Ученые записки Таврического национального университета им. В.И. Вернадского

Серия "Математика. Механика. Информатика и Кибернетика" N 2 (2002) 133 – 136

УДК 681:612.82

Г.А. Махина

АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ РЕШЕНИЯ ЗАДАЧИ СЛАБООПРЕДЕЛЕННОЙ ОПТИМИЗАЦИИ С ЛИНЕЙНЫМИ ОГРАНИЧЕНИЯМИ

ВВЕДЕНИЕ

В работе [1] В.И. Донской определяет некоторые основные типы задач оптимизации с неполными данными. В данной статье рассматривается вопрос построения нейронной сети для задач, относящихся к типу *B1* или задач с заданной системой ограничений, но с частично заданной целевой функцией. Требуется определить вектор $\bar{x} \in \Omega$ (Ω -- заданная область допустимых решений), доставляющий экстремальное значение целевой функции $f(\bar{x})$, которая существует, но не задана явно. Известен лишь набор пар $\left\{(\bar{x}_j, f_j = f(\bar{x}_j)), j = \overline{1,q}\right\}$ называемых прецедентами. Мы ограничиваемся рассмотрением топологии нейронной сети для частного случая ограничений, заданных в виде линейных неравенств, и решаем задачу условной оптимизации следующего вида

$$\max_{\bar{x}} f(\bar{x}), \quad A\bar{x} \le \bar{b}, \quad f : \left\{(\bar{x}_j, f_j = f(\bar{x}_j), j = \overline{1,q}\right\}. \tag{1}$$

Здесь A - матрица ограничений размерности $m \times n$, $\bar{b}$ -- m-мерный вектор. Вопрос построения нейронной сети для решения задачи слабоопределенной оптимизации с прецедентной информацией как о целевой функции, так и об области допустимых решений рассматривался в работе [2]. В данной статье предполагается, что на первом этапе целевая функция восстанавливается так же, как это делалось в [2] после чего полученная нейронная сеть расширяется за счет введения в нее дополнительных элементов, отвечающих за линейные ограничения типа неравенств. Для введения ограничений используется метод барьерных функций.

1. Описание метода логарифмических барьерных функций

Метод логарифмических барьерных функций служит основанием архитектуры нейронной сети, реализующей задачу слабо-определенной оптимизации с линейными ограничениями. Пусть $\bar{w}$ обозначает вектор дополнительных переменных $w = \bar{b} - A\bar{x}$ Также введем коэффициент "усиления" $g(t) > 0$, являющийся скалярной функцией времени. Под g будем подразумевать функцию $g(t)$, взятую в конкретный момент времени. Определим расширенную целевую функцию

$$F_g(\bar{x}) = f(\bar{x}) + \frac{1}{g}\sum_{i=1}^{m} \log(w_i(\bar{x})), \tag{2}$$

добавив к исходной целевой функции штрафное слагаемое $(1/g)\sum_{i=1}^{m}\log(w_i(\bar{x}))$. Тогда соответствующая пополненная оптимизационная задача принимает вид

$$\max_{\bar{x}} F_g(\bar{x}) \tag{3}$$

При $g \to \infty$ пополненная задача приближает исходную. Построенная таким образом функция F_g для всех $\bar{x}$, находящихся вне области допустимых решений, принимает отрицательные бесконечные значения. Отметим, что, если F_g является строго вогнутой функцией внутри области допустимых значений, то тогда внутри этой области существует единственной решение задачи (3), известное как аналитический центр, соответствующий g.

Пусть $\bar{a}_i$ — i-й столбец матрицы A^T. Поле градиента функции F_g в точке $\bar{x}$ — это

$$\nabla F_g(\bar{x}) = \nabla f(\bar{x}) - \frac{1}{g}\sum_{i=1}^{m}\frac{\bar{a}_i}{\bar{w}_i(\bar{x})} \tag{4}$$

взвешенная сумма $m+1$ вектора. Интуитивно понятно, что первое слагаемое направляет градиент пополненной функции в сторону градиента исходной целевой функции, второе же барьерное слагаемое имеет направление $-\bar{a}_i$, прочь от соответствующей ограничивающей гиперплоскости $\{\bar{x}|\bar{a}_i^T\bar{x} = \bar{b}_i\}$. Степень влияния i-го барьера обратно пропорциональна его расстоянию $\bar{w}_i(\bar{x}) > 0$ от точки $\bar{x}$.

Пусть коэффициент усиления $g(t)$ является растущей функцией времени. Тогда решение связанной с ним пополненной задачи (3) сходится (в случае отсутствия локальных минимумов!) к оптимальному решению (1). Траектория решения является гладкой, аналитической кривой, и ее часто называют центральной траекторией. При вычислении начального расширенного решения и приближенном вычислении центральной траектории на компьютере, как правило, используются методы второго и высшего порядков (напр., метод Ньютона). И, хотя система уравнений является плохо обусловленной вблизи границы области допустимых решений, с достаточными предосторожностями решение (3) может быть найдено. Методы первого порядка почти не используются, поскольку не могут достаточно быстро реагировать на быстрые изменения функции и поэтому размер дискретного шага приходится делать достаточно малым, чтобы не попасть в самый центр сингулярности.

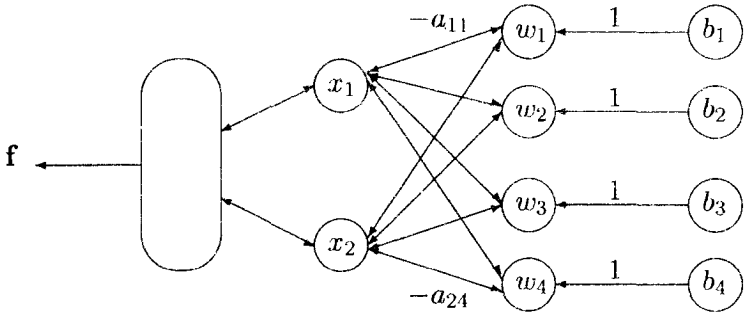
Первоначально метод барьерных функций был спроектирован для решения нелинейных оптимизационных задач с ограничениями и был основан на идее, предложенной Кэрроллом С. Позже он был модифицирован Тэмом Д. [4] для решения задач линейного программирования с линейными ограничениями. Также Тэмом Д. была предложена архитектура так называемой ЛП-сети, являющейся нейронной сетью решения задачи линейного программирования с линейными ограничениями.

2. Нейронная сеть задачи слабоопределенной оптимизации с линейными ограничениями

Нейронная сеть задачи слабоопределенной оптимизации с линейными ограничениями имитирует динамическую систему первого порядка, использующую метод барьерных функций: воображаемая точка без массы движется в изменяющемся

со временем поле градиента пополненной барьерными функциями целевой функции $F_g(t)$. Эволюцией точки управляет система дифференциальных уравнений. Для определения вектора $\bar{w}'(\bar{x},t)$ используется вектор дополнительных переменных $\bar{w}(\bar{x}) = \bar{b} - A\bar{x} > 0$. При этом он определяется как $w_i'(\bar{x},t) = [g(t)w_i(\bar{x})]^{-1}$. Градиент функции $F_g(t)$ вычисляется как $\nabla F_g(\bar{x}) = \nabla f(\bar{x}) - A^T\bar{w}'(\bar{x},t)$. Для нахождения градиента функции $f(\bar{x})$, которая представляет собой восстановленную в виде нейронной сети типа многослойный перцептрон целевую функцию, используется правило цепи, напоминающее алгоритм обратного распространения ошибки. Правило нахождения градиента нейроподобной функции в направлении входного вектора состоит из четырех основных шагов и описано в работе [2].

Следующий рисунок представляет собой архитектуру нейронной сети, предназначенной для решения задачи с двумя переменными и четырьмя ограничениями. На первом этапе по исходной прецедентной информации строится блок восстановленной целевой функции, который представляет собой многослойную нейронную сеть с прямыми связями. На рисунке данный блок представлен чисто символически, и его более подробное описание можно увидеть в работе [2]. На следующем этапе построенная нейронная сеть пополняется новыми компонентами: вводятся слои, соответствующие вектору дополнительных переменных $\bar{w}$ и вектору ограничений $\bar{b}$. На рисунке данные слои представлены в блоке линейных ограничений.



Для функционирования сети необходимо иметь хотя бы одну точку, принадлежащую области допустимых решений. Поскольку, в соответствии с постановкой задачи, все точки обучающего множества принадлежат этой области, то данное условие не будет вызывать сложностей.

Предположим, что $\bar{x}^0$ является известной строго внутренней допустимой точкой $(A\bar{x}^0 < \bar{b})$ Полную систему дифференциальных уравнений, описывающих данную динамическую систему, можно записать в следующем виде. Нейроны x_j инициализируются как $x_j = x_j^0$ и управляются уравнением

$$\frac{dx_j}{dt} = \frac{df}{dx_j} - \sum_{i=1}^{m} A_{ij}w_i' \quad \text{для всех } j. \tag{5}$$

Вычисление $\frac{df}{dx_j}$ происходит в блоке восстановленной целевой функции и разбито на несколько этапов, описанных в [2].

Нейроны w_i' инициализируются так, чтобы удовлетворялось равенство $\bar{w} = \bar{b} - A\bar{x}^0$. Их динамика определяется уравнением

$$\frac{dw_i}{dt} = -w_i + b_i - \sum_{j=1}^{n} A_{ij}x_j \quad \text{для всех } i \tag{6}$$

и нелинейным преобразованием

$$w_i' = h(w_i, t).$$

В явном виде

$$h(w_i, t) = \frac{1}{g(t)w_i}.$$

Здесь $g(t)$ является монотонно растущей функцией времени (ее, например, можно определить как $g(t) = \max(t^0, t); \quad t^0 > 0$), величина τ является достаточно малой в сравнении с dg/dt. Нейроны b_i не изменяются и сохраняют постоянное значение. Дифференциальное уравнение получено, исходя из определения дополнительных переменных $\bar{w}(\bar{x}) = \bar{b} - A\bar{x} > 0$. При этом временная производная $\bar{w}(\bar{x})$ приравнивается отклонению $\bar{d}(t) = -\bar{w}(\bar{x}) + \bar{b} - A\bar{x}$. При стабилизации динамической системы должно быть достигнуто равенство $\bar{d}(t) = 0$.

Для всех нейронов b_i сила связей приравнивается единице. Нейроны w_i' и x_j полностью взаимосвязаны двунаправленными весовыми коэффициентами $-A_{ij}$. В данной системе не происходит никакого непосредственного обучения или адаптации весовых коэффициентов, хотя для вычисления производных $\frac{df}{dx_j}$ и используется правило цепи, сходное с методом обратного распространения. Параметры A_{ij}, b_j $(j = 1..n; i = 1..m)$ устанавливаются в нулевой момент времени и не изменяются на протяжении всей эпохи функционирования сети.

Список литературы

[1] В. И. Донской, А. И. Башта Дискретные модели принятия решений при неполной информации. Таврия, Симферополь (1992).

[2] В. Ф. Блыщик, В. И. Донской, А. Минин, Г. А. Махина, Интеллектуализированная программная система INTMAN поддержки принятия решений в задачах планирования и управления. — Искусственный интеллект, №2 (2002), с.245-257.

[3] Г. А. Махина. Нейросетевой подход к задачам слабоопределенной оптимизации. — Искусственный интеллект, №2 (2000), с.145-148.

[4] T.P. Caudell , K. Zikan. Neural Network Architecture for Linear Programming. Proc. On NNC, 1992. Vol 3, pp. 91-96.