

ОБУЧЕНИЕ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ МЕТОДОМ ОБРАТНОГО РАСПРОСТРАНЕНИЯ ОШИБКИ

Попова Ю.Б., Яцынович С.В.

БНТУ, г. Минск, Беларусь, julia_popova@mail.ru

БНТУ, г. Минск, Беларусь, hawkrai@yandex.ru

ВВЕДЕНИЕ

В последние несколько лет наблюдается взрыв интереса к искусственным нейронным сетям (ИНС), которые успешно применяются в самых различных областях: бизнесе, медицине, технике, геологии, физике. Нейронные сети (НС) вошли в практику везде, где существует необходимость решения задач прогнозирования, классификации или управления.

Несмотря на то, что среди исследователей нет единого определения нейронной сети, большинство из них согласны, что это сеть простых обрабатывающих элементов (ОЭ), которые могут показать сложное глобальное поведение, определенное связями между обрабатывающими элементами и параметрами элемента. Идея нейронной сети пришла в технику после исследований центральной нервной системы: нейронов, аксонов, дендритов и синапсов, которые образуют один существенный обрабатывающий информацию элемент. В модели нейронной сети обрабатывающие элементы называют нейронами (также встречаются названия нейроды, neurodes или юниты). Описание нейрона приведено в [1].

Поскольку НС не должны быть адаптивными по существу, на практике вместе с ними используют алгоритмы, разработанные для изменения загруженности связей в сети, чтобы пропустить необходимый сигнал. Еще одно подобие биологическим нейронным сетям состоит в том, что функции выполняются юнитами все вместе, параллельно.

Для того чтобы создать искусственную нейронную сеть, необходимо потратить немало времени и пройти все этапы ее разработки от формирования условий работы сети до обучения смоделированных и связанных нейронов.

ЭТАПЫ РАЗРАБОТКИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Можно выделить 4 основных этапа в процессе разработки искусственной нейронной сети для решения прикладной задачи [2]:

1. *Сбор данных.* На этом этапе необходимо составить набор признаков для описания моделируемого процесса. Аналитик должен представлять возможные связи между признаками и ясно понимать сущность задачи. После выбора признаков необходимо установить, насколько значимы они и не отражают ли другие существенные признаки.

2. *Преобразование данных.* Основная цель преобразования данных заключается в приведении их к одним единицам, например, в интервале $[0;1]$ или $[-1;1]$. Достичь результата можно, выполнив определенные преобразования: линейные, с помощью статистических характеристик, с помощью логистической функции, с помощью функции активации нейронов и т.д. Также полезными будут определенные комбинации признаков, такие как произведения, частные и т.д. С их помощью можно получить более простую модель.

3. *Построение модели сети.* Одним из важных факторов в построении модели сети является определение ее топологии. Топология сети может быть выбрана на основе имеющихся данных с применением статистических, конструктивных, деструктивных либо эволюционных методов.

4. *Обучение.* Немаловажную роль в успешном обучении сети играет активационная функция нейронов. Выделим перечень наиболее используемых функций активации [3]: сигмовидная с выходными значениями в промежутке $[0,1]$, сигмовидная с выходными

значениями в промежутке $[-1,1]$, рациональная сигмовидная, гиперболический тангенс, логарифмическая, гауссовская, радиально-симметричная, пороговая, многопороговая, тождественная (линейная), линейная с насыщением, биполярная. Из перечисленных выше сигмовидная функция явно выделяется благодаря своему преимуществу эффективно вычислять производную и ограничивать выходное значение.

ПАРАДИГМЫ ОБУЧЕНИЯ

Для конструирования процесса обучения, прежде всего, необходимо иметь модель внешней среды, в которой функционирует нейронная сеть, т.е. знать доступную для сети информацию. Эта модель определяет парадигму обучения. Существует три парадигмы обучения: с учителем, без учителя (самообучение) и смешанная [4].

Обучение с учителем основано на том, что имеются знания об окружающей среде, представленные в виде готовых пар вход–выход, где выход представляет собой желаемый результат, к которому должна стремиться обучаемая сеть. Разность между желаемым сигналом и текущим откликом нейронной сети называют сигналом ошибки. При таком обучении возникает замкнутая система с обратной связью, которая не включает в себя окружающую среду. В процессе обучения сеть получает только те данные об окружающей среде, которые были отобраны учителем. Схема обучения с учителем представлена на рисунке 1.

Несмотря на многочисленные прикладные достижения, обучение с учителем критикуется за свою биологическую неправдоподобность. Трудно вообразить обучающий механизм в мозге, который бы сравнивал желаемые и действительные значения выходов, выполняя коррекцию с помощью обратной связи. Если допустить подобный механизм в мозге, то откуда тогда возникают желаемые выходы?



Рисунок 1 – Схема обучения с учителем [4]

Обучение без учителя является намного более правдоподобной моделью обучения в биологической системе. Модель не нуждается в целевом векторе для выходов и, следовательно, не требует сравнения с predetermined идеальными ответами. Обучающее множество состоит лишь из входных векторов. Обучающий алгоритм подстраивает веса сети так, чтобы получались согласованные выходные векторы, т. е. чтобы предъявление достаточно близких входных векторов давало одинаковые выходы. Процесс обучения, следовательно, выделяет статистические свойства обучающего множества и группирует сходные векторы в классы. Предъявление на вход вектора из данного класса даст определенный выходной вектор, но до обучения невозможно предсказать, какой выход будет производиться данным классом входных

векторов. Следовательно, выходы подобной сети должны трансформироваться в некоторую понятную форму, обусловленную процессом обучения. Схема обучения без учителя представлена на рисунке 2.

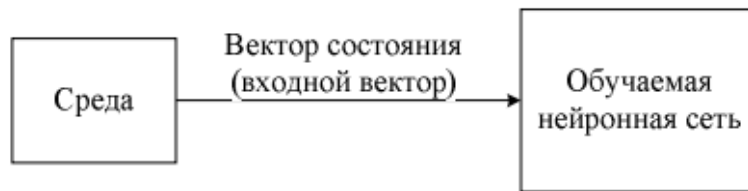


Рисунок 2 – Схема обучения без учителя

При организации обучения нейронной сети необходимо уделить большое внимание отбору обучающих примеров, к которым предъявляются условия:

- 1) исходные данные должны быть непротиворечивы, т.е. один и тот же входной вектор не может производить разные выходные сигналы;
- 2) обучающее множество должно быть полным, т.е. количество примеров, предоставляемых алгоритму обучения, должно быть достаточным, чтобы обеспечить его всей необходимой информацией для изучения зависимости.

В случае, когда невозможно выполнить первое условие из-за отсутствия однозначной и чёткой зависимости между входными сигналами (параметрами информационной системы) и результатом работы нейронной сети (оценкой качественных характеристик системы), следует использовать нейронные нечёткие сети – сети, функционирующие на основе теории нечётких множеств и нечёткой логики. Основная идея состоит во введении нечёткости в структуру нейронной сети и алгоритм её обучения.

Существует достаточно большое количество алгоритмов обучения нейронной сети, наиболее популярным из которых является алгоритм обратного распространения ошибки.

АЛГОРИТМ ОБРАТНОГО РАСПРОСТРАНЕНИЯ ОШИБКИ

Алгоритм включает в себя три фазы [5]:

- проведение через сеть обучающего примера (фаза прямого распространения);
- вычисление значения ошибки между полученным выходом сети и ожидаемым, на основе которого затем вычисляются значения ошибок для каждого предыдущего слоя вплоть до входного (фаза обратного распространения ошибки);
- единовременная корректировка всех весов, основываясь на их текущем значении, значении ошибки, соответствующей рассматриваемому нейрону и активационному значению нейрона.

Обучение с помощью алгоритма обратного распространения ошибки может потребовать значительных вычислительных средств и времени, но получившаяся в итоге сеть работает крайне быстро. Для реализации и использования уже обученной сети необходима только первая фаза алгоритма.

Для работы алгоритма необходимо рассчитать производную функции ошибки. Функция ошибки для сети с J выходами задается аналогичным способом [5]:

$$E(\vec{x}) = \sum_{j \in J} (y_j - t_j)^2, \quad (1)$$

где y_j – реальный выход j -го нейрона сети; t_j - ожидаемый выход j -го нейрона.

Пусть $f(u)$ – функция активации нейронов в выходном слое, тогда значение j -го выходного нейрона будет определяться следующим способом:

$$y_j = f(u_j) = f\left(\sum_{i \in I} w_{ij} * z_i\right), \quad (2)$$

где w_{ij} – веса при связях, соединяющие i -й нейрон предыдущего слоя и j -й нейрон выходного слоя; z_i – выходное значение i -го нейрона предыдущего слоя; I – число нейронов в предыдущем слое.

Производная функции ошибки вдоль синаптической связи w_{ij} :

$$\frac{\partial E(\bar{x})}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \left(\sum_{k \in J} (y_k - t_k)^2 \right) = \frac{\partial}{\partial y_j} \left(\sum_{k \in J} (y_k - t_k)^2 \right) * \frac{\partial y_j}{\partial w_{ij}}. \quad (3)$$

Подобное преобразование возможно, так как t_k не зависит от w_{ij} . Рассмотрим первый множитель:

$$\frac{\partial}{\partial y_j} \left(\sum_{k \in J} (y_k - t_k)^2 \right) = 2 * (y_j - t_j). \quad (4)$$

Учтем, как выражается y_j через активационную функцию, и рассмотрим второй множитель:

$$\frac{\partial y_j}{\partial w_{ij}} = \frac{\partial f(u_j)}{\partial w_{ij}} = \frac{\partial f(u_j)}{\partial u_j} * \frac{\partial u_j}{\partial w_{ij}} = f'(u_j) * \frac{\partial (\sum_k w_{kj} * z_k)}{\partial w_{ij}} = f'(u_j) * z_i, \quad (5)$$

Конечный вид производной будет иметь вид:

$$\frac{\partial E(\bar{x})}{\partial w_{ij}} = 2 * (y_j - t_j) * f'(u_j) * z_i. \quad (6)$$

Легко заметить, что, если выход сети равен ожидаемому значению, то значение производной функции ошибки обращается в нуль.

На основе производной вычисляется величина изменения веса w_{ij} в ходе обучения:

$$\Delta w_{ij} = -l * \frac{\partial E(\bar{x})}{\partial w_{ij}}. \quad (7)$$

В ходе первой фазы алгоритма набор данных \bar{x} из обучающей выборки пропускается через сеть. Последовательно обходится каждый слой сети, начиная с входного слоя и заканчивая выходным. Каждый нейрон вычисляет свой вход и выход и распространяет его дальше по синаптическим связям.

На втором шаге на основе вектора \bar{y} , полученного из сети, и целевого вектора \bar{t} , соответствующего входным данным, для каждого нейрона выходного слоя вычисляется значение ошибки e_j . Эти значения используются для распространения ошибки на все нейроны предыдущего слоя (последнего из скрытых слоев) и для изменения весов связей между скрытым слоем и выходным слоем. Затем для каждого предыдущего слоя рассчитываются

значения e_i для соответствующих нейронов. Слои обходят в порядке, обратном порядку обхода в первой фазе алгоритма.

Рассмотрим произвольный скрытый слой с функцией активации f . Пусть в нем находится I нейронов, а в следующем слое находится J нейронов. Рассмотрим i -й нейрон. Он связан со всеми нейронами следующего слоя, и для каждого из них уже подсчитано соответствующее значение ошибки e_j . Также в ходе фазы прямого распространения было вычислено значение входа нейрона u_i . Тогда для i -го нейрона значение e_i будет рассчитываться по формуле:

$$e_i = \sum_{j=1}^J e_j * w_{ij} * f'_{w_{ij}}(u_i). \quad 8)$$

Нет необходимости считать значения ошибок для входного слоя, но веса, связывающие входной слой и скрытый, также будут модифицированы на следующем шаге алгоритма.

После того, как значения ошибок для всех нейронов всех слоев вычислены, можно приступить к корректировке весовых коэффициентов. Значение веса w_{ij} , соответствующее синаптической связи, соединяющей i -й нейрон скрытого слоя с j -м нейроном следующего слоя, изменяется по формуле:

$$w_{ij}^{new} = w_{ij}^{old} + l * e_j * z_i. \quad 9)$$

где z_j – это значение сигнала, которое было передано по синаптической связи между i -м нейроном скрытого слоя и j -м нейроном выходного слоя, или, другими словами, выходное значение i -го нейрона; l – скорость обучения, задающая скорость изменения весов. Для связей, соединяющих входной слой с первым скрытым слоем, формула примет следующий вид:

$$w_{ij}^{new} = w_{ij}^{old} + l * e_j * x_i. \quad 10)$$

где x_i – это i -й компонент входного вектора \bar{x} .

Обучение нейронной сети с помощью алгоритма обратного распространения ошибки является вычислительно сложной задачей. Несмотря на наличие высокоскоростных вычислительных систем, обучение сети может занять несколько часов или дней.

ВЫВОДЫ

Таким образом, для успешного решения задачи обучения нейронной сети необходимо:

1) Правильно выбрать структуру сети (т.е. количество слоев и количество нейронов в каждом слое). Выбор подходящей структуры сети должен соответствовать характеру решаемой задачи.

2) Правильно выбрать параметры обучения, в частности, шаг обучения сети, норму обучения, количество обучающих примеров.

3) Правильно подготовить входные данные. Желательно, чтобы перед началом обучения исследователь в общих чертах представлял себе структуру входных данных, отфильтровал шум, избавился от малозначимых составляющих входных данных. В некоторых случаях предварительная обработка данных может предусматривать линейное преобразование, выделяющее из всей массы входных данных некоторый набор наиболее значимых направлений или векторов. Иногда такое преобразование можно поручить дополнительной нейронной сети,

содержащей в скрытом слое меньше нейронов, чем во входном или выходном слоях, что позволяет понижать размерность входных данных.

ЛИТЕРАТУРА

1. Попова Ю.Б., Яцынович С.В. Нейронные сети в обучающих системах: Информационные технологии в технических и социально-экономических системах: сб. материалов науч.-практ. конф. – Минск: БНТУ, 2016. – С. 20.
2. Нейронные сети // [Электронный ресурс]. – Режим доступа: <http://statsoft.ru/home/textbook/modules/stneunet.html> – Дата доступа: 06.11.2016.
3. Хайкин С. Нейронные сети: полный курс. – 2-е изд. – М.: Вильямс, 2006. – 1104 с.
4. Математические методы распознавания образов // [Электронный ресурс]. – Режим доступа: http://window.edu.ru/resource/800/73800/files/lect_Lepskiy_Bronevich_pass.pdf – Дата доступа: 06.11.2016.
5. Разработка нейросетевого алгоритма // [Электронный ресурс]. – Режим доступа: <http://pandia.ru/text/80/127/55955-6.php> – Дата доступа: 06.11.2016.