Тема 5

Постановка и возможные пути решения задачи обучения нейронных сетей

Частичная задача обучения

Пусть у нас есть некоторая нейросеть N. В процессе функционирования эта нейронная сеть формирует выходной сигнал $\bar{y} \in Y$ в соответствии с входным сигналом $\bar{x} \in X$, реализуя некоторую функцию $g : X \to Y, \quad \bar{y} = g(\bar{x})$. Если архитектура сети задана, то есть, заданы количество и вид нейронов, а также структура связи между ними, то вид функции g определяется значениями синаптических весов, смещений сети и параметрами функций активации нейронов. Обозначим буквой G множество всех возможных функций g, соответствующих заданной архитектуре сети.

Пусть решение некоторой задачи - функция $r : X \to Y$. Рассмотрим случай, когда функция r задана парами входных-выходных векторов $(\bar{x}^1, \bar{y}^1), (\bar{x}^2, \bar{y}^2), \ldots, (\bar{x}^M, \bar{y}^M)$, для которых $\bar{y}^m = r(\bar{x}^m), \ (m = 1,2,\ldots,M)$.

Определение. Набор пар $(\bar{x}^1, \bar{y}^1), (\bar{x}^2, \bar{y}^2), \ldots, (\bar{x}^M, \bar{y}^M)$ таких, что $\bar{y}^m \in X, \bar{x}^m \in Y$ и $\bar{y}^m = r(\bar{x}^m)$, будем называть обучающей выборкой функции $r : X \to Y$.

Определим функцию ошибки $D_r : G \to R$. Эта функция, показывает для каждой из функций g степень близости к r. Функцию ошибки также часто называют целевой функцией обучения нейронной сети.

Пример 5.1. Пусть $g(x)$ и $r(x)$ непрерывные, интегрируемые функции определённые на отрезке $[a,b]$.

Тогда определим $D_r(g) = \int_a^b (g(t) - r(t))^2 dt$.

Пример 5.2. Пусть $g(\bar{x})$ - некоторая непрерывная на R^n функция, а $r(\bar{x})$ задана обучающей выборкой $(\bar{x}^i, \bar{y}^i), i = 1,2,...,M$. Тогда целевую функцию можно определить как

$$D_r(g) = \frac{1}{2}\sum_{i=1}^{M} (g(\bar{x}^i) - r(\bar{x}^i))^2 .$$

Решить поставленную задачу с помощью нейронной сети заданной архитектуры - это значит построить (синтезировать) функцию $g' \in G$, подобрав параметры нейронов (как

правило это синаптические веса и смещения) таким образом, чтобы функционал D обращался в минимум для всех пар $(\bar{x}^m, \bar{y}^m)$:

$$D_r(g') = \min_{g \in G} D_r(g)$$

Задача обучения определяется совокупностью трех элементов: $\langle N, r, D_r \rangle$, где

N – архитектура нейронной сети, определяющая множество G – множество функций $g : X \to Y$;

$r : X \to Y$ определяет желаемый результат обучения (часто определяется обучающей выборкой);

$D_r(g)$ - функция ошибки, показывающая для каждой сети N степень близости к r.

Необходимо найти функцию $g' \in G$ (реализуемую нейронной сетью N'), минимизирующую функционал *D:*

$$D_r(g') = \min_{g \in G} D_r(g).$$

Обучение – это, как правило, итерационная процедура. На каждой итерации происходит уменьшение функции ошибки. Обучение требует длительных вычислений.

В этой лекции мы рассматриваем только частичную задачу обучения, то есть задачу, в условиях которой однозначно задана архитектура сети. Дело в том, что если расширить исходные условия и предположить, что мы не знаем количества нейронов, их вида и структуры связей, то задача обучения переводится из класса задач отыскания экстремума линейно-параметризированного нелинейного отображения в класс задач многомерной нелинейной оптимизации с множеством решений.

Различают три основных вида стратегии обучения: «с учителем», «без учителя», смешанную. В первом случае, нейросеть настраивают некоторым алгоритмом по заданной обучающей выборке $(\bar{x}^m, \bar{y}^m)$ $(m = 1,2,\ldots,M)$. Во втором случае обучающая выборка содержит лишь входные значения для сети, то есть имеет вид $\{\bar{x}^m\}$ $(m = 1,2,\ldots,M)$, а сеть в процессе обучения настраивается в соответствии с некоторым правилом. В третьем случае часть параметров сети настраивается по заданной обучающей выборке, а другая – без использования знаний о правильных ответах.

Если выбраны множество обучающих примеров - пар $(\bar{x}^m, \bar{y}^m)$ $(m = 1,2,\ldots,M)$ - и способ вычисления функции ошибки *D*, обучение нейронной сети при априорно заданной архитектуре – это задача отыскания экстремума линейно-параметризированного нелинейного отображения. Размерность задачи зависит от вида функции r (количества

пар обучающей выборки) и архитектуры нейросети. Для сетей небольшой размерности (порядка нескольких сот нейронов) и количества пар обучающей выборки не более ста - количество итераций обучения может быть от нескольких тысяч до 10^8.

Функция D может иметь произвольный вид. Поэтому обучение в общем случае - многоэкстремальная невыпуклая задача оптимизации.

Для решения этой задачи могут быть использованы следующие алгоритмы:

- алгоритмы линейной регрессии (алгоритмы нулевого порядка),
- алгоритмы локальной оптимизации с вычислением частных производных первого порядка,
- алгоритмы локальной оптимизации с вычислением частных производных первого и второго порядка,
- алгоритмы прямого вычисления значений параметров сети по известным исходным данным,
- стохастические алгоритмы и алгоритмы глобальной оптимизации.

К первой группе относятся метод главных компонент, авторегрессия, схема «Гусеница», фильтр Калмана.

Ко второй группе относятся метод скорейшего спуска, методы тяжелого шарика, методы с одномерной и двумерной оптимизацией целевой функции в направлении антиградиента, метод сопряженных градиентов.

К третьей группе относятся метод Ньютона, методы оптимизации с разреженными матрицами Гессе, квазиньютоновские методы, метод Гаусса-Ньютона, метод Левенберга-Марквардта.

К четвертой группе относятся алгоритмы использующие методы решения систем линейных уравнений.

Стохастическими алгоритмами являются поиск в случайном направлении, имитация отжига, метод Монте-Карло (численный метод статистических испытаний), эволюционные (генетические) алгоритмы, а также алгоритмы перебора значений переменных, от которых зависит целевая функция.

Задача аппроксимации функции в стандартной постановке

Для некоторой функции $r(x)$, заданной обучающей выборкой $(\bar{x}^i, \bar{d}^i), i = 1,..., N$, необходимо найти вектор параметров $\bar{w}'$ такой, что НС реализующая функцию $\bar{y} = f(\bar{w}', \bar{x})$ наилучшим образом аппроксимирует функцию r, т. е. верно

$$D_r(f(\bar{w}',\bar{x})) = \sum_{n=1}^{N} e_n = \min_{\bar{w}} D_r(f(\bar{w},\bar{x})),$$

где e_n - ошибка НС на *n*-й паре выборки.

Как правило, для вычисления ошибки на одной паре применяют формулу

$$e_n = \frac{1}{2}\left\|\bar{d}^n - f(\bar{w},\bar{x}^n)\right\|^2.$$

Если функцию $f(\bar{w}',\bar{x})$ можно представить как суперпозицию функций $f_1, f_2, ..., f_l$,

$$\bar{y}^m = \sum_{j=1}^{l} w_j f_j(\bar{x}^m) = \bar{w}\cdot\bar{f},$$

то задача аппроксимации сводится к решению системы уравнений

$$\begin{pmatrix} f_1(\bar{x}^1) & \dots & f_l(\bar{x}^1) \\ \vdots & \ddots & \vdots \\ f_1(\bar{x}^M) & \cdots & f_l(\bar{x}^M) \end{pmatrix}\begin{pmatrix} w_1 \\ \vdots \\ w_l \end{pmatrix} = \begin{pmatrix} \bar{d}^1 \\ \vdots \\ \bar{d}^M \end{pmatrix},$$

или, в сокращенной записи,

$$F\cdot\bar{w} = \bar{d}.$$

Решение такой системы имеет вид

$$\bar{w} = (F^T F)^{-1} F^T \bar{d}.$$

Это решение существует при условии невырожденности матрицы $(F^T F)$. Кроме того, при росте количества элементов обучающей выборки, растёт размерность системы и значительно увеличиваются вычислительные затраты на решение этой системы. Поэтому, как правило, задача обучения решается методами, использующими стандартные алгоритмы оптимизации.