МАТЕМАТИКА

УДК 517.977.58

М. В. Крючков, Пермский филиал Национального исследовательского университета «Высшая школа экономики»

СРАВНИТЕЛЬНЫЙ АНАЛИЗ НЕКОТОРЫХ АЛГОРИТМОВ РЕШЕНИЯ МНОГОМЕРНОЙ ЗАДАЧИ УСЛОВНОЙ ОПТИМИЗАЦИИ

Изложены результаты решения многомерной задачи условной оптимизации четырьмя методами: классическим градиентным спуском, точным аналитическим решением системы уравнений, генетическим алгоритмом, а также авторским методом «подтягивание к среднему». Необходимость постановки данной задачи возникает при решении систем линейных алгебраических уравнений (СЛАУ), построении регрессионных моделей, обучении искусственных нейронных сетей (ИНС) и др. Для тестовой оптимизационной задачи последовательно применялся каждый из вышеперечисленных методов, что позволило провести сравнительный анализ и выявить преимущества и недостатки используемых алгоритмов.

Ключевые слова: условная оптимизация, математическое программирование, нейронная сеть.

Рассмотрим функционал

$$E=\sum_{k=1}^{m}\left(\sum_{i=1}^{n}(w_i x_i^k)-y^k\right)^2, \qquad (1)$$

где x_i^k, y^k – числовые значения; w_i – управляющие параметры. Необходимость его минимизации возникает при постановке математических задач различного класса: уменьшение нормы вектора невязки при решении системы линейных алгебраических уравнений, нахождение коэффициентов линейной регрессионной модели и при обучении искусственных нейронных сетей [1]. В данной статье уделим особое внимание последней задаче.

Введем дополнительное условие

$$\begin{gathered}\forall\, a,b\in\{x_i^k, y^k\}\;\; i=\overline{1,n}\;\; k=\overline{1,m}:\\ |a-b|\le 0,4\cdot\max(a,b)\end{gathered} \qquad (2)$$

и ограничение

$$0,8\le w_i\le 1,2 \;\; i=\overline{1,n} \qquad (3)$$

и прокомментируем их, придав смысловую окраску. Условие (2) означает, что числовые значения как входных, так и выходных данных отличаются друг от друга не более чем на 40 %, а ограничение (3) – отклонение значения каждого исходного параметра при влиянии на итоговый результат не превышает 20 %. Такая постановка применима для решения задачи прогнозирования на основе только эмпирических наблюдений исследуемой величины: краткосрочный прогноз рынка ценных бумаг, курсов валют, спортивный прогноз [2] и др. [3].

Перепишем исходную задачу в матричной форме: X_i^k – матрица входных данных размерности $m\times n$; $\vec{w}=(w_1,w_2,...,w_n)^T$ – вектор-столбец управляющих параметров; $\vec{y}=(y_1,y_2,...,y_m)^T$ – вектор-столбец выходных данных. Обозначим $\vec{z}=X\cdot\vec{w}$, тогда компоненты вектора $\vec{w}^*$, соответствующего минимальной норме вектора $\vec{y}-\vec{z}$, будут доставлять минимум функционалу (1). В каждом из тестовых примеров будем находить такой вектор $\vec{w}^0$, при котором значения функционала (1) не будет превышать заранее заданного порогового значения. При решении задачи условной оптимизации [4] будем проводить сравнительный анализ четырех используемых методов: метод градиентного спуска, решение СЛАУ, генетический алгоритм (ГА), а также собственный метод, именуемый «подтягивание к среднему». Кратко опишем основные особенности предложенных методов для решения поставленной задачи.

1. Метод градиентного спуска [4]. Поскольку особое внимание в данной статье посвящено задаче обучения ИНС, выберем постоянный шаг движения в сторону антиградиента, так как в традиционном алгоритме обратного распространения ошибки аналогом данного множителя является параметр, называемый скоростью обучения сети [1]. В качестве начального приближения был выбран вектор $\vec{w}_1=(1,1,...,1)^T$, а направление антиградиента для выполнения дальнейшего движения каждый раз вычислялось по формуле $\vec{g}_i=-\dfrac{\partial E}{\partial w_i}$. Для завершения итерационного процесса требовалось достижение функционалом (1) порогового значения либо превышение критического числа шагов.

2. Решение СЛАУ [5] фактически предполагает совпадение векторов $\vec{y}$ и $\vec{z}$, т. е. совместность системы $X\cdot\vec{w}=\vec{y}$. Для простоты реализации в качестве одного из тестовых примеров была выбрана квадрат-

Получено 24.01.14

ная матрица X_i^k с отличным от нуля определителем, а исходная система решена по формулам Крамера.

3. Эволюционный (генетический) алгоритм предполагает наличие некоторой «популяции решений», к которой последовательно применяются операции селекции, скрещивания и мутации [6]. Приведем основные цифры, применяемые в нашей работе: число популяций в каждом поколении – 100, при селекции 20 худших из них заменяется на потомков, полученных в ходе скрещиваний 1-го и 11-го, 2-го и 12-го и т. д. до 10-го и 20-го. При скрещивании особи обмениваются случайным набором генов (равновероятное значение от 3 до 8), причем один из которых «мутирует» – изменяется не более чем на 1 % с вероятностью $p = 0{,}2$. Итерационный процесс завершается, когда вектор генотипа «лучшей» особи доставит функционалу (1) значение ниже заданного порогового уровня либо произойдет превышение критического числа поколений.

4. Для применения алгоритма «подтягивание к среднему» примем допущение, что для минимизации функционала (1) необходимо минимизировать, а в идеале устремить к нулю каждую из компонент вектора $\vec{y}-\vec{z}$. Такое допущение особенно уместно при решении задачи обучения ИНС, поскольку на пространстве данных обучающей выборки выходные сигналы нейросети должны быть близки к априорным значениям. Для демонстрации идеи алгоритма рассмотрим k-ю компоненту вектора $\vec{y}-\vec{z}$:

$$x_1^k w_1 + x_2^k w_2 + \ldots + x_n^k w_n - y^k. \qquad (4)$$

Устремление данной компоненты к нулю предлагается провести посредством того, что каждое слагаемое $x_i^k w_i$ приблизить к значению y^k / n, т. е. «усреднить» за счет коэффициента w_i, а именно увеличить w_i, если $x_i^k w_i < y^k / n$, и уменьшить в случае $x_i^k w_i > y^k / n$. Немаловажно отметить, что порядок прохождения по компонентам вектора отклонений осуществляется от первого к последнему. В этом случае последние компоненты оказывают большее влияние на изменение вектора решений $\vec{w}^0$. Такой подход разумен при решении задачи прогнозирования, так как сведения последних наблюдений гораздо ценнее данных, полученных в самом начале измерений (пример: при прогнозировании курса доллара в первую неделю декабря тренд за ноябрь более показателен, чем тренд за январь, хотя и тот, в свою очередь, влияет на итоговый прогноз).

Для создания тестовых примеров генерировался случайный вектор $\vec{w}^э$, компоненты которого удовлетворяют условию (3), матрица X_i^k, для элементов которой справедливы ограничения (2), и вычислялся вектор-столбец $\vec{y} = X \cdot \vec{w}$. Для полученных значений X_i^k и $\vec{y}$ находилось субоптимальное значение $\vec{w}^0$ по каждому из четырех вышеперечисленных методов. Для удобства сравнительного анализа результаты работы алгоритмов приведены в таблицах (табл. 1 и 2).

Таблица 1. **Результаты для $n = m = 10$**

Алгоритм	E	Число итераций
Антиградиент	0,007	119 012
Решение СЛАУ	10^(–18)	
ГА	0,004	30
Подтягивание к среднему	0,216	8 081

Таблица 2. **Результаты для $n = 10$, $m = 20$**

Алгоритм	E	Число итераций
Антиградиент	0,007	295 618
Решение СЛАУ		
ГА	0,011	41
Подтягивание к среднему	0,042	10 595

Лучше всего с первым тестовым примером справился ГА, однако, несмотря на малое число итераций, не стоит забывать, что для применения операции «селекции» требуется частичная сортировка массива из 100 «особей», что является весьма затратным по времени. Метод СЛАУ хоть и доставил минимально возможное (с учетом погрешности машинных вычислений) значение функционалу E, но 2 компоненты вектора $\vec{w}^0$ вышли за пределы ограничений (3), что недопустимо для решения поставленной задачи. Метод антиградиента «выиграл» у метода «подтягивание к среднему» по минимизации отклонения, зато сильно уступил ему в числе итераций и в сложности реализации.

Для второй тестовой задачи метод решения СЛАУ оказался полностью непригоден ввиду несовместности системы $X \cdot \vec{w} = \vec{y}$, а решив ее лишь для некоторого набора входных данных, получим вектор $\vec{w}^0$, не удовлетворяющий условию (3) и с большим значением функционала E. Хорошо решил поставленную задачу ГА, однако он потребовал много «творческих экспериментов» по ручной настройке параметров основных операций и оказался слишком затратным по времени работы. Алгоритм «подтягивание к среднему» дал большее значение E по сравнению с антиградиентом, но существенно «победил» по количеству итераций и простоте реализации.

Подводя итоги сравнительного анализа, хотелось бы сделать два важных вывода: для решения задачи минимизации (1) при условии (2) и ограничении (3) с лучшей стороны себя проявил ГА; для решения задачи обучения ИНС, следствием из которой является исходная постановка, метод «подтягивание к среднему» дает результаты, существенно не уступающие по точности традиционным, а в некоторых случаях даже лучше, поскольку в него «зашиты» механизмы, учитывающие ценность наблюдений, полученных непосредственно перед прогнозом.

Библиографические ссылки

1. *Калацкая Л. В.*, *Новиков В. А.*, *Садов В. С.* Организация и обучение искусственных нейронных сетей : экспериментальное учеб. пособие. – Минск : Изд-во БГУ, 2003.
2. *Крючков М. В.* Построение нейросетевой модели для решения задачи спортивного прогнозирования // Вестник ИжГТУ. – 2013. – № 4(60). – С. 159–161.
3. Курс социально-экономической статистики : учебник / М. Г. Назаров [и др.]. – Изд. 9-е. – М. : Омега-Л, 2011.
4. *Акулич И. П.* Специальные задачи линейного программирования. – М. : Высш. шк., 1986.
5. *Ильин В. А.*, *Позняк Э. Г.* Линейная алгебра : учебник для вузов. – Изд. 6-е, стер. – М. : Физматлит, 2004.
6. *Гладков Л. А.*, *Курейчик В. В.*, *Курейчик В. М.* Генетические алгоритмы : учеб. пособие. – Изд. 2-е. – М. : Физматлит, 2006.

M. V. Kryuchkov, National Research University “Higher School of Economics”, Perm branch

Comparative Analysis of Some Algorithms for Solving Problems of Multidimensional Constrained Optimization

The paper presents the results of solving problem of a multidimensional constrained optimization by four methods: classical gradient descent, accurate analytical solution of the system, genetic algorithm, and own method “pulling up the middle”. The need for setting this problem arises when solving systems of linear algebraic equations, building regression models, training of artificial neural networks, etc. For the test optimization problem each of the above methods was consistently applied, which allowed to perform a comparative analysis and identify the advantages and disadvantages of the used algorithms.

Key words: constrained optimization, mathematical programming, neural network.

УДК 004.042

Я. М. Далингер, кандидат технических наук, доцент, Санкт-Петербургский государственный университет гражданской авиации

МОДЕЛЬ СИСТЕМЫ ОБРАБОТКИ С ТИРАЖИРОВАНИЕМ ПОСТУПАЮЩИХ СООБЩЕНИЙ

Приводятся результаты исследования системы обработки сообщений с их тиражированием при обработке. Получены формулы для вычисления характеристик системы. Результаты могут быть использованы при разработке и анализе подобных систем различного назначения.

Ключевые слова: поток сообщений, алгоритм тиражирования, системы массового обслуживания, обработка информации, математическое моделирование.

Во многих системах обработки информации, особенно с иерархической структурой, наблюдается эффект тиражирования, когда одно поступающее для обслуживания сообщение преобразуется в несколько копий (тиражируется), каждая из которых самостоятельно обрабатывается. В результате чего после обработки одного сообщения в системе появляется несколько новых. Такой эффект свойственен, например, иерархическим системам обработки документов при движении документов в системе сверху вниз.

Алгоритм обработки копий определяется спецификой системы. Здесь представлены результаты исследования алгоритма обработки при тиражировании, когда поступившее сообщение образует заданное число копий, каждая из которых обрабатывается как отдельное сообщение.

Анализ всей иерархической системы с тиражированием возможен только при проведении анализа работы отдельных обслуживающих устройств, входящих в состав системы. Поэтому далее приводятся результаты анализа одного узла обработки, где наблюдается эффект тиражирования.

Описание узла обработки

В статье, в качестве узла обработки сообщений, исследуется система, состоящая из одного обслуживающего устройства, на вход которого поступает N взаимно независимых пуассоновских потоков сообщений. Интенсивность потока j – λ_j ($0<\lambda_j<\infty$; $j=1, 2, ..., N$).

Для каждого сообщения потока j задана константа тиражирования R_j ($1 \le R_j < \infty$; $j=1, 2, ..., N$), определяющая количество создаваемых копий тиражируемого сообщения.

Длительность тиражирования считаем равной нулю, то есть все копии сообщения появляются одновременно.

В данном случае все копии сообщения обрабатываются на обслуживающем устройстве параллельно. Новое сообщение может поступить для обслуживания только после окончания обработки всех копий предыдущего.

Считаем, что все поступающие сообщения образуют общую очередь и обслуживаются в порядке поступления. Длительность ожидания в очереди и длина очереди не ограничены.

Получено 21.03.14