности не будут проводиться замеры времени.

Приведем среднее значение мощности найденной МСП.

N	Комбинаторный метод	Релаксационный метод	Метод аппроксимации сигмоидами
25	27	20.03	21.2
50	56	43.2	42.6
75	77	61.5	69.1
100		84.0	84.8
150		120.8	120.0
200		178.3	156.1

Таблица 2: Среднее значение мощности системы

Отличительной чертой сравнения методов в данной задаче является то, что больше - значит лучше, так как каждый метод выдает решение, которое обращает в верные неравенства необходимое число подсистем. В данных экспериментах заметно, что релаксационный и метод аппроксимации сигмоидами выдают приближенные решения. Причем метод аппроксимации работает иногда лучше, чем релаксационный, иногда хуже, что обусловлено локальными максимумами сглаженной функции качества.

Сравним теперь методы на случайно сгенерированной матрице по подходу 2.

N	Комбинаторный метод	Релаксационный метод	Метод аппроксимации сигмоидами
25	17	13	15
50	38	23	29
75	52	35	42
100		44	55
150		67	82
200		88	107

Таблица 3: Среднее значение мощности системы по подходу 2.

В данной задаче алгоритм аппроксимации уже заметно выигрывает у алгоритма релаксации. И наконец проведем эксперименты на случайно сгенерированной матрице по подходу 1.

N	Комбинаторный метод	Релаксационный метод	Метод аппроксимации сигмоидами
25	18	13	16
50	34	25	30
75	49	37	44
100		50	57
150		83	89
200		109	120

Таблица 4: Среднее значение мощности системы по подходу 1

В данной ситуации релаксационный метод снова оказался хуже. Можно сделать предположение, что чем больше «случайность» системы и ее мера «несовместности», тем хуже работает релаксация.

3.3 Эксперименты на данных малой размерности

При переходе к данным большей размерности не имеет смысла рассматривать комбинаторный метод, в силу того, что время его подсчета слишком велико. Остановимся на сравнении релаксационного и аппроксимирующего методов поиска МСП. Будем пользоваться подходом 2, что бы иметь нижнюю оценку на число совместных подсистем.

Рассмотрим для начала, как будут вести себя методы при фиксированном числе неравенств(500) и при возрастающем m .

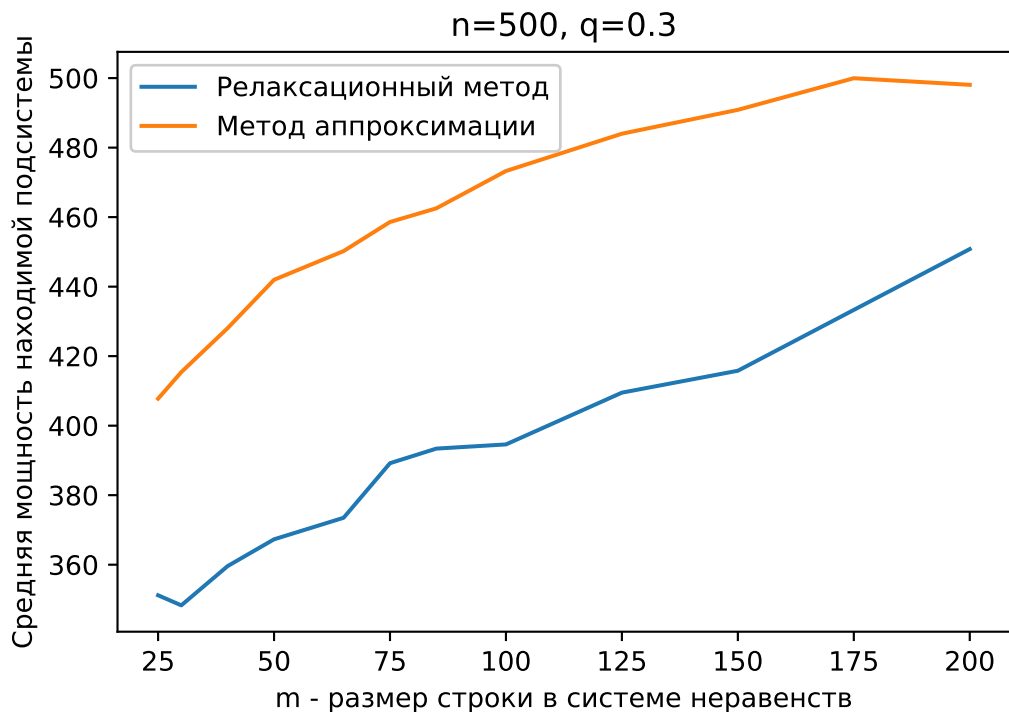


Рис. 2: Средняя мощность обнаруживаемой МСП

В данной ситуации метод релаксации показывает уже значительно худшие результаты, по сравнению с методом аппроксимации сигмоидами. Следует отметить, что рост мощности МСП, которую находят алгоритмы, связан в первую очередь с увеличением «степени свободы» из-за увеличения размерности вектора λ . Из-за этого в точной МСП может содержаться больше, чем 500 неравенств. Время работы алгоритмов, в отличие от комбинаторного метода, практически не зависит от размерности, что обусловлено векторными вычислениями.

Теперь зафиксируем параметр m и будем увеличивать число неравенств.

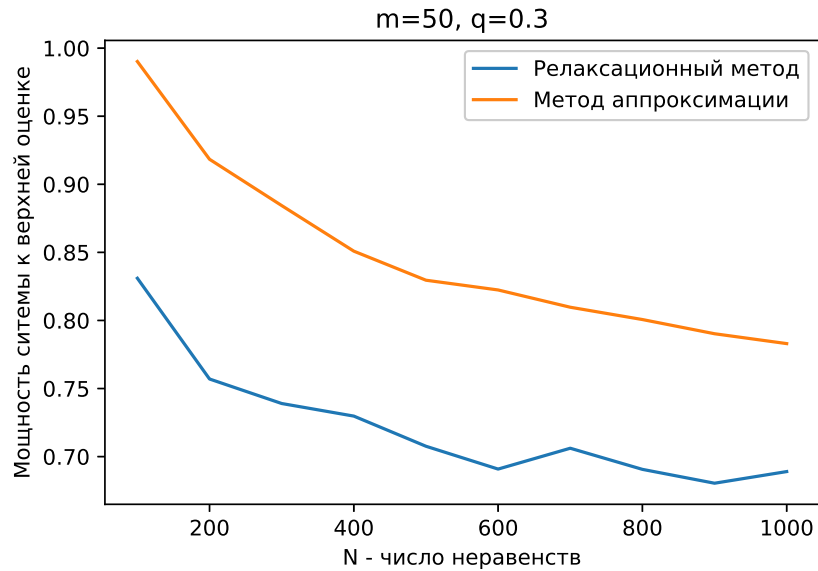


Рис. 3: Средняя мощность обнаруживаемой МСП в отношении к верхней оценке

На этом графике интересно то, что при росте числа неравенств, уменьшается и мощность приближенно находимой МСП, в отношении к верхней оценке, которая гарантирована нам алгоритмом. При этом релаксационный алгоритм продолжает проигрывать аппроксимирующему.

Так же исследуем время работы методов поиска МСП.

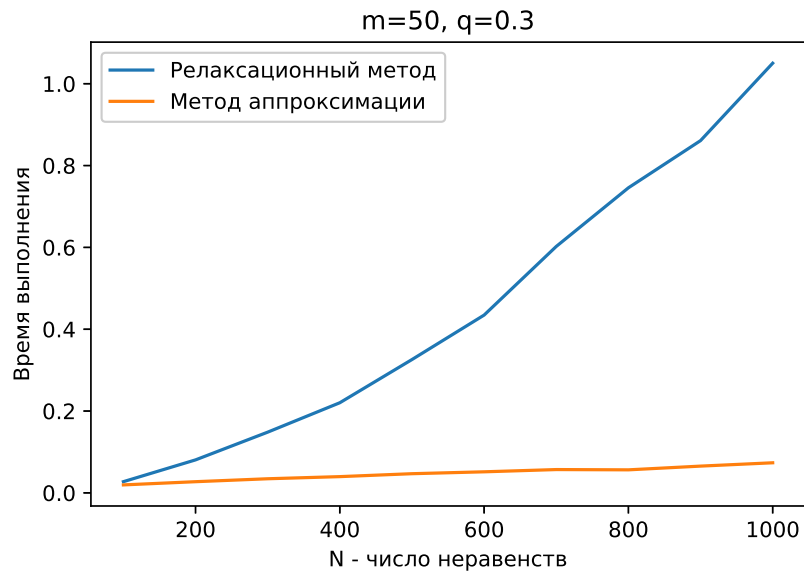


Рис. 4: Среднее время работы методов поиска МСП

Релаксационный метод растет практически линейно по времени выполнения. В дальнейшем перейдем к рассмотрению Аппроксимирующего метода более детально.

3.4 Анализ параметризации аппроксимирующего метода

В предыдущих экспериментах не затрагивался вопрос выбора «крутизны» сигмоиды в аппроксимирующем алгоритме. Из теоретических соображений более «пологие» сигмоиды должны быть достаточно удобны для оптимизации, но при этом решение, которое они находят - будет далеким от настоящего. «Крутые» же сигмоиды позволяют находить глобальный минимум более точно, но при этом они менее удобны для оптимизации. В любом случае у задачи много локальных оптимумов. Зафиксируем размер системы - $m = 1000, n = 30$. Воспользуемся подходом 3 для генерации системы.

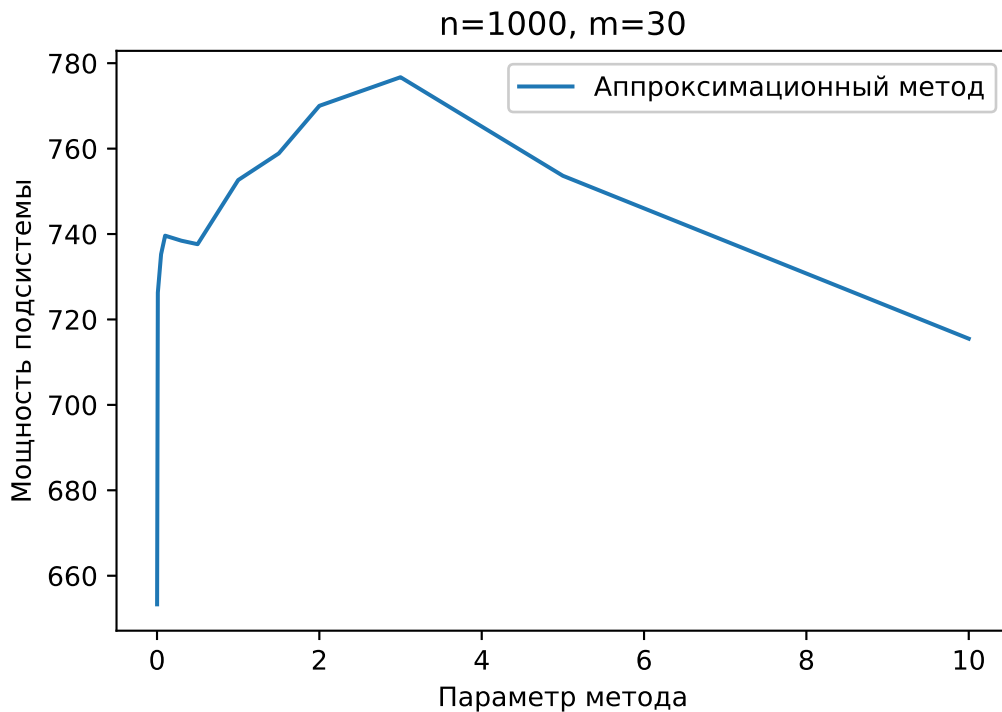


Рис. 5: Средняя мощность обнаруживаемой МСП в зависимости от θ

Как видно, функция имеет оптимальное значение, при этом уменьшение или увеличение от него дает ухудшение в качестве. Это связано с тем, что при параметре меньшем, чем оптимальный, процесс находит более локальный максимум, при больших же методу труднее сойтись.

Также интересным вопросом является размерность системы, при которой применим метод. Исследуем время выполнения, в зависимости от параметра N .

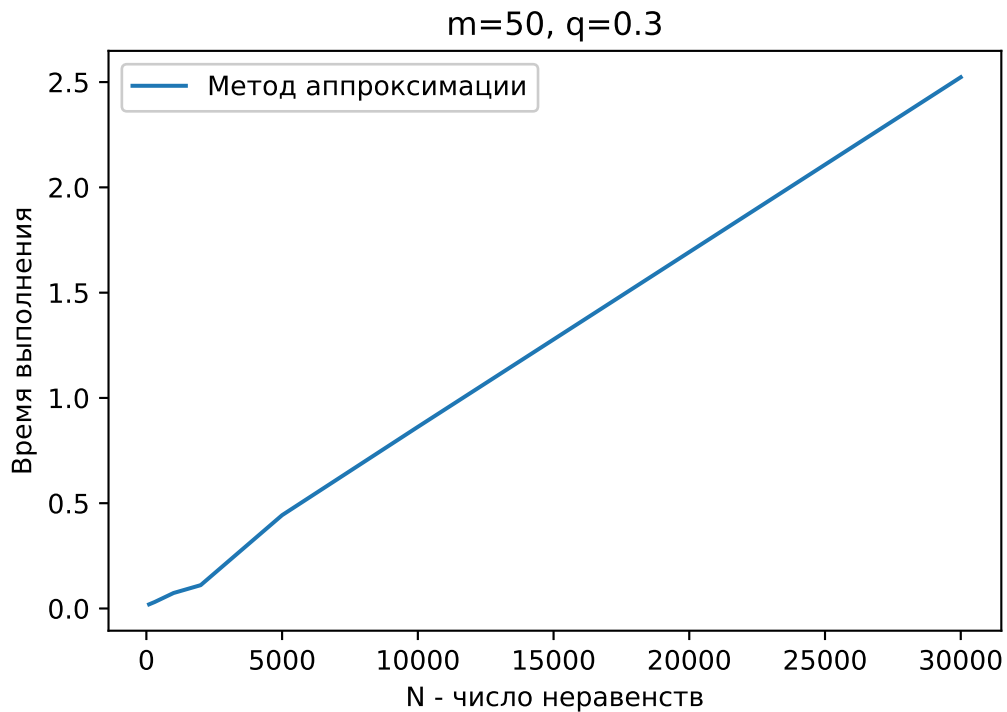


Рис. 6: Время работы метода

Данный график так же подтверждает линейный рост метода.

Интересную картину представляет из себя график, построенный по размеру МСП в отношении к нижней оценке на ее размер.

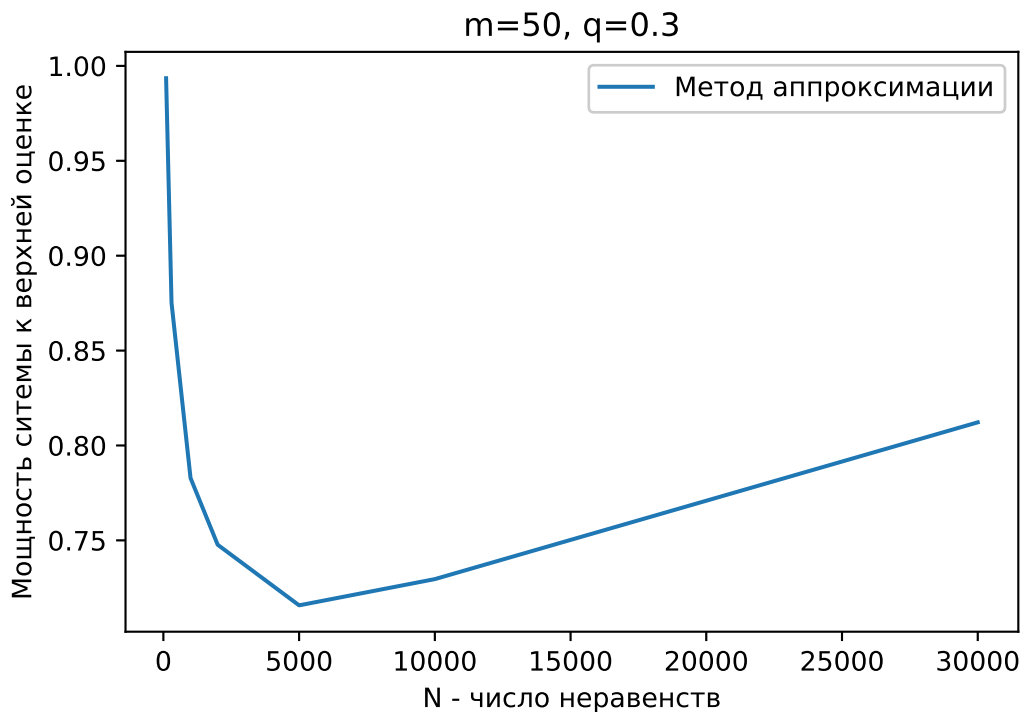


Рис. 7: Мощность системы в отношении к верхней оценке

При изначальном увеличении N точность падает, после же начинает возрастать. Это может быть объяснено тем, что с увеличением числа неравенств растет число «хороших» глобальных оптимумов.

3.5 Анализ предлагаемой модификации алгоритма

В рамках экспериментов был проведен запуск на системах разного размера. Применение модификации алгоритма давало небольшое, но стабильное увеличение мощности находимой МСП. При этом результат хуже, чем у исходного алгоритма получен быть не может, при одинаковом начальном приближении, учитывая, что он является первым шагом для модифицированного. Ниже приведена таблица сравнения средней мощности МСП для исходного и модифицированного алгоритмов при различных N, m . Для исходного алгоритма выбирался максимальное значение для перебираемых α .

N, m	Исходный метод	Модифицированный метод
1000, 30	771	804
500, 60	434	463
500, 30	404	426
300, 10	236	241

Таблица 5: Среднее значение мощности системы для исходного и модифицированного алгоритмов.

Как видно из таблицы, метод работает тем лучше, чем больше параметр m системы.

4 Заключение

В данной работе были программно реализованы и исследованы три метода для решения задачи нахождения МСП несовместной системы неравенств.

- **Комбинаторный метод** — является точным методом поиска решения задачи нахождения МСП. Применяется только в задачах с малой размерностью, в силу NP-трудности задачи. Для задач с большим числом неравенств или параметров в одном неравенстве работает слишком долго.
- **Релаксационный подход** — итерационный метод поиска МСП, является простым в реализации, при этом достаточно затратен по времени.
- **Метод аппроксимации сигмоидами** — метод, аппроксимирующий функционал, который мы хотим оптимизировать, заменяя индикаторы выполнения неравенства на сигмоиды. Дает достаточно хороший результат, но зависит от параметризации.

Последний метод в целом показывает хорошие результаты. При этом следует отметить, что результаты запуска сильно зависят от структуры данных. Параметризация достаточно сильно влияет на метод аппроксимации, при этом скорость его работы значительно выше, чем у остальных, за счет сведения к стандартной задаче оптимизации.

В работе был предложен метод поиска МСП, модифицирующий метод аппроксимации сигмоидами. Этот метод основан на итерационном изменении параметра α и

построении цепочки приближений. Данный метод всегда дает результат не хуже, чем исходный, но при этом почти всегда дает результат лучше, чем метод аппроксимации сигмоидами.

В работе было предложено несколько методов генерации случайных данных для экспериментов, позволяющие проводить сравнение приближенных методов поиска МСП, обладая нижней оценкой на мощность МСП в точном решении.

Список литературы

- [1] Катериночкина Н. Н. Методы выделения максимальной совместной подсистемы системы линейных неравенств. Сообщение по прикладной математике. — Москва: Вычислительный центр РАН, 1997.
- [2] Amaldi E., Kann V. The complexity and approximability of finding maximum feasible subsystems of linear relations. // *Theoret. Comput. Sci.* — 1995. — Vol. 147, no. 1-2. — Pp. 181–210. citeseer.ist.psu.edu/freund99short.html.
- [3] Журавлёв Ю. И. Об алгебраическом подходе к решению задач распознавания или классификации // *Проблемы кибернетики*. — 1978. — Т. 33. — С. 5–68. www.ccas.ru/frc/papers/zhuravlev78prob33.pdf.
- [4] Черников С. Н. Линейные неравенства. — М.: Наука., 1968.
- [5] Рязанов В. В. О выборе релаксационных алгоритмов решения систем линейных неравенств. // *Сб. работ по математической кибернетике. Вып. 2.* — 1977. — С. 171–192.
- [6] И.И.Еремин. О системах неравенств с выпуклыми функциями в левых частях. // *Известия АН СССР, серия математическая.* — 1966. — № 30. — С. 265–278.
- [7] Журавлёв Ю. И., Рязанов В. В., Сенько О. В. «Распознавание». Математические методы. Программная система. Практические применения. — М.: Фазис, 2006.