КОМП'ЮТЕРНІ ЗАСОБИ, МЕРЕЖІ ТА СИСТЕМИ

УДК 519. 7004. 62

Т.А. САМОЛЮК

ИССЛЕДОВАНИЕ СТОХАСТИЧЕСКИХ ГРАДИЕНТНЫХ МЕТОДОВ ОПТИМИЗАЦИИ АЛГОРИТМОВ ОБУЧЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

T. Samolyuk

STUDY OF STOCHASTIC GRADIENT METHODS FOR OPTIMIZATION OF ALGORITHMS OF LEARNING ARTIFICIAL NEURAL NETWORKS

Stochastic gradient methods for optimizing the learning of artificial neural networks are analyzed. Considerable attention is paid to the SAG (stochastic mean gradient method).
Key words: neural networks, information systems, gradient methods.

Проаналізовано стохастичні градієнтні методи оптимізації навчання штучних нейронних мереж. Значна увага приділена методу SAG (стохастичного середнього градієнта).
Ключові слова: нейронні мережі, інформаційні системи, градієнтні методи.

Проанализированы стохастические градиентные методы оптимизации обучения искусственных нейронных сетей. Значительное внимание уделено методу SAG (стохастического среднего градиента).
Ключевые слова: нейронные сети, информационные системы, градиентные методы.

Введение. Информация – это ключевой элемент в принятии решений. В настоящее время большое значение в обработке информации приобретают задачи обработки и классификации больших объемов данных и задачи распознавания образов.

Одной из важнейших задач, возникающих при создании информационных систем, является разработка алгоритмов машинного обучения.

В последнее время все большую популярность приобретает глубинное обучение (*deep learning*), заключающееся в обучении нейросетей с очень большим числом параметров. С помощью глубинных нейросетей успешно решаются различные задачи, связанные с распознаванием речи, компьютерным зрением, обработкой текстов и т. д. Нейронные сети применяются для распознавания графических объектов.

Многие задачи машинного обучения, в частности обучения нейросетей, сводятся к задаче минимизации конечной суммы измерительных функций для большого количества точек данных.

В статье рассматривается минимизация конечных сумм градиентными методами: *FG, AFG, SG, SAG, IAG, L-BFGS*. Приведено сравнение применения методов в конкретных случаях, зависящих от определенных параметров.

Общая часть. Градиентные методы – это широкий класс оптимизационных алгоритмов, используемых не только в машинном обучении.

Рассмотрим один из самых распространенных типов нейросетей – многослойную нейронную сеть. Будем считать, что объекты принадлежат пространству R^d, а ответы – пространству Z^m. Пусть многослойная нейросеть состоит из L слоев. Входной слой нейросети состоит из d нейронов $\nu_1^0, \ldots, \nu_d^0$, каждый из которых принимает значение, соответствующее одному из признаков объекта: $\nu_j^0(x) = x_j$. Последний, L-й слой, называется выходным, а слои с 1-го по (L - 1)-й скрытыми. Выходной слой состоит из m нейронов (столько же, сколько элементов в векторе ответов $z \in Z$), а s-й скрытый слой состоит из n_s нейронов. Каждый нейрон суммирует с некоторыми весами выходы всех нейронов предыдущего слоя, а затем применяет к сумме функцию активации σ:

$$v_j^s = \sigma_s\left(\sum_{k=1}^{n_s-1} w_{kj}^s v_k^{s-1}(x)\right), \tag{1}$$

где $s = 1, \ldots, L$; $j = 1, \ldots, n_s$.

Вообще говоря, каждый нейрон v_j^s может иметь собственную функцию активации σ_{sj}. Чтобы задать нейросеть, нужно настроить ее веса $\{ w_{kj}^s \}$. Для этого, минимизируем среднеквадратичную ошибку:

$$Q(X, w) = \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{m}(\sigma_j^L(x_i) - z_{ij})^2 \to \min, \tag{2}$$

где $Q(X, w)$ – функционал, задающий среднеквадратичную ошибку, σ_j^L – функция активации, применяемая к выходу j-го нейрона L- го слоя, x_i – вектор признаков обучающего объекта, z_{ij} – значение выхода на j-ом нейроне выходного слоя для i- го обучающего объекта.

Рассмотрим одно слагаемое функционала, соответствующее ошибке на одном объекте:

$$Q(x, w) = \frac{1}{2}\sum_{j=1}^{m}(v_j^L(x) - z_j)^2. \tag{3}$$

Для настройки весов требуется найти производные функционала по весам $\partial Q / \partial w_{kj}^i$ для всех нейронов, которые могут быть вычислены эффективно, если воспользоваться методом обратного распространения ошибки [1].

На практике для ускорения обучения нейросетей используют градиентные методы. Рассмотрим некоторые градиентные методы и гарантии их сходимости, которые могут быть даны для выпуклых функций.

Пусть $Q(w)$ – функционал, представимый в виде суммы n функций:

$$Q(w) = \sum_{i=1}^{n} q_i(w). \tag{4}$$

В таком виде, например, может быть представлен квадратичный функционал (3) для нейросети (1). Отдельные функции $q_i(w)$ будут соответствовать ошибкам на отдельных объектах.

Для оптимизации (4), стандартный метод детерминированного или полного градиента (*FG*) [2], использует итерации вида

$$w^{k+1} = w^k - \alpha^k Q'(w^k) = w^k - \frac{\alpha_k}{n} \sum_{i=1}^{n} q_i'(w^k), \tag{5}$$

где α_k – размер шага на итерации k. Предполагая, что минимизация w^* существует, тогда, ошибка, достигнутая на k-ой итерации метода *FG* (полного градиента) с постоянным шагом, дается выражением:

$$q(w^k) - q(w^*) = O(1/k), \tag{6}$$

где $O(1/k)$ – поведение функции $q_i(w)$, когда ее аргумент стремится к $1/k$, когда q выпукло [3].

К сожалению, метод *FG* может оказаться слишком трудоемким, когда n велико, поскольку его скорость итерации масштабируется линейно в n.

Тогда используют метод стохастического градиента *SG*, имеющий скорость итерации, не зависящую от n, что делает его пригодным там, где n может быть очень большим. Основной метод *SG* для оптимизации (4) использует итерации вида:

$$w^{k+1} = w^k a^k q'_{i_k}(w^k), \tag{7}$$

где на каждой итерации индекс i_k выбирается случайно из множества $\{1,...,n\}$.

Метод стохастического градиента имеет менее трудоемкие итерации по сравнению с полным градиентом (5), но и скорость сходимости у него существенно меньше.

Рассмотрим метод стохастического среднего градиента (*SAG*) для оптимизации суммы конечного количества гладких выпуклых функций. Как и метод стохастического градиента (*SG*), метод *SAG* по скорости итерации не зависят от числа слагаемых в сумме. Тем не менее, за счет включения в память предыдущих значений градиента метод *SAG* достигает более высокой скорости сходимости, чем метод *SG*.

Численные эксперименты показывают, что алгоритм *SAG* часто значительно превосходит существующие *SG* и детерминированные градиентные методы, и что его производительность может быть дополнительно улучшена за счет использования стратегий неравномерного отбора проб.

Итерации *SAG* принимают форму:

$$w^{k+1} = w^k - \frac{\alpha_k}{n} \sum_{i=1}^{n} q_i^k \ , \tag{8}$$

где на каждой итерации выбираем случайный индекс i_k и принимаем

$$q_i^k = \begin{cases} q'_i(w^k), & \text{если } i = i_k \\ q_i^{k-1}, & \text{иначе} \end{cases} . \tag{9}$$

То есть, имея доступ к i_k и сохраняя память о самой последней величине градиента, вычисленного для каждого индекса i, итерация *SAG* достигает высшей скорости сходимости, чем это возможно для стандартного *SG* метода. Обширные исследования по методам *SG*, предполагают, что *SAG* − первый общий метод, который достигает скорости сходимости *FG* методов, сохраняя при этом стоимость итерации стандартных методов *SG*. Численное сравнение реализации, основанной на *SAG* с конкурирующими *SG* и *FG* методами, указывает, что метод может быть очень полезным для задач, где можно позволить себе лишь сделать несколько проходов через набор данных.

Результаты работы по алгоритму *SAG* показывают, что небольшой размер шага дает медленную линейную скорость сходимости. Эта скорость сравнима со скоростью методов *FG*, с точки зрения эффективности прохода через данные, в то время как гораздо больший размер шага дает более высокую скорость сходимости, но это требует, чтобы n было достаточно велико. Еще более высокие скорости сходимости могут быть достигнуты за счет неравномерной выборки и представленные численные результаты, показывают, что это может привести к существенно улучшенной производительности [4].

Далее приводится связь между *SAG* методом и несколькими наиболее тесно связанными идеями.

Импульс. Методы *SG* , включающие термин импульс, используют итерацию в форме:

$$w^{k+1} = w^k - \alpha_k q'_{ik}(w^k) + \beta_k (w^k - w^{k+1}). \quad (10)$$

Если установить все $\beta_k = \beta$, где β – некоторая постоянная, то в этом случае можно переписать *SG* методом импульса как

$$w^{k+1} = w^k - \sum_{j=1}^{k} \alpha_j \beta^{k-j} q'_{i_j}(w^j). \quad (11)$$

Также переписать обновления *SAG* в аналогичной форме, как

$$w^{k+1} = w^k - \sum_{j=1}^{k} \alpha_k S(j, i_{1:k}) q'_{i_j}(x^j), \quad (12)$$

где функция выбора $S(j, i_{1:k})$ равна $1/n$, если j соответствует последней итерации, где $j = j_k$ и устанавливается в 0, в противном случае.

Таким образом, импульс использует геометрическое взвешивание предыдущих градиентов, в то время как итерации *SAG* выбирают в среднем самую последнюю оценку каждого предыдущего градиента.

При этом импульс может вести к улучшенному практическому выполнению, но он все еще требует использования уменьшения размеров шага и не известно, приводит ли к более высокой скорости сходимости.

Усреднение градиента. Этот метод, близко связанный с импульсом, похож на итерации *SAG* в виде (5), но в нём используются все предыдущие градиенты:

$$w^{k+1} = w^k - \frac{\alpha_k}{k} \sum_{j=1}^{k} q'_{i_j}(x^j). \quad (13)$$

Этот подход использовался в сопряженном методе усреднения [5].

Усреднение итераций. Вместо того чтобы менять составляющие в среднем градиенте, некоторые авторы предлагают выполнить базовую *SG* итерацию, но используют усреднение по определенным w_k значениям в качестве конечной оценки [6].

Стохастические варианты методов *FG*. Для ускорения сходимости метода *FG* для гладких функций используется метод ускоренного полного градиента (*AFG*) [7], а также классические методы, основанные на квадратичных приближениях, такие как диагонально-масштабируемые методы *FG*, нелинейный метод сопряженных градиентов. Значительное количество работ по разработке стохастических вариантов этих алгоритмов, показали улучшенную сходимость, связанную с использованием диагонального масштабирования, принимающую во внимание предыдущие значения градиентных величин.

С другой стороны, если разбить скорость сходимости в детерминированной и стохастической части, эти методы могут улучшить зависимость конверсионной нормы скорости от детерминированной части. Однако, неизвестен метод этого вида, улучшающий на $O(1/\sqrt{k})$ и $O(1/k)$ зависимости от стохастической части. Кроме того, многие из этих методов, как правило, требуют тщательной настройки параметров (за размером шага) и часто не в состоянии воспользоваться разреженностью в градиенте q_i'.

Постоянный размер шага. Если итерации *SG* используются для строго выпуклой оптимизации с постоянным размером шага (а не убывающей последовательности), то исследования [8] показали, что скорость сходимости метода можно разделить на две части. Первая – зависит от k и сходится линейно к 0, вторая – не зависит от k и не сходится к 0. Таким образом, с постоянным размером шага, витки *SG* имеют линейную скорость сходимости с некоторым допуском, и вообще после этого момента итерации не добиться дальнейшего прогресса. В самом деле, вплоть до последних работ [9], сходимости основного метода *SG* с постоянным размером шага были только доказаны в строго выпуклом квадратичном случае (с усредненной итерацией), а также при очень сильных предположениях об отношениях между функциями q_i [10]. Это контрастирует с методом SAG, который сходится к оптимальному решению с использованием постоянного размера шага и делает это с линейной скоростью (без дополнительных предположений).

Ускоренные методы. Ускоренными методами *SG*, к которым, несмотря на свое название, не относится вышеупомянутый метод *AFG*, можно воспользоваться для большей скорости сходимости метода *SG* с постоянным размером шага. В частности, ускоренные методы *SG* используют постоянный размер шага по умолчанию, и необходимо только уменьшить размер шага для итераций, где оценка градиента является отрицательной [11]. Это приводит к сходимости метода и позволяет ему достичь потенциально периода более быстрой сходимости, для постоянного размера шага. Тем не менее, общая скорость сходимости метода не улучшается.

Гибридные методы. Некоторые авторы предлагают варианты метода *SG* постепенно трансформировать в метод *FG* для того, чтобы добиться более высокой скорости сходимости. Метод достижения линейной скорости сходимости для сильновыпуклой квадратичной функции [12] позволяет пройти через данные циклически со специализированным весом. Тем не менее, взвешивание численно неустойчивы и представленная линейная скорость сходимости обрабатывает полные проходы через данные как итерации. Связанная с этим стратегия состоит в том, чтобы сгруппировать функции q_i в «партии» увеличения размера и выполнять итерации *SG* на пакетах.

Инкрементальный ускоренный градиент (*IAG*). В работе [13] представлен алгоритм метода *IAG* наиболее тесно связанный с *SAG* алгоритмом. Этот метод идентичен итерации *SAG* (5), но использует циклический выбор i_k, а не

выборки i_k значения. Это различие имеет несколько важных последствий. В частности показано, что скорость сходимости является линейной для сильно выпуклых квадратичных функций (без получения явной скорости), а также анализируется полный проход через итерации.

Кроме того, как показывают анализы и эксперименты, витки *SAG* требуют значительно большего размера шага, чем требуется для сходимости метода *IAG*. Это приводит к еще большей робастности для выбора размера шага, а также, если шаг соответствующим образом выбран, приводит к более высокой скорости сходимости и существенно улучшает практическую производительность. Это показывает, что простое изменение (случайный выбор против цикла) может значительно повысить производительность оптимизации.

Сравнение *FG* и *SG* методов. Теоретически по скорости сходимости, можно предложить следующие стратегии для решения того, использовать *FG* или *SG* метод:

- если достаточно сделать только один проход через данные, то следует использовать метод *SG*;

- если можно сделать много проходов через данные (скажем, несколько сотен), тогда должен быть использован метод *FG*.

Итерации *SAG* будет наиболее полезными между этими двумя методами, в них можно сделать больше, чем один проход по данным, но можно не сделать достаточно проходов, чтобы гарантировать какой алгоритм использовать *FG* или *AFG* или метод *L-BFGS*. Чтобы проверить, так ли это на самом деле, на практике, проводились различные эксперименты для оценки эффективности алгоритма *SAG*.

Сравнивался следующий ряд конкурентных *FG* и *SG* методов.

• *AFG*: Вариант ускоренного метода полного градиента, где итерации (3) с размером шага $1/L^k$ пересекаются с шагом экстраполяции. Использовался адаптивный line-поиск, чтобы оценить локальный L в расчете на вариант, предложенный для l_2-регулированной логистической регрессии по [14].

• *L-BFGS*: Открыто доступный квазиньютоновский метод ограниченной памяти, построенный для логарифмической линейной модели логистической регрессии. Этот метод является наиболее сложным методом.

• *SG*: Метод стохастического градиента, описываемый итерацией (4). Так как установка размера шага в этом методе является шатким вопросом, был выбран постоянный размер шага, который дал лучшую производительность (обнаружено, что стратегия постоянного размера шага дала лучшую производительность по сравнению с разнообразием стратегий уменьшения размера шага).

• *ASG*: Метод ускоренного стохастического градиента. Среднее значение для итераций, полученных методом *SG* выше, если выбрать размер шага среди степеней 10.

• *LAG*: Добавочный агрегированный метод градиента описывается итерацией (5) с циклическим выбором i_k. Использовались повторное взвешивание, точ-

ная регуляризация, размер шага среди всех степеней 10, который дал лучшую производительность.

• *SAG-LS*: Предложенный метод стохастического среднего градиента описывается итерацией (5), и был использован размер шага $\alpha_k = 1/L^k$; где L^k – приблизительная константа Липшица.
http://www.di.ens.fr/mschmidt/Software/minFunc.html.

Можно наблюдать несколько тенденций в этих экспериментах.

• *FG* против *SG*: На первых нескольких проходах через данные методы *SG* (*SG* и *ASG*) всегда лучше, чем методы *FG* (*AFG* и *L-BFGS*), если размер шага выбран тщательно.

• (*FG* и *SG*) по сравнению с *SAG*: Итерации *SAG*, достигают лучшего из обоих способов. Они начинают существенно лучше, чем *FG* методы, часто получая аналогичные показатели с *SG* методом с лучшим ступенчатым размером шага, выбранным задним числом. Но итерации *SAG* продолжают делать устойчивый прогресс даже после первых нескольких проходов через данные. Это приводит к более высокой производительности, чем методы *SG* на более поздних итерациях, и на большинстве наборах данных методы *FG* уступают методу *SAG* даже после 50 проходов по данным.

• *IAG* против *SAG*. Выбор точек данных в произвольном порядке (*SAG*) по сравнению с циклическим проходом через данные (*IAG*) повышает производительность *SAG* над *IAG*, (даже если методу *IAG* было позволено выбрать лучший размер шага в ретроспективе). Это происходит из-за больших размеров шагов, разрешенных итерациям *SAG*, которые могли бы вызвать расхождение итерации *IAG*.

Выводы. Проведенный анализ показывает, что существует большое количество градиентных методов для оптимизации обучения нейронных сетей. Обучение нейронной сети характеризуется четырьмя специфическими ограничениями, выделяющими обучение нейросетей из общих задач оптимизации: астрономическое число параметров, необходимость высокого параллелизма при обучении, многокритериальность решаемых задач, необходимость найти достаточно широкую область, в которой значения всех минимизируемых функций близки к минимальным значениям. В остальных случаях проблему обучения можно, как правило, сформулировать как задачу минимизации оценки.

Осторожность предыдущей фразы («как правило») связана с тем, что на самом деле нам неизвестны и никогда не будут известны все возможные задачи для нейронных сетей, и, быть может, в неизвестности есть задачи, которые несводимы к минимизации оценки.

Минимизация оценки – сложная проблема: параметров астрономически много (для стандартных примеров, реализуемых на ПК – от 100 до 1000000), адаптивный рельеф (график оценки как функции от подстраиваемых параметров) сложен, может содержать много локальных минимумов.

1. Хайкин С. Нейронные сети: Полный курс. Пер. с англ. Н. Н. Куссуль, А. Ю. Шелестова. 2-е изд., испр. М.: Издательский дом Вильямс, 2008. 1103 с.
2. Cauchy M.A. Metode generale pour la resolution des systemes d'equations simultanees. Comptesrendus des'séances del Academie des sciences de Paris. 1847. N 25. P. 536 – 538.
3. Nesterov Y. Introductory lectures on convex optimization: A basic course. Springer, 2004.
4. Le Roux N., Schmidt M., Bach F. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with nite training sets. Advances in Neural Information Processing System. 2012.
5. Nesterov Y. Primal-dual subgradient methods for convex problems. Mathematical programming. 2009. N 120 (1). P. 221 − 259.
6. Polyak B.T., Juditsky A.B. Acceleration of stochastic approximation by averaging. SIAM Journal on Control and Optimization. 1992. N 30(4). P. 838 − 855.
7. Nesterov Y. A method for unconstrained convex minimization problem with the rate of convergencen $O(1/k^2)$. Doklady AN USSR. 1983. N 269(3). P. 543 − 547.
8. Bordes A., Bottou L., Gallinari P. Careful quasi-newton stochastic gradient descent. Journal of Machine Learning Research. 2009. N 10. P. 1737 − 1754.
9. Hu C., Kwok J., Pan W. Accelerated gradient methosd for stochastic optimization and online learning. Advances in Neural Information Processing Systems. 2009.
10. Sunehag P., Trumpf J., Vishwanathan S., Schraudolph N. Variable metric stochastic approximation theory. International Conference on Articial Intelligence and Statistics. 2009.
11. Martens J. Deep learning via Hessian-free optimization. International Conference on Machine Learning. 2010.
12. Xiao L. Dual averaging methods for regularized stochastic learning and online optimization. Journal of Machine Learning Research. 2010. N 11. P. 2543 − 2596.
13. Nedic A., Bertsekas D. Convergence rate of incremental subgradient algorithms in Stochastic Optimization. Algorithms and Applications. Kluwer Academic. 2000. P. 263 − 304.
14. Bach F., Moulines E. Non-strongly-convex smooth stochastic approximation with convergence rate O(1=n). Arhiv preprint. 2013.

Получено 15.09.2017