УДК 004.81

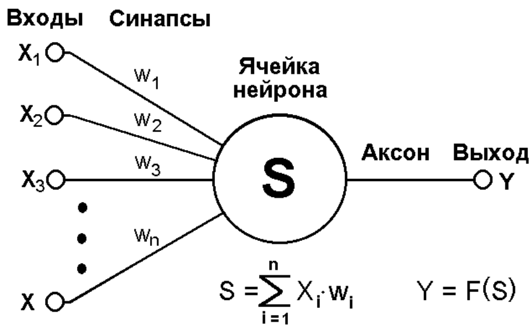
ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ АЛГОРИТМОВ ОБУЧЕНИЯ НЕЙРОННОЙ СЕТИ

Сорокина К.М.,
научный руководитель канд. техн. наук Липинский Л.В.
Сибирский федеральный университет

В последние десятилетия в мире бурно развивается новая прикладная область математики, специализирующаяся на искусственных нейронных сетях (НС). Актуальность исследований в этом направлении подтверждается массой различных применений НС. Это автоматизация процессов распознавания образов, адаптивное управление, аппроксимация функционалов, прогнозирование, создание экспертных систем, организация ассоциативной памяти и многие другие приложения. С помощью НС можно, например, предсказывать показатели биржевого рынка, выполнять распознавание оптических или звуковых сигналов, создавать самообучающиеся системы, способные управлять автомашиной при парковке или синтезировать речь по тексту.

Широкий круг задач, решаемый НС, не позволяет в настоящее время создавать универсальные, мощные сети, вынуждая разрабатывать специализированные НС, функционирующие по различным алгоритмам.

Общий вид нейрона:



Нелинейная функция F называется активационной и может иметь различный вид. Одной, из наиболее распространенных, является нелинейная функция с насыщением, так называемая логистическая функция или сигмоид : $f(x) = \dfrac{1}{1+e^{-\alpha x}}$

Следует отметить, что сигмоидная функция дифференцируема на всей оси абсцисс, что используется в некоторых алгоритмах обучения. Кроме того она обладает свойством усиливать слабые сигналы лучше, чем большие, и предотвращает насыщение от больших сигналов, так как они соответствуют областям аргументов, где сигмоид имеет пологий наклон.

Среди различных структур НС одной из наиболее известных является многослойная структура, в которой каждый нейрон произвольного слоя связан со всеми аксонами нейронов предыдущего слоя или, в случае первого слоя, со всеми входами НС. Такие НС называются полносвязными. Когда в сети только один слой, алгоритм ее обучения с учителем довольно очевиден, так как правильные выходные состояния нейронов единственного слоя заведомо известны, и подстройка синаптических связей идет в направлении, минимизирующем ошибку на выходе сети. По этому принципу строится, например, алгоритм обучения однослойного перцептрона. В многослойных же сетях оптимальные выходные значения нейронов всех слоев, кроме последнего, как правило, не известны, и двух или более слойный перцептрон уже невозможно обучить, руководствуясь только величинами ошибок на выходах НС. Один из вариантов решения этой проблемы – динамическая подстройка весовых коэффициентов синапсов, в ходе которой выбираются, как правило, наиболее слабые связи и изменяются на малую величину в ту или иную сторону, а сохраняются только те изменения, которые повлекли уменьшение ошибки на выходе всей сети. Ещё один вариант – распространение сигналов ошибки от выходов НС к ее входам, в направлении, обратном прямому распространению сигналов в обычном режиме работы. Этот алгоритм обучения НС получил название процедуры обратного распространения.

Согласно методу наименьших квадратов, минимизируемой целевой функцией ошибки НС является величина:

$$E(w) = \frac{1}{2}\sum_{j,p}(y_{j,p}^{(N)} - d_{j,p})^2$$

где $y_{j,p}^{(N)}$ – реальное выходное состояние нейрона j выходного слоя N нейронной сети при подаче на ее входы p-го образа; d_{jp} – идеальное (желаемое) выходное состояние этого нейрона.

Суммирование ведется по всем нейронам выходного слоя и по всем обрабатываемым сетью образам. Минимизация ведется методом градиентного спуска.

В данном докладе мы будем рассматривать алгоритмы обучения НС:

1. Градиентный спуск с постоянным шагом
2. Метод сопряженных градиентов (алгоритм Флетчера-Ривса)

Эти методы характеризуются проблемами выбора начальной точки поиска, локальным характером поиска, возможностью попадания в локальные экстремумы функционала обучения, необходимостью расчета производных целевой функции по весам сети, что является существенной проблемой для написания программного кода. В связи с этим в программе был реализован численный метод взятия производной.

Так как перед нами стоит задача сравнения эффективности уже двух реалиизованных в программе методов, то мы должны выставить критерий. На основании которого, мы будем сравнивать эффективность полученного результата тем или иным способом. Есть два более распространенных способа:

1. Ограничить количество вычислений целевой функции, и сравнивать полученную ошибку.
2. Обозначить «желаемую» ошибку, и сравнивать количество вычислений целевой функции, необходимых методу для получения данной ошибки.

Мы воспользуемся первым методом, так как для нас он более эффективен. Потому что из-за недостатков которыми обладают данные методы мы не можем гарантировать, что функция ошибки сойдется с необходимой нам точностью.

Задачи, на которых мы будем тестировать реализованные в программе методы:

1. Парабола
2. Парабола с «шумом» (отклонение порядка 0-0.1%)
3. X^3+Y^2

В таблице представлено среднее значение минимизируемой целевой функциии ошибки полученной на тестовой выборке десятикратным прогоном после обучения:

	Градиентный спуск с постоянным шагом	Метод сопряженных градиентов
Парабола	0,0025261131	0,0021574816
Парабола с «шумом» (отклонение порядка 0-0.1%)	0,1854083162	0,1694469861
X^3+Y^2	1,509863241	1,237563010

В данных примерах мы ограничились 68 000 итерациями и начальным шагом 0,001. Активационная функция – $f(x)=\frac{1}{1+e^{-\alpha x}}$. Структура НС - многослойный персептрон с размером 3x4.

В процессе тестирования функций было выявлено, что из-за обозначенных выше недостатков данных методов, существует проблема выбора «подходящей» функции и параметров для работы с ними. Поэтому первоначально стоит задача в усовершенствовании данных методов и реализации дополнительных методов, с помощью которых можно будет получать представление об эффективности того или иного способа.

На основании полученных результатов можно сделать вывод, что метод сопряженных градиентов при сопоставимом количестве вычислений целевой функции работает более эффективно, чем метод градиентного спуска с постоянным шагом.