

МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ АРХИТЕКТУРЫ НЕЙРОСЕТЕВЫХ КЛАССИФИКАТОРОВ

Альберт Воронин, Юрий Зиатдинов, Анна Антонюк

Аннотация. Рассматривается постановка задачи и процедура векторной оптимизации архитектуры нейросетевого классификатора. В качестве целевой функции предложена скалярная свертка критериев по нелинейной схеме компромиссов. Используются поисковые методы оптимизации с дискретными аргументами. Приведен пример – нейросетевой классификатор текстов.

Ключевые слова: многокритериальная оптимизация, нейронные сети, классификатор.

ACM Classification Keywords: H.1 Models and Principles – H.1.1 – Systems and Information Theory; H.4.2 – Types of Systems; C.1.3 Other Architecture Styles – Neural nets

Conference: The paper is selected from International Conference "Classification, Forecasting, Data Mining" CFDM 2009, Varna, Bulgaria, June-July 2009

Содержание проблемы

Важной разновидностью искусственных нейронных сетей являются нейросетевые классификаторы. Они применяются для технической и медицинской диагностики, классификации различного рода информационных источников и пр. В достаточно общем случае структура q -слойного нейросетевого классификатора с прямыми связями представлена на Рис. 1.

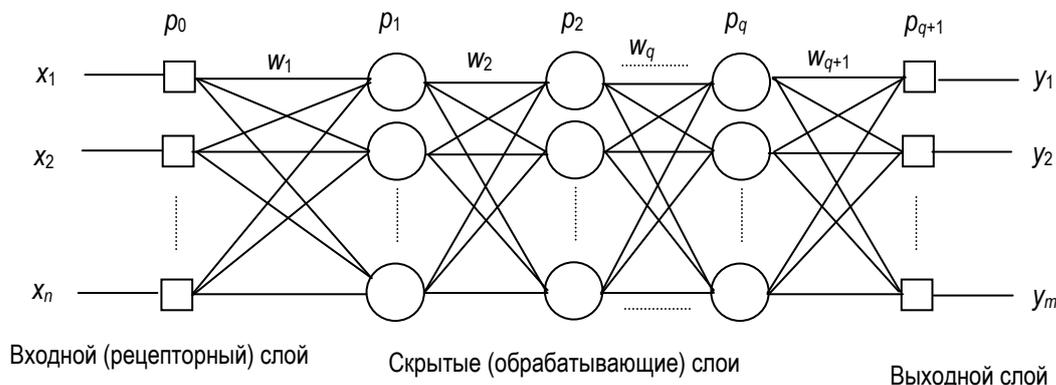


Рис.1

Здесь x_1, x_2, \dots, x_n – признаки объекта классификации, составляющие входной вектор $x = \{x_i\}_{i=1}^n$; $p_0 = n$ – число нейронных элементов в рецепторном слое; p_1, p_2, \dots, p_q – число нейронов в каждом из скрытых слоев; $p_{q+1} = m$ – число нейронов в выходном слое (количество классов);

$y = \{y_k\}_{k=1}^m$ – выходной вектор нейронной сети, определяющий принадлежность объекта классификации одному из m классов; $w_1, w_2, \dots, w_q, w_{q+1}$ – векторы синаптических весов нейронной сети.

Приведем необходимые сведения из теории нейронных сетей [1-3]. Искусственная нейронная сеть – это совокупность нейронных элементов и связей между ними. Каждый нейрон имеет группу синапсов – однонаправленных входных связей, соединенных с выходами других нейронов. Каждый синапс характеризуется величиной синаптической связи или ее весом w_i (определяется при обучении нейронной

сети). Нейрон имеет текущее состояние, определяемое как взвешенная сумма его входов: $s = \sum_{i=1}^n w_i x_i$.

Выход нейрона есть функция его состояния, которая называется функцией активации: $y = f(s)$. Сигнал возбуждения или торможения посредством аксона (выходная связь данного нейрона) поступает на синапсы следующих нейронов. Функции активации бывают пороговыми и непрерывными (биполярный сигмоид, гауссиан и пр). Множество всех нейронов искусственной нейронной сети разделяется на подмножества, называемые слоями. Слой – это множество нейронов, на которые в каждый такт времени параллельно поступают сигналы от других нейронов данной сети [2]. На выходе классификатора получается вектор функций активации $y = \{y_k\}_{k=1}^m$. Номер j , для которого выход y_j имеет максимальную активность, т.е. $\max_{k \in [1, m]} y_k = y_j$, соответствует номеру класса объекта классификации.

Количество нейронов входного слоя $p_0 = n$ определяется размерностью входного вектора признаков и не подлежит изменениям. Аналогично, количество нейронов выходного слоя $p_{q+1} = m$ определяется числом областей (классов), на которые делится пространство признаков и тоже является постоянным. Количество же обрабатывающих (скрытых) слоев q и число нейронов в каждом из них p_1, p_2, \dots, p_q составляют понятие **архитектуры** [1] нейронной сети и могут служить аргументами (независимыми переменными) при ее оптимизации.

В настоящей работе ограничим исследование случаем, когда число q является фиксированным и заданным. Тогда аргументами оптимизации архитектуры нейронного классификатора являются количества нейронов в каждом из обрабатывающих слоев, составляющие вектор независимых переменных $p = \{p_j\}_{j=1}^q$. От выбора архитектуры p зависит качество функционирования нейронного классификатора.

Проблема заключается в таком выборе архитектуры, при котором нейронный классификатор в заданных условиях функционирования характеризуется наилучшими свойствами.

Постановка задачи

В общем виде проблема может быть формально представлена задачей

$$p^* = \arg \operatorname{extr}_{p \in P} Y(p), \quad (1)$$

где $Y(p)$ – целевая функция; $\operatorname{extr}_{p \in P}$ – оператор экстремизации целевой функции по аргументам p ; P –

допустимая область независимых переменных.

Для конструктивного решения задачи сделаем дополнительные частные предположения. Каждому свойству нейронного классификатора поставим в соответствие количественную характеристику $f(p)$,

имеющую смысл критерия качества его функционирования. Одним из таких критериев является вероятность ошибки классификации. Будем определять этот критерий экспериментально и приближенно представим его как количество ошибок классификации $e(p)$, отнесенное к общему, достаточно большому количеству испытаний N :

$$f_1(p) = \frac{e(p)}{N}. \quad (2)$$

Предполагается, что с ростом в некоторых разумных пределах числа нейронов в обрабатывающих слоях точность классификации повышается, и величина этого критерия уменьшается. Предельно допустимое значение ошибки сети должно быть известно из физических соображений и задано как ограничение $f_1(p) \leq A_1$.

Второй критерий характеризует время, необходимое для обучения нейронной сети при данной архитектуре p . Наблюдается тесная корреляция между таким временем и суммарным количеством нейронов в скрытых слоях классификатора. Поэтому представим этот критерий в виде

$$f_2(p) = \sum_{k=1}^q p_k. \quad (3)$$

Отметим, что данным критерием характеризуется и время прохождения сигнала через нейронную сеть от входа к выходу. С ростом числа нейронов значение критерия увеличивается. Предельно допустимое значение второго критерия определяется допустимым временем обучения нейронной сети и задается как ограничение $f_2(p) \leq A_2$.

Существуют и другие критерии для характеристики различных свойств нейронного классификатора. В данной работе мы ограничимся только приведенными двумя основными критериями, имея в виду, что излагаемая методика допускает включение в рассмотрение и других свойств классификатора.

Допустимая область аргументов оптимизации задается параллелепипедным ограничением $P = \{p \mid 0 \leq p_k \leq P_u, k \in [1, P_u], u \in [1, q]\}$, где P_u – максимальное число нейронов в u -м слое.

Поскольку оба включенных в рассмотрение критерия подлежат минимизации (чем критерий меньше, тем лучше соответствующее свойство классификатора), то оператор экстремизации целевой функции приобретает вид: $extr = \min_{p \in P}$.

Таким образом, оба критерия являются противоречивыми, неотрицательными, минимизируемыми и ограниченными. Налицо все предпосылки для использования в качестве целевой функции скалярной свертки критериев по нелинейной схеме компромиссов [4]. Такая свертка в унифицированной версии выражается формулой

$$Y(p) = Y[f(p)] = \frac{A_1}{A_1 - f_1(p)} + \frac{A_2}{A_2 - f_2(p)}. \quad (4)$$

где $f(p) = \{f_r(p)\}_{r=1}^2$ – двумерный вектор частных критериев. Учитывая (2), (3) и (4), выражение (1) для задачи оптимизации архитектуры нейронного классификатора преобразуется к виду

$$p^* = \arg \min_{p \in P} \left[\frac{A_1}{A_1 - e(p)/N} + \frac{A_2}{A_2 - \sum_{k=1}^q p_k} \right]. \quad (5)$$

Нетрудно видеть, что в формуле (5) зависимость $e(p)$ априори является неизвестной и подлежащей экспериментальному определению.

Метод решения

Среди задач многокритериальной оптимизации имеются такие, аргументы которых по своей физической природе могут принимать только дискретные значения. Специальной нормировкой дискретные значения обычно всегда могут быть сведены к целочисленным. Такие задачи значительно сложнее непрерывных многокритериальных задач и для их решения должны применяться иные подходы [5].

Множество допустимых дискретных значений может быть бесконечным, конечным или даже состоящим всего из двух значений, например, 0 и 1. В первом случае задача вырождается в непрерывную задачу оптимизации. Для ее решения в [4] предложено эффективное и формализованное алгоритмическое и программное обеспечение. В последнем случае имеет место целочисленное программирование с булевыми переменными со своими специфическими методами (логический синтез конечных автоматов, функции Рвачева и пр.). С нашей точки зрения наиболее интересен и содержателен случай, когда множество допустимых дискретных значений не настолько велико, чтобы задача вырождалась в непрерывную, но и не настолько мало, чтобы ее можно было решить простым перебором. Именно такой является задача (5) – задача нелинейного дискретного (целочисленного) программирования.

Методы дискретного программирования не обладают таким единством, как методы вариационного исчисления, и в большинстве представляют собой набор частных приемов, пригодных для решения частных задач. Но их актуальность требует их развития и совершенствования, т.к. наиболее важные прикладные задачи сводятся, как правило, к задачам частично или полностью дискретного программирования. Сложность решения задач дискретного (целочисленного) программирования возрастает в том случае, когда задача является многокритериальной.

В том случае, когда компоненты возможных решений многокритериальных задач могут принимать только дискретные значения $p_k^{(P_u)}, k \in [1, P_u], u \in [1, q]$, скалярная свертка по нелинейной схеме компромиссов $Y(p)$ является *решетчатой* функцией, заданной на дискретном множестве P . Оптимизация решетчатой целевой функции, построенной по нелинейной схеме компромиссов, сводится к задаче нелинейного программирования с дискретными (целочисленными) аргументами, решение которой, как отмечено выше, достаточно сложно.

Для решения этой проблемы мы предполагаем, что при дискретном множестве P существует вспомогательная область непрерывных аргументов $p_c \in P_c$, которая содержит все дискретные точки $p_k^{(P_u)}$ и всё непрерывное пространство между ними. В области P_c определена непрерывная функция $Y(p_c)$, которая в точках $p_k^{(P_u)}$ совпадает с решетчатой функцией $Y(p)$.

Это предположение позволяет получить аналитическое решение, если в выражении (5) зависимость $e(p)$ задана, например, в виде регрессионной модели. Тогда можно воспользоваться необходимым

условием минимума функции: $\frac{\partial Y(p_c)}{\partial p_c} = 0$. Решение этой системы уравнений дает компромиссно-

оптимальную непрерывную точку p_c^* . Последний шаг алгоритма – поиск на P ближайшей к p_c^* возможной дискретной точки, которая и будет искомым дискретным решением p^* . В нашем случае, к сожалению, задание аналитических зависимостей весьма затруднительно или вообще невозможно.

Мы рассматриваем как основной случай, когда функции $e(p)$ и, следовательно $Y(p)$, неизвестны, но есть возможность определять значения функции $Y(p)$ в точках $p_k^{(P_u)}$ измерением или вычислением.

Тогда можно организовать натуральный или вычислительный эксперимент, в результате которого осуществляется поисковое движение к искомой дискретной компромиссно-оптимальной точке p^* .

Возможны различные подходы к организации поисковой процедуры, которая должна давать последовательность улучшающихся решений. Один из них – дискретный аналог метода симплекс-планирования Нелдера-Мида (метод деформируемого многогранника) [4]. Это – разновидность градиентных методов, весьма часто и успешно применяющихся на практике. Второй – нелокальный (дуальный) подход [4], часто более эффективный, чем градиентные методы.

Так как в поисковых процедурах используются локальные или нелокальные модели непрерывной функции $Y(p_c)$, то общей для названных вариантов является необходимость поиска на P возможной дискретной точки p_d , ближайшей к непрерывному решению p_c на текущей или заключительной итерации. Если число скрытых слоев q невелико, то решение этой задачи не вызывает затруднений (простое округление до целого). При многослойных классификаторах мы рекомендуем использовать следующий алгоритмический прием. В точке p_c помещается центр гиперсферы, диаметр которой возрастает от нуля до тех пор, пока поверхность сферы не коснется ближайшей дискретной точки, которая тем самым идентифицируется как p_d . Возможны разные программные реализации этого алгоритма.

Известны нейросетевые классификаторы различного вида и назначения.

Многокритериальная оптимизация нейросетевого классификатора текстов

В качестве примера рассмотрим в общих чертах задачу многокритериальной оптимизации архитектуры нейросетевого классификатора текстов. Система текстовой классификации [3] состоит из двух основных частей: частотный анализатор с системным словарем и собственно нейросетевой классификатор (Рис.2).



Рис.2

На вход системы поступает текст, на выходе получается номер темы, к которой относится этот текст (бизнес, политика, медицина, спорт, просто спам и т.п.).

Прежде чем приступить к оптимизации архитектуры нейросетевого классификатора, необходимо выполнить следующие этапы:

1. Определяются m классов, с которыми будет работать система.
2. Подбираются соответствующие учебные тексты $t_k, k \in [1, m]$ и проверочные (тестовые) тексты $t_l, l \in [1, L], L \geq m$.
3. Из множества учебных текстов специальным образом выделяются слова $v_i, i \in [1, n]$ и формируется системный словарь V .
4. Частотный анализатор определяет для каждого слова v_i из системного словаря V его частоту вхождения x_i в данный текст t_k . Частотная характеристика – это вектор $x = \{x_i\}_{i=1}^n$ признаков текста t_k , размерность которого равна количеству слов в системном словаре $v_i \in V$.

Получив результаты частотного анализа учебных текстов, можно приступить к обучению нейронного классификатора при некоторой архитектуре $p = \{p_j\}_{j=1}^q$. Процесс обучения нейронной сети заключается в установлении таких весовых коэффициентов ее связей $w_1, w_2, \dots, w_q, w_{q+1}$, при которых максимальная ошибка сети на учебных текстах для данной архитектуры не превышает предельно допустимое значение. Конкретные алгоритмы обучения здесь не рассматриваются.

Теперь можно приступить непосредственно к процедуре векторной оптимизации. Для оптимизации архитектуры нейросетевого классификатора воспользуемся поисковым методом симплекс-планирования. Пусть для определенности число обрабатываемых слоев $q = 2$. Тогда идею метода в непрерывном варианте можно иллюстрировать при помощи Рис.3.

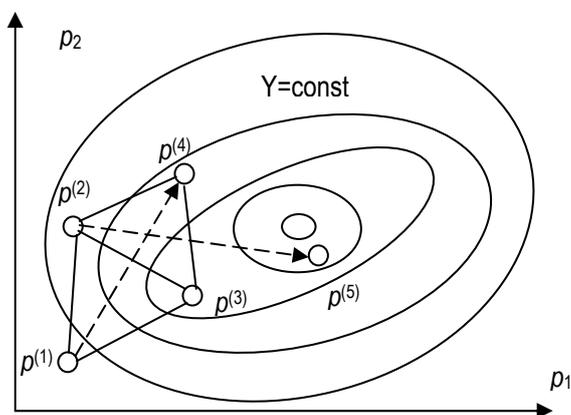


Рис.3

На плоскости аргументов $p_1 - p_2$ в некоторой стартовой области строим исходный регулярный симплекс, который в двумерном случае представляет собой равнобедренный треугольник с вершинами $p^{(1)}, p^{(2)}, p^{(3)}$. Для каждой из трех архитектур симплекса осуществляем процесс обучения классификатора и подаем на вход серию тестовых текстов t_l . В каждой вершине симплекса определяем количество ошибок классификации $e^{(1)}, e^{(2)}, e^{(3)}$ при общем количестве испытаний $N = L$. По формуле

(2) получаем критерии $f_1^{(1)}, f_1^{(2)}, f_1^{(3)}$. По формуле (3) определяются критерии $f_2^{(1)}, f_2^{(2)}, f_2^{(3)}$. Формула (4), выступающая в этом случае в роли не целевой, а оценочной функции, для нашего примера имеет вид

$$Y(p_1, p_2) = \frac{A_1}{A_1 - e(p_1, p_2)/L} + \frac{A_2}{A_2 - p_1 - p_2}. \quad (6)$$

Для архитектур стартового симплекса она дает значения скалярных сверток $Y^{(1)}, Y^{(2)}, Y^{(3)}$. Сравнивая между собой эти значения, находим, что одно из них, например, $Y^{(1)}$, оказалось больше (т.е. хуже), чем другие. С большой вероятностью можно утверждать, что архитектура $p^{(4)}$, полученная зеркальным отображением худшей в исходном симплексе точки $p^{(1)}$ относительно центра противоположной грани, окажется лучше. Осуществив все расчеты для архитектуры $p^{(4)}$, образуем новый симплекс с вершинами $p^{(2)}$, $p^{(3)}$ и $p^{(4)}$. Сравнив значения $Y^{(2)}, Y^{(3)}, Y^{(4)}$, обнаружим, что одна из точек, например, $p^{(2)}$, оказалась хуже других в смысле второго симплекса. Отобразив эту точку относительно центра противоположной грани второго симплекса, получим архитектуру $p^{(5)}$, и т.д. до тех пор, пока мы получим архитектуру p^* , соответствующую минимуму целевой функции.

Это лишь иллюстрация идеи метода симплекс-планирования. На самом деле этот метод в модификации Нелдера-Мида предусматривает адаптацию симплексов к топографии целевой функции за счет деформации многогранников, он имеет хорошо разработанное алгоритмическое и программное обеспечение. Кроме того, нельзя забывать, что у нас имеет место случай оптимизации с целочисленными аргументами, что диктует необходимость для каждого полученного непрерывного решения p_c искать ближайшее дискретное решение p_d .

Второй, нелокальный поисковый метод несколько сложнее в реализации, но он обычно более эффективен [4,5]. Метод основан на итерационном построении «плывущей» вместе с системой изменяющихся базисных точек уточняющейся по результатам эксперимента нелокальной модели $Y(p)$, причем совокупность опорных точек сжимается и стягивается к точке искомого экстремума («шагреновая кожа»). На каждой итерации одновременно и взаимозависимо осуществляется как уточнение наших представлений о целевой функции в области экстремума, так и определение такой оценки аргументов экстремума, которая адекватна уровню этих представлений на данной итерации. Поэтому нелокальный метод оптимизации относится к классу дуальных и может быть назван методом дуального программирования.

Оба поисковых метода предусматривают проведение серии экспериментов. Полученные при этом экспериментальные данные могут быть использованы для построения аналитических регрессионных моделей частного критерия $f_1(p) = e(p)/L$. С помощью таких моделей можно осуществлять не поисковую, а аналитическую векторную оптимизацию архитектуры других нейросетевых классификаторов такого же вида. Если это окажется сложным, то проводится поисковая процедура, но уже с применением не натурального, а *вычислительного* эксперимента, что существенно проще.

Решая задачу построения регрессионных моделей, мы должны задать вид аппроксимирующей зависимости, известной с точностью до коэффициентов регрессии. Анализ задачи приводит к предположению, что с достаточной для практики точностью можно ограничиться линейной регрессией:

$$f_1(p_1, p_2) \approx (a_1 p_1 + a_2 p_2) / L, \quad (7)$$

где a_1, a_2 – коэффициенты регрессии, определяемые по экспериментальным данным методом наименьших квадратов. Линейная регрессионная модель проверяется на адекватность методами математической статистики. При необходимости модель может быть усложнена.

Рассмотренные методы предусматривают старт поисковой процедуры от архитектуры, которая, по мнению разработчика, находится достаточно близко к оптимальной точке. Если в процессе поиска имеет место возрастание числа нейронов в обрабатывающих слоях, то теория нейронных сетей [1] характеризует данный подход как *конструктивный*. При избыточном стартовом количестве нейронов подход именуется *деструктивным* (принцип Родена: чтобы изваять скульптуру, нужно взять целую глыбу мрамора и удалить из нее лишнее).

Осуществление изложенных в работе этапов векторной оптимизации позволяет получить архитектуру нейросетевого классификатора, при которой системно увязываются противоречивые критерии эффективности его функционирования.

Благодарности

Статья частично финансирована из проекта **ITHEA XXI** Института Информационных теории и Приложений FOI ITHEA и Консорциума FOI Bulgaria (www.ithea.org, www.foibg.com).

Библиография

1. Бодянский Е.В., Руденко О.Г. Искусственные нейронные сети: архитектуры, обучение и применение. – Харьков: ТЕЛЕТЕХ, 2004. – 372 с.
2. Головкин В.А. Нейронные сети: обучение, организация и применение. – М.: ИПРЖР, 2001. – 256 с.
3. Борисов В.С. Самообучающийся классификатор текстов на естественном языке // Кибернетика и системный анализ – 2007. – №3. – С.169-176.
4. Воронин А.Н., Зиатдинов Ю.К., Козлов А.И. Векторная оптимизация динамических систем. – Киев: Техніка, 1999. – 284 с.
5. Воронин А.Н., Мосорин П.Д., Ясинский А.Г. Многокритериальные задачи оптимизации с дискретными аргументами // Автоматика-2000. Міжнародна конференція з автоматичного управління: Праці. –Т.1. – Львів: ДНДІІІ, 2000. – С. 75-78.

Сведения об авторах

Воронин Альберт Николаевич – профессор, доктор технических наук, профессор кафедры компьютерных информационных технологий Национального авиационного университета, проспект Комарова, 1, Киев-58, 03058 Украина; e-mail: alnv@voliacable.com

Зиатдинов Юрий Кашафович – профессор, доктор технических наук, заведующий кафедрой компьютерных информационных технологий Национального авиационного университета, проспект Комарова, 1, Киев-58, 03058 Украина; e-mail: oberst@nau.edu.ua

Антонюк Анна Александровна – аспирант Национального авиационного университета, проспект Комарова, 1, Киев-58, 03058 Украина, e-mail niuriel@mail.ru