

Н. Т. Абдуллаев, канд. техн. наук, доц., a.namik46@mail.ru,

Азербайджанский технический университет,

К. Ш. Исмаилова, канд. техн. наук, доц., Is_kamalya@yahoo.com,

Азербайджанский государственный университет нефти и промышленности

Выбор шага итерации в процессе обучения нейронной сети при использовании релаксационных алгоритмов

Рассматриваются методы выбора шага итерации при обучении нейронных сетей. Представлена таблица возможных методов выбора шага и рекомендуемые варианты применения этих методов. Разработан алгоритм выбора шага при использовании релаксационных алгоритмов с учетом эффективности и временных затрат реализации. Данный алгоритм эффективен в реальной работе и прост в реализации. Помимо этого такой алгоритм удобно применять вместе с алгоритмами "тяжелый шарик", PARTAN и даже BFGS, которые широко используются в реальных задачах.

Ключевые слова: нейронная сеть, релаксационные алгоритмы, шаг итерации, методы градиентного спуска, эффективность, простота реализации, оптимизация, обучение

Введение

В информационных источниках имеется достаточно много материалов, посвященных алгоритмам оптимизации и реализации этих алгоритмов для нейронных сетей. Тем не менее вопросы алгоритмического обеспечения при выборе шага итерации для обучения нейронных сетей достаточно актуальны.

В методе градиентного спуска [1] осуществляется поиск по самому крутому направлению спуска, противоположному направлению градиента целевой функции. Он может обеспечить уменьшение целевой функции до тех пор, пока текущая точка итерации не станет неподвижной. Различные варианты ступенчатых размеров приводят к различным алгоритмам градиентного спуска. Распространенным вариантом метода градиентного спуска является метод точного поиска по линии, который выбирает следующую итерацию путем достижения наименьшего значения функции. Однако этот метод имеет недостаток, поскольку часто имеет медленную сходимость.

В последние годы были проведены исследования, направленные на выбор размера шага для метода градиентного спуска согласно рекомендациям Барзилая и Борвеина [2], где дан

конкретный выбор размера шага, подтверждающий суперлинейную сходимость для двумерных выпуклых квадратичных задач, и указаны рекомендуемые размеры шага для метода градиентного спуска.

Авторы работы [3] используют один из вариантов метода градиентного спуска для повышения эффективности наиболее применяемого алгоритма при обучении многослойных нейронных сетей — Backpropagation, который использует адаптивную скорость обучения, суть которого заключается в обратном распространении ошибок с переменным шагом. Параллельные касательные методы используются в глобальной оптимизации для изменения и улучшения простого алгоритма градиентного спуска, в этом случае иногда при поиске направления спуска используется разница между текущей точкой и точкой до двух шагов от текущей. Для обучения нейронной сети авторы предлагают комбинацию метода BPVS с методом Parallel Tangent.

В работе [4] рассматривается вариация часто используемого алгоритма для решения проблемы выбора трафика (в транспорте) — алгоритм Франка — Вольфа. Вариацией этого алгоритма является алгоритм PARTAN (сокращение от параллельного касания), в котором ускоре-

ние конвергенции достигается путем введения дополнительного алгоритма поиска по линии. Показано, что в варианте PARTAN алгоритма Франка — Вольфа для вычисления минимальной длины шага для распределения равновесия могут быть использованы либо аналитические формулы, либо простые правила.

В последние годы методы стохастического градиентного спуска (СГС) стали стандартным инструментом для обучения нейронных сетей. В [5] проведен анализ конвергенции для СГС на богатом подмножестве двухслойных сетей прямого доступа с функциями активации ReLU (Rectified Linear Units). Это подмножество характеризуется специальной структурой, называемой "отображение идентичности". Доказывается, что если входные данные выбраны из гауссовского распределения при стандартной инициализации весов, то СГС сходится к глобальному минимуму за полиномиальное число шагов. В отличие от обычных сетей "отображение идентичности" делает такую сеть асимметричной и, следовательно, глобальный минимум уникальным. В дополнение к теории можно экспериментально показать, что многослойные сети с этим отображением имеют лучшую производительность по сравнению с нормальными сетями. Такой подход к сходимости алгоритма отличается от традиционных невыпуклых методов оптимизации.

В работе [6] исследуются алгоритмы автоматической адаптации скорости обучения нейронных сетей (НС). Начиная со стохастического градиентного спуска, большое разнообразие методов обучения было предложено для настройки НС. Однако эти методы обычно чувствительны к начальной скорости обучения, которая должна быть выбрана экспериментатором. В работе [6] исследуются несколько функций и показано, как адаптивный контроллер может настроить обучение без предварительного знания изучаемой проблемы.

В статье [7] рассмотрен метод обучения для двухслойных первичных нейронных сетей на основе анализа чувствительности, который использует алгоритм линейного обучения для каждого из двух уровней. Сначала выходам присваиваются случайные значения первого уровня; позже эти начальные значения обновляются на основе формул чувствительности, которые используют веса в каждом из слоев.

Поскольку эти веса изучаются при решении линейной системы уравнений, то сохраняется важная экономия вычислительного времени. При этом появляется возможность анализа

локальной чувствительности методом наименьших квадратов для оценки ошибок по входным и выходным данным без дополнительных вычислительных затрат, поскольку необходимая информация становится доступной без дополнительных вычислений. Основное преимущество анализа локальной чувствительности реализуется, в частности, в многомерных задачах, где глобальный анализ требует слишком много времени. Этот метод, называемый методом линейного обучения на основе чувствительности, также может быть использован для обеспечения начального набора весов, что значительно улучшает поведение других алгоритмов обучения. Дана теоретическая основа метода, и его производительность иллюстрируется применением на нескольких примерах, в которых его сравнивают с разными алгоритмами обучения на хорошо известных наборах данных. Результаты показали высокую скорость обучения по сравнению с другими существующими методами. Кроме того, его можно использовать в качестве инструмента инициализации для других хорошо известных методов со значительными улучшениями.

В работе [8] предложен алгоритм оптимизации обучения неглубоких нейронных сетей. Высокоразмерные стохастические задачи оптимизации представляют собой интересные проблемы для таких существующих алгоритмов обучения. Разрабатывается расширение, подходящее для алгоритмов оптимизации обучения, и демонстрируется, что изученный алгоритм оптимизации последовательно превосходит другие известные алгоритмы оптимизации даже для труднорешаемых задач и устойчив к изменениям стохастичности градиентов и архитектуры нейронной сети.

В работе [9] вводится новый метод "важности веса" (IW) для ускорения обучения на основе градиента. Этот метод особенно полезен для "сложных" наборов данных, включая несбалансированные данные, данные с высоконелинейными отношениями между переменными или с долгосрочными зависимостями в последовательностях. Характеристика "важность веса" присваивается каждой точке данных обучающей выборки и контролирует вклад точки данных в общую ошибку обучения в соответствии с его информативностью до получения результатов хорошей предсказуемости. Характеристику "важность веса" можно также использовать для индивидуального выбора размера шага для локального градиента в конкретной точке данных. Эта цель достига-

ется путем квадратичной оптимизации, которая сводит к минимуму абсолютное значение изменения вектора параметров во время этапа обучения с (мягким) ограничением, так чтобы общая ошибка была уменьшена на наименьшее заданное фиксированное значение. Для линейных классификаторов (способ решения задач классификации, когда решение принимается на основании линейного оператора над входными данными) показано, что данный метод поддерживает стандарты векторного обучения. Применяется метод IW и к многослойным персептронам, и к рекуррентным нейронным сетям (LSTM).

Методы выбора шага итерации

Выбор в пользу градиентных методов обоснован тем, что, как правило, в задачах обучения критерий обучения можно выразить в виде дифференцируемой функции от весов нейронной сети. Тем не менее неопределенность выбора метода обучения сохраняется. В автоматизированных системах нейросетевого программирования следует стремиться к сокращению неопределенности, которая присуща этим технологиям. Неопределенность в выборе алгоритма обучения в некоторой степени устраняется в предлагаемом адаптивном методе обучения [10]. На основании проведенного анализа выбора шага итерации рассмотрены методы и их применение. Результаты сведены в таблицу.

Описание выбора шага и применение алгоритмов

Метод выбора шага	Применение
Задать постоянным (убывание градиента при приближении к минимуму)	Градиентные методы
Увеличить шаг при убывании ошибки и уменьшить при возрастании	Простые релаксационные алгоритмы (алгоритм "тяжелого шарика", PARTAN, BFGS (Broyden — Fletcher — Goldfarb — Shanno algorithm))
Последовательное приближение зависимости функционала ошибки от шага с помощью квадратичной зависимости	Методы сопряженных градиентов, BFGS, методы второго порядка
Вычислить производную функционала ошибки в нуле	Методы сопряженных градиентов, BFGS, методы второго порядка
Определение направления с большими вычислительными затратами	Метод статистического градиента

Различные формы регулярных итеративных методов отличаются друг от друга конкретным выбором шага [11].

Алгоритмическая реализация выбора шага

К алгоритмам оптимизации с "нелинейным" шагом и соответствующей дискретной системой с нелинейными коэффициентами усиления относятся релаксационные алгоритмы, в которых постоянная матрица на каждом шаге выбирается так, чтобы уменьшалась функция ошибки $E(w_k + \eta_k - \textcolor{red}{1}p_k)$, где w_k — вектор коэффициентов, p_k — параметр, влияющий на изменение направления, η_k — размер шага на k -й итерации.

Как известно, если p_k — направление убывания, то алгоритмы "тяжелый шарик" и PARTAN удобно применять при выборе шага. Идея выбора состоит в том, чтобы увеличивать шаг при убывании ошибки и уменьшать при возрастании. Алгоритм выбора шага итерации более удачный и простой в реализации [12] и его удобно применять с релаксационными алгоритмами. К таким относится известный алгоритм наискорейшего спуска. Предлагается алгоритм выбора шага, который состоит из пяти шагов (см. рисунок).

Шаг 1. В начале всего алгоритма задается начальный шаг. Задаются параметры алгоритма, удовлетворяющие следующим неравенствам: $\gamma_1 > 1$; $0 < \gamma_2 < 1$.

Шаг 2. После нахождения очередного направления движения в качестве начального приближения берется шаг из предыдущей итерации.

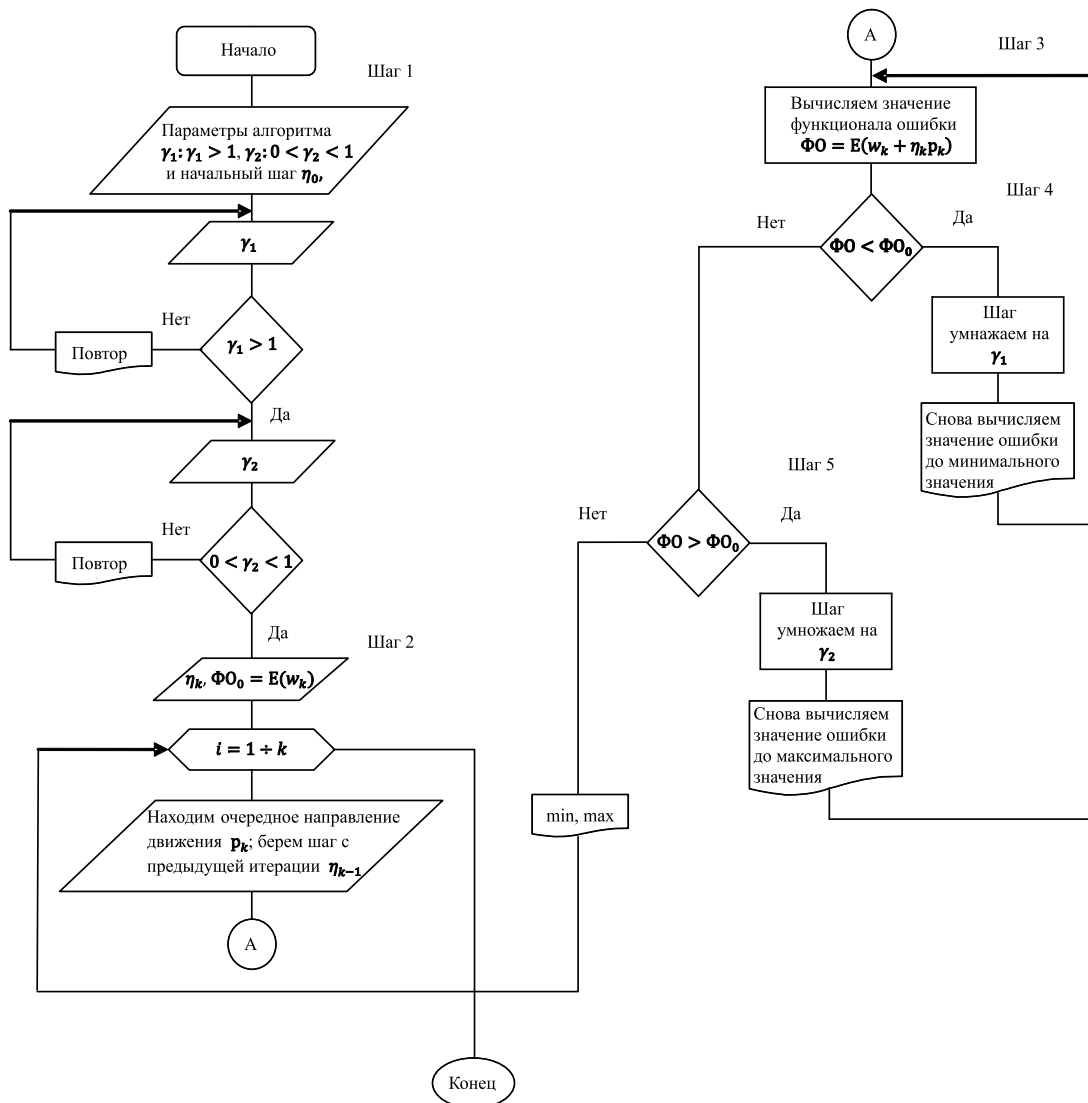
Шаг 3. Вычисляется значение функционала ошибки $\Phi O = E(w_k + \eta_k - \textcolor{red}{1}p_k)$.

Шаг 4. Если это значение меньше предыдущего $E(w_k)$, тогда шаг умножаем на γ_1 и снова вычисляем значение ошибки до тех пор, пока ошибка не начинает возрастать. Предпоследняя минимальная точка фиксируется, и далее вычисляется новое направление.

Шаг 5. Если $E(w_k + \eta_k - \textcolor{red}{1}p_k) > E(w_k)$, шаг умножается на γ_2 , и продолжается уменьшение шага, пока ошибка не станет меньше $E(w_k)$. Полученный шаг фиксируется, и вычисляется новое направление.

Заключение

Данный алгоритм эффективен в реальной работе и прост в реализации. Такой алгоритм удобно применять вместе с алгоритмами "тяжелый шарик", PARTAN и даже BFGS, кото-



Алгоритм выбора шага итерации

рые широко используются в реальных задачах. Таблица методов выбора шага и их реализации полезна пользователям при выборе шага итерации в экспериментах.

Список литературы

1. Ya-xiang Y. Step-Sizes for the Gradient Method. URL: <ftp://lsec.cc.ac.cn/pub/yyx/papers/p0504.pdf> 30.07.2018
2. Barzilai J., Borwein J. M. Two point step size gradient methods // IMA J. Numer. Anal. 1988. Vol. 8. P. 141–148.
3. Petalas Y. G., Vrahatis M. N. Parallel Tangent Methods with Variable Stepsize. URL: <http://www.math.upatras.gr/~petalas/ijcnn04.pdf> (дата обращения: 30.07.2018)
4. Arezki Y., Van Vliet D. A Full Analytical Implementation of the PARTAN/Frank-Wolfe Algorithm for Equilibrium Assignment. URL: https://www.researchgate.net/publication/244956239_A_Full_Analytical_Implementation_of_the_PARTANFrank-Wolfe_Algorithm_for_Equilibrium_Assignment (дата обращения: 30.07.2018)
5. Yuanzhi Li, Yang Y. Convergence Analysis of Two-layer Neural Networks with ReLU Activation // 31st Conference on

Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA. 2017. 11 p.

6. Daniel Ch., Taylor J., Nowozin S. Learning Step Size Controllers for Robust Neural Network Training // Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16). 2016. P. 1519–1525.
7. Castillo E., Guijarro-Berdinas B., Fontenla-Romero O., Alonso-Betanzos A. A Very Fast Learning Method for Neural Networks Based on Sensitivity Analysis // Journal of Machine Learning Research. 2006. Iss.7. P. 1159–1182.
8. Li K., Jitendra M. Learning to Optimize Neural Nets. URL: <https://arXiv:1703.00441v2> [cs.LG]. 2017. 10 p. (дата обращения: 05.09.2018)
9. Hochreiter S., Obermayer K. Optimal Gradient-Based Learning Using Importance Weights. URL: www.bioinf.jku.at/publications/20051205.pdf (дата обращения: 05.09.2018)
10. Борисович Л. В. Алгоритм и программная реализация адаптивного метода обучения искусственных нейронных сетей. URL: <https://cyberleninka.ru/article/n/algoritmi-programmnaya-realizatsiya-adaptivnogo-metoda-obucheniya-iskusstvennyh-neyronnyh-setey.pdf> (дата обращения: 10.08.2018)
11. Разновидности алгоритмов оптимизации. URL: http://sernam.ru/book_zip.php?id=17 (дата обращения: 01.08.2018)
12. Тархов Д. А. Нейросетевые модели и алгоритмы. Справочник. М.: Радиотехника, 2014. 352 с.

N. T. Abdullayev, Ph. D., Associate Professor, e-mail: a.namik46@mail.ru,
Azerbaijan Technical University,

K. Sh. Ismayilova, Ph. D., Associate Professor, e-mail: Is_kamalya@yahoo.com,
Azerbaijan State University of Oil and Industry

The Choice of the Iteration Step in the Process of Learning a Neural Network Using Relaxation Algorithms

In information sources, there are a lot of materials devoted to algorithms for optimization and implementation of these algorithms for neural networks. However, the issues of algorithmic support in selecting the iteration step, for learning neural networks are quite relevant. The choice in favor of gradient methods is justified by the fact that, as a rule, in learning problems, the learning criterion can be expressed in the form of a differentiable function of the weights of the neural network. Nevertheless, the uncertainty of the choice of the method of instruction is preserved. In automated systems of neural network programming, one should strive to reduce the uncertainty that is inherent in these technologies. The uncertainty in the choice of the learning algorithm is to some extent eliminated in the proposed adaptive learning method. The algorithm for selecting a step consists of 5 levels (steps).

Step 1. At the beginning of the whole algorithm the initial step is set. Parameters of the algorithm must satisfy the following inequalities $-\gamma_1: \gamma_1 > 1, \gamma_2: 0 < \gamma_2 < 1$.

Step 2. After finding the next direction of motion, the step from the previous iteration is taken as the initial approximation.

Step 3. After that, the value of the error functional $E(w_k + \eta_k - \beta p_k)$ is calculated.

Step 4. If this value is less than the previous one $E(w_k)$, then step umnazhaem to γ_1 and again calculate the value of the error until the error begins to increase. The penultimate minimum point is fixed and a new direction is computed.

Step 5. If $E(w_k + \eta_k - \beta p_k) > E(w_k)$, the step is multiplied by γ_2 and the step decreases until the error becomes less than $E(w_k)$. The resulting step is fixed and a new direction is calculated.

This algorithm is effective in real work and easy to implement. In addition, it is convenient to apply this algorithm together with the heavy ball, Partan and even BFGS algorithms, which are widely used in real problems. The table of methods for selecting a step and their implementation is useful to users when choosing the iteration step in experiments.

Keywords: neural network, relaxation algorithms, iteration step, gradient methods, efficiency, prostate implementation, optimization, learning

DOI: 10.17587/it.25.174-178

References

1. **Ya-xiang Y.** Step-Sizes for the Gradient Method, available at: <ftp://lsec.cc.ac.cn/pub/yyx/papers/p0504.pdf> 30.07.2018
2. **Barzilai J., Borwein J. M.** Two point step size gradient methods, *IMA J. Numer. Anal.*, 1988, vol. 8, pp. 141–148.
3. **Petalas Y. G., Vrahatis M. N.** Parallel Tangent Methods with Variable Stepsize, available at: <http://www.math.upatras.gr/~petalas/ijcnn04.pdf> (date of access: 30.07.2018)
4. **Arezki Y., Van Vliet D.** A Full Analytical Implementation of the PARTAN/Frank-Wolfe Algorithm for Equilibrium Assignment, available at: https://www.researchgate.net/publication/244956239_A_Full_Analytical_Implementation_of_the_PARTANFrank-Wolfe_Algorithm_for_Equilibrium_Assignment (date of access: 30.07.2018)
5. **Yuanzhi Li, Yang Y.** Convergence Analysis of Two-layer Neural Networks with ReLU Activation, *31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, 2017, 11 p.
6. **Daniel Ch., Taylor J., Nowozin S.** Learning Step Size Controllers for Robust Neural Network Training, *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, 2016, pp. 1519–1525.
7. **Castillo E., Guijarro-Berdinas B., Fontenla-Romero O., Alonso-Betanzos A.** A Very Fast Learning Method for Neural Networks Based on Sensitivity Analysis, *Journal of Machine Learning Research*, 2006, iss.7, pp. 1159–1182.
8. **Li K., Jitendra M.** Learning to Optimize Neural Nets, available at: <https://arXiv:1703.00441v2> [cs.LG], 2017, 10 p. (date of access: 05.09.2018).
9. **Hochreiter S., Obermayer K.** Optimal Gradient-Based Learning Using Importance Weights, available at: www.bioinf.jku.at/publications/20051205.pdf (date of access: 05.09.2018).
10. **Borisovich L. V.** *Algoritmi i programnaya realizatsiya adaptivnogo metoda obucheniya iskusstvennyh neyronnyh setey* (Algorithm and software implementation of the adaptive method of training artificial neural networks), available at: <https://cyberleninka.ru/article/n/algoritmi-i-programnaya-realizatsiya-adaptivnogo-metoda-obucheniya-iskusstvennyh-neyronnyh-setey>.pdf (date of access: 10.08.2018).
11. **Raznovidnosti algoritmov optimizatsii** (Varieties of optimization algorithms), available at: http://sernam.ru/book_zip.php?id=17 (date of access: 01.08.2018).
12. **Tarhov D. A.** *Neyrosetevye modeli i algoritmi* (Neural network models and algorithms), Moscow, Radiotekhnika, 2014, 352 p. (in Russian).