Быстрый инкрементальный метод оптимизации больших сумм функций с суперлинейной скоростью сходимости

А. О. Родоманов   Д. А. Кропотов

МГУ им. М. В. Ломоносова, Москва

ММРО, 2015

- Задача минимизации ℓ_2-регуляризованного эмпирического риска:

$$\min_{\mathbf{w}\in\mathbb{R}^D} \quad \left[F(\mathbf{w}) := \frac{1}{N}\sum_{i=1}^{N} f_i(\mathbf{w}) + \frac{\lambda}{2}\|\mathbf{w}\|_2^2 \right] \qquad (1)$$

где $\lambda > 0$ – коэффициент регуляризации.

  - Например, логистическая регрессия:

$$f_i(\mathbf{w}) := \ln(1+\exp(-y_i\mathbf{w}^\top\mathbf{x}_i)) \qquad (2)$$

- Случай «больших данных»:

$$N \gg 1.$$

- Предположения:
  - все функции f_i дважды непрерывно дифференцируемы и выпуклы
  - гессианы $\boldsymbol{\nabla}^2 f_i$ удовлетворяют условию Липшица:

$$\left\|\boldsymbol{\nabla}^2 f_i(\mathbf{w}) - \boldsymbol{\nabla}^2 f_i(\mathbf{u})\right\|_2 \le M\,\|\mathbf{w}-\mathbf{u}\|_2, \qquad \forall \mathbf{w},\mathbf{u}\in\mathbb{R}^D.$$

Стохастический градиентный спуск (SGD) [Robbins and Monro, 1951]

Итерация метода:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \alpha_k(\nabla f_{i_k}(\mathbf{w}_k) + \lambda \mathbf{w}_k), \tag{3}$$

где $i_k \in \{1, \ldots, N\}$ – случайно выбираемый номер компоненты.

Достоинства:

- Низкая стоимость итерации ($O(D)$) и требования по памяти ($O(D)$);
- Минимальные требования к $F(\mathbf{w})$ и $f_i(\mathbf{w})$;
- Простота реализации.

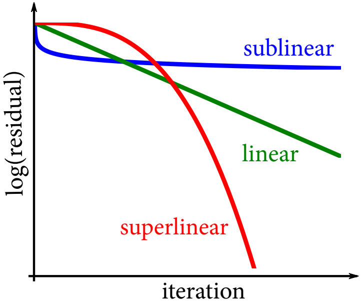
Недостатки:

- Необходимость тонкой настройки параметров (стратегия уменьшения α_k, коэффициент momentum, размер мини-батча, параметры условия останова и др.)
- Скорость сходимости: сублинейная, $O(1/k)$.

Скорости сходимости методов оптимизации

Невязка $r_k := F(\mathbf{w}_k) - F(\mathbf{w}_*)$.
Скорость сходимости:

- Сублинейная: $r_k \to 0$;
- Линейная: $r_{k+1} \leq c r_k$ для некоторого $0 < c < 1$;
- Суперлинейная: $r_{k+1} \leq c_k r_k$ и $c_k \to 0$;
- Квадратичная: $r_{k+1} \leq c r_k^2$.



Стохастический средний градиент (SAG) [Schmidt et al., 2013]

Шаг метода:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \alpha(\mathbf{g}_k + \lambda \mathbf{w}_k), \tag{4}$$

где $\mathbf{g}_k$ – «средний» градиент:

$$\mathbf{g}_k = \frac{1}{N}\sum_{i=1}^{N} \nabla f_i(\mathbf{v}_i^k), \tag{5}$$

который обновляется в итерациях как:

$$\mathbf{g}_k = \mathbf{g}_{k-1} + \frac{1}{N}\left(\boldsymbol{\nabla} f_{i_k}(\mathbf{w}_k) - \mathbf{y}_{i_k}^{k-1}\right). \tag{6}$$

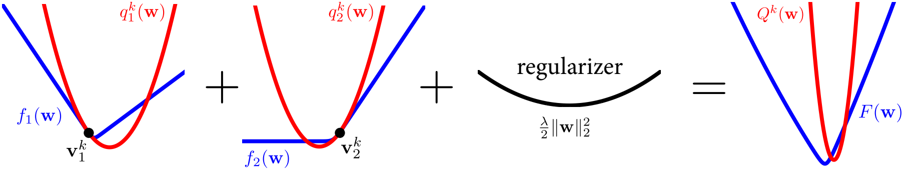
- Память: $O(ND)$ для хранения $\mathbf{y}_i^k := \boldsymbol{\nabla} f_i(\mathbf{v}_i^k)$, где $\mathbf{v}_i^k$ = последняя точка, в которой вычислялась f_i.
- Скорость сходимости: линейная, $O(\rho^k)$, где $\rho \in (0, 1)$.

- Квадратичная модель одного слагаемого f_i с центром в $\mathbf{v}_i^k$:

$$q_i^k(\mathbf{w}) := f_i(\mathbf{v}_i^k) + \nabla f_i(\mathbf{v}_i^k)^\top (\mathbf{w} - \mathbf{v}_i^k) + \frac{1}{2}(\mathbf{w} - \mathbf{v}_i^k)^\top \nabla^2 f_i(\mathbf{v}_i^k)(\mathbf{w} - \mathbf{v}_i^k). \qquad (7)$$

- Модель полной функции F:

$$Q^k(\mathbf{w}) := \frac{1}{N}\sum_{i=1}^{N} q_i^k(\mathbf{w}) + \frac{\lambda}{2}\|\mathbf{w}\|_2^2 . \qquad (8)$$



Инкрементальный метод Ньютона (NIM)

- Квадратичная модель одного слагаемого f_i с центром в $\mathbf{v}_i^k$:
  $q_i^k(\mathbf{w}) := f_i(\mathbf{v}_i^k) + \nabla f_i(\mathbf{v}_i^k)^\top(\mathbf{w}-\mathbf{v}_i^k) + \frac{1}{2}(\mathbf{w}-\mathbf{v}_i^k)^\top \nabla^2 f_i(\mathbf{v}_i^k)(\mathbf{w}-\mathbf{v}_i^k).$
- Модель полной функции F:

$$Q^k(\mathbf{w}) := \frac{1}{N}\sum_{i=1}^{N} q_i^k(\mathbf{w}) + \frac{\lambda}{2}\|\mathbf{w}\|_2^2 .$$

- Итерация метода:
  - Выбрать номер компоненты $i_k \in \{1, \ldots, N\}$.
  - Обновить модель только для одной компоненты:
    $\mathbf{v}_{i_k}^k := \mathbf{w}_k, \quad \mathbf{v}_i^k := \mathbf{v}_i^{k-1},\ i \neq i_k.$
  - Найти минимум модель полной функции:
    $\bar{\mathbf{w}}_k := \operatorname{argmin}_{\mathbf{w}\in\mathbb{R}^D} Q^k(\mathbf{w}).$
  - Сделать шаг в направлении минимума модели:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k(\bar{\mathbf{w}}_k - \mathbf{w}_k), \tag{9}$$

    где $\alpha_k > 0$ – длина шага.

- Минимум модели:

$$\bar{\mathbf{w}}_k = (\mathbf{H}_k + \lambda \mathbf{I})^{-1}(\mathbf{p}_k - \mathbf{g}_k), \tag{10}$$

где

$$\mathbf{H}_k := \frac{1}{N}\sum_{i=1}^{N} \nabla^2 f_i(\mathbf{v}_i^k), \ \mathbf{p}_k := \frac{1}{N}\sum_{i=1}^{N} \nabla^2 f_i(\mathbf{v}_i^k)\mathbf{v}_i^k, \ \mathbf{g}_k := \frac{1}{N}\sum_{i=1}^{N} \nabla f_i(\mathbf{v}_i^k). \tag{11}$$

- Обновление модели по схеме «прибавить-вычесть»:

$$\begin{aligned} \mathbf{H}_k &= \mathbf{H}_{k-1} + \frac{1}{N}\left(\nabla^2 f_{i_k}(\mathbf{w}_k) - \nabla^2 f_{i_k}(\mathbf{v}_{i_k}^{k-1})\right), \\ \mathbf{p}_k &= \mathbf{p}_{k-1} + \frac{1}{N}\left(\nabla^2 f_{i_k}(\mathbf{w}_k)\mathbf{w}_k - \nabla^2 f_{i_k}(\mathbf{v}_{i_k}^{k-1})\mathbf{v}_{i_k}^{k-1}\right), \\ \mathbf{g}_k &= \mathbf{g}_{k-1} + \frac{1}{N}\left(\nabla f_{i_k}(\mathbf{w}_k) - \nabla f_{i_k}(\mathbf{v}_{i_k}^{k-1})\right), \end{aligned} \tag{12}$$

где $i_k \in \{1, \ldots, N\}$ – номер обновляемой компоненты.

- Сложность итерации: $O(D^3)$ для решения СЛАУ.
- Память: $O(ND + D^2)$ для хранения $\mathbf{H}_k$ и всех центров $\mathbf{v}_i^k$.

Модификация метода NIM на случай линейных моделей

- Линейные модели: $f_i(\mathbf{w}) := \phi_i(\mathbf{x}_i^\top \mathbf{w})$ для некоторого $\mathbf{x}_i \in \mathbb{R}^D$
- Градиенты и гессианы имеют специальную структуру:

$$\begin{aligned} \nabla f_i(\mathbf{w}) &= \phi_i'(\mathbf{x}_i^\top \mathbf{w})\mathbf{x}_i, \\ \nabla^2 f_i(\mathbf{w}) &= \phi_i''(\mathbf{x}_i^\top \mathbf{w})\mathbf{x}_i\mathbf{x}_i^\top. \end{aligned} \tag{13}$$

- Вместо сохранения центра $\mathbf{v}_i^k$, можно хранить только результат скалярного произведения:

$$\mu_i^k := \mathbf{x}_i^\top \mathbf{v}_i^k. \tag{14}$$

- Нет необходимости решать СЛАУ, обновление $\mathbf{B}_k := (\mathbf{H}_k + \lambda \mathbf{I})^{-1}$:

$$\mathbf{B}_k = \mathbf{B}_{k-1} - \frac{\delta_k \mathbf{B}_{k-1}\mathbf{x}_{i_k}\mathbf{x}_{i_k}^\top \mathbf{B}_{k-1}}{N + \delta_k \mathbf{x}_{i_k}^\top \mathbf{B}_{k-1}\mathbf{x}_{i_k}}, \tag{15}$$

  где $\delta_k := \phi_{i_k}''(\mu_{i_k}^k) - \phi_{i_k}''(\mu_{i_k}^{k-1})$.
- Стоимость итерации: $O(D^2)$ вместо $O(D^3)$.
- Память: $O(N + D^2)$ вместо $O(ND + D^2)$.

Скорость сходимости метода NIM

Теорема (локальная скорость сходимости)

- *Пусть все центры инициализированы в окрестности оптимума* $\mathbf{w}_*$*:*
$$\left\|\mathbf{v}_i^0 - \mathbf{w}_*\right\|_2 \le \frac{2\lambda}{M\sqrt{N}}. \qquad (16)$$
- *Предположим, что используется единичный шаг* $\alpha_k \equiv 1$.

Тогда $\{\mathbf{w}_k\}$ *сходится к* $\mathbf{w}_*$ *с R-суперлинейной скоростью сходимости:*
$$\|\mathbf{w}_k - \mathbf{w}_*\|_2 \le r_k \qquad \text{and} \qquad \lim_{k\to\infty} \frac{r_{k+1}}{r_k} = 0.$$
Кроме того, $\{\mathbf{w}_k\}$ *также сходится R-квадратично по эпохам (каждую* N*-ю итерацию):*
$$r_{k+N} \le \frac{M}{2\lambda} r_k^2, \qquad k = 2N, 2N+1, \ldots .$$

Сравнение с другими методами

Функция: $F(\mathbf{w}) := (1/N) \sum_{i=1}^{N} \phi_i(\mathbf{x}_i^\top \mathbf{w}) + (\lambda/2) \|\mathbf{w}\|_2^2$.

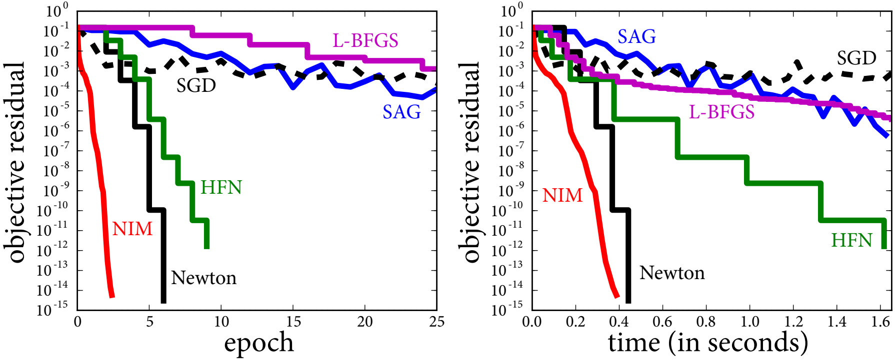
Метод	Стоимость итерации	Память	Скорость сходимости	
			По итерациям	По эпохам
SGD	$O(D)$	$O(D)$	Сублинейная	Сублинейная
SAG	$O(D)$	$O(N + D)$	Линейная	Линейная
NIM	$O(D^2)$	$O(N + D^2)$	Суперлинейная	Квадратичная

Обозначения:

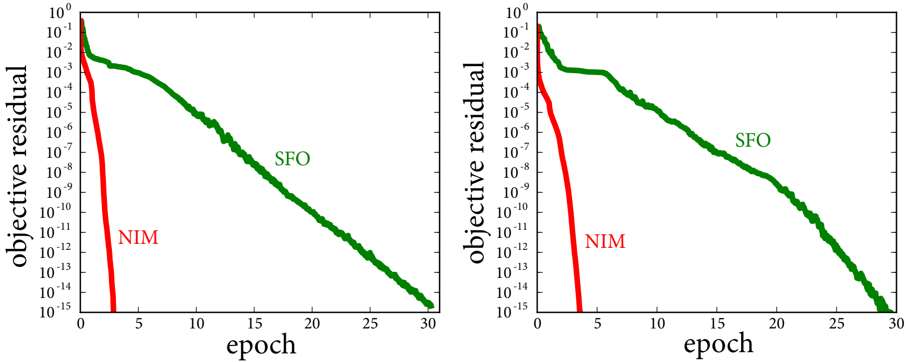
- N = кол-во слагаемых;
- D = кол-во оптимизируемых переменных;
- Одна эпоха $= N$ итераций.
- SGD = стохастический градиентный спуск.
- SAG = стохастический средний градиент [Schmidt et al., 2013].

Эксперименты: небольшое N

- Функционал: ℓ_2-регуляризованная логистическая регрессия.
- Данные *quantum* (25 MB; $N = 50\,000$, $D = 65$):

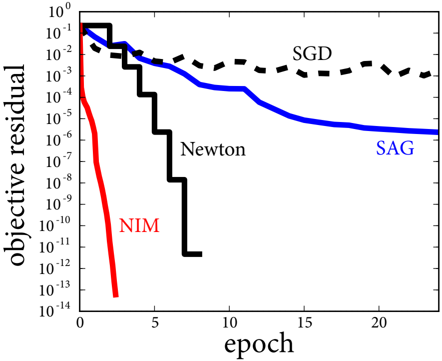


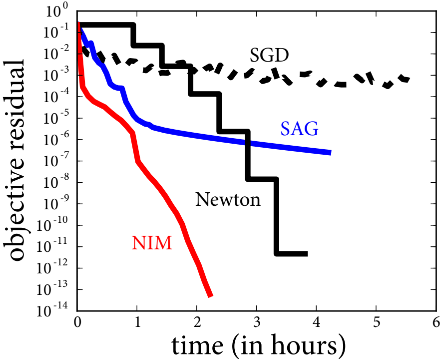
- Данные *a9a* ($N = 32\,561$, $D = 125$) и *covtype* ($N = 581\,012$, $D = 54$).
- Сравнение с SFO [Sohl-Dickstein et al., 2014]:



Эксперименты: большие данные #1

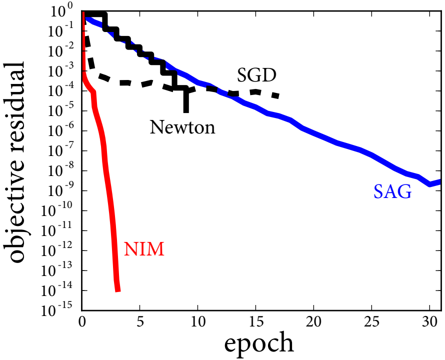
- Dataset *mnist8m* (47 GB; $N = 8\,100\,000$, $D = 784$):

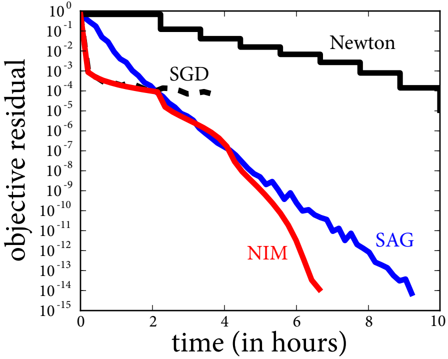




Эксперименты: большие данные #2

- Данные *dna18m* (107 GB; $N = 18\,000\,000$, $D = 800$):





Выводы:

- Предложен новый инкрементальный метод оптимизации с суперлинейной скоростью сходимости;
- Настройка параметров не требуется;
- Эффективная адаптация для случая линейных моделей;
- На практике метод всегда сходится за 3–5 эпох;
- При небольшом количестве переменных опережает многие другие методы;
- При большом количестве переменных характеристики метода значительно снижаются.

Планы на будущее:

- Доказательство глобальной сходимости метода;
- Адаптация метода для других «простых» регуляризаторов $\Omega(\mathbf{w})$.