Курс: Методы оптимизации в машинном обучении, 2012

Базовые методы многомерной оптимизации

Рассмотрим задачу безусловной оптимизации в многомерном пространстве:

$$f(\boldsymbol{x}) \to \min_{\boldsymbol{x} \in \mathbb{R}^N}. \tag{1}$$

Обозначим её решение через $\boldsymbol{x}_* \in \mathbb{R}^N$. Общая идея решения задачи (1) состоит в рассмотрении последовательности задач одномерной оптимизации, каждая из которых может быть эффективно решена одним из методов одномерной оптимизации. Пусть известно некоторое начальное приближение $\boldsymbol{x}_0$. Тогда на основании информации от оракула в точке $\boldsymbol{x}_0$ метод многомерной оптимизации строит некоторое направление $\boldsymbol{d}_0 \in \mathbb{R}^N$ и решает задачу вида

$$\alpha_0 = \arg\min_{\alpha \geq 0} f(\boldsymbol{x}_0 + \alpha \boldsymbol{d}_0),\ \alpha \in \mathbb{R}.$$

В результате находится новая точка $\boldsymbol{x}_1 = \boldsymbol{x}_0 + \alpha_0 \boldsymbol{d}_0$. С помощью информации от оракула в точке $\boldsymbol{x}_1$ и, быть может, информации с предыдущей итерации, метод строит новое направление оптимизации $\boldsymbol{d}_1$, решает одномерную задачу оптимизации функции f вдоль выбранного направления и таким образом находит новую точку $\boldsymbol{x}_2$. Данный процесс повторяется до тех пор, пока не будет выполнен один из критериев останова, например, достигнута сходимость по аргументу $\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\| < \varepsilon$, сходимость по функции $|f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k-1})| < \varepsilon$ или проведено максимально допустимое количество итераций. В результате различные методы многомерной оптимизации отличаются стратегией выбора очередного направления $\boldsymbol{d}_k$, а также, возможно, ограничениями на выбор метода одномерной оптимизации.

Покоординатный спуск

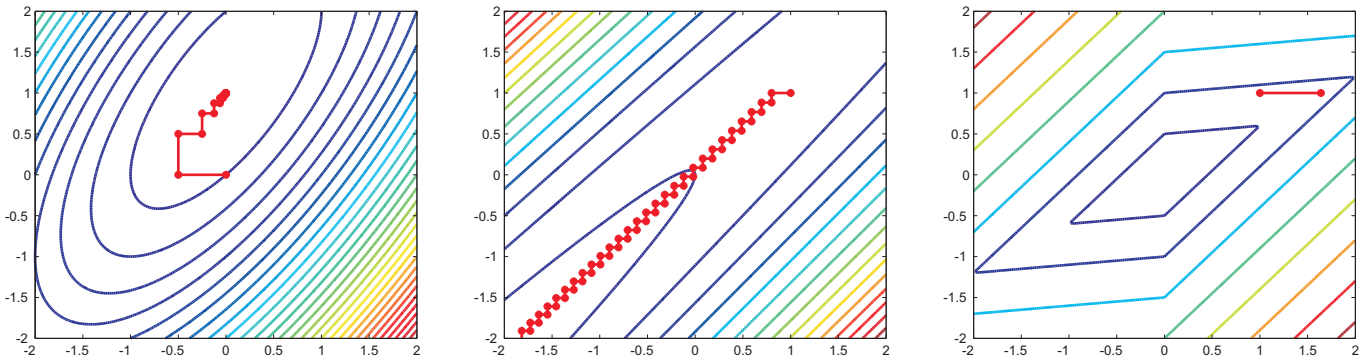


Рис. 1: Примеры работы метода покоординатного спуска. Случай (a): быстрая сходимость метода (всего 34 итерации для точности по аргументу 10^{-5}), случай (b): медленная сходимость метода (всего 1455 итераций для точности по аргументу 10^{-5}), случай (c): остановка метода в промежуточной точке.

В методе покоординатного спуска в качестве очередного направления $\boldsymbol{d}_k$ выбирается одна из координат $\boldsymbol{d}_k = [0, \ldots, 0, 1, 0, \ldots, 0]^T$. При этом координаты перебираются последовательно либо в случайном порядке. На этапе одномерной оптимизации вдоль очередной координаты можно использовать как точные методы, так и методы неточной оптимизации (например, backtracking или метод Флетчера). Применение метода покоординатного спуска является особенно эффективным в ситуациях, когда задача одномерной оптимизации может быть решена аналитически. Пример работы метода для задачи минимизации квадратичной функции вида $f(x_1, x_2) = 2x_1^2 - 2x_1x_2 + x_2^2 + 2x_1 - 2x_2$ из начального приближения $\boldsymbol{x}_0 = [0, 0]^T$ показан на рис. 1,a.

В методе покоординатного спуска значение оптимизируемой функции f не увеличивается на каждой итерации. При дополнительном предположении об ограниченности f снизу на $\mathbb{R}^N$ можно показать, что для непрерывно-дифференцируемых функций метод покоординатного спуска гарантированно сходится к локальному минимуму. Более того, для строго выпуклых функций f метод сходится с линейной скоростью [1].

Достоинствами метода покоординатного спуска являются:

- Простота вычисления направлений $\boldsymbol{d}_k$. В результате метод может применяться, в том числе, в пространствах сверх-большой размерности;

- Отсутствие требований к вычислению любых производных функции f. В результате метод может применяться для случая оракула нулевого порядка.

К недостаткам метода следует отнести:

- Возможность застревания в промежуточной точке для негладких функций f. Пример такой ситуации для функции $f(x_1, x_2) = \max(|x_2 - 1.1x_1|, |x_2 - 0.1x_1|)$ показан на рис. 1,c;
- Возможность совершать большое количество очень маленьких шагов даже при оптимизации строго выпуклых функций. Пример подобной ситуации для функции $f(x_1, x_2) = 5x_1^2 - 9x_1x_2 + 4.075x_2^2 + x_1$ показан на рис. 1,b. Такая ситуация частично связана с неспособностью метода идти по диагонали.

Градиентный спуск

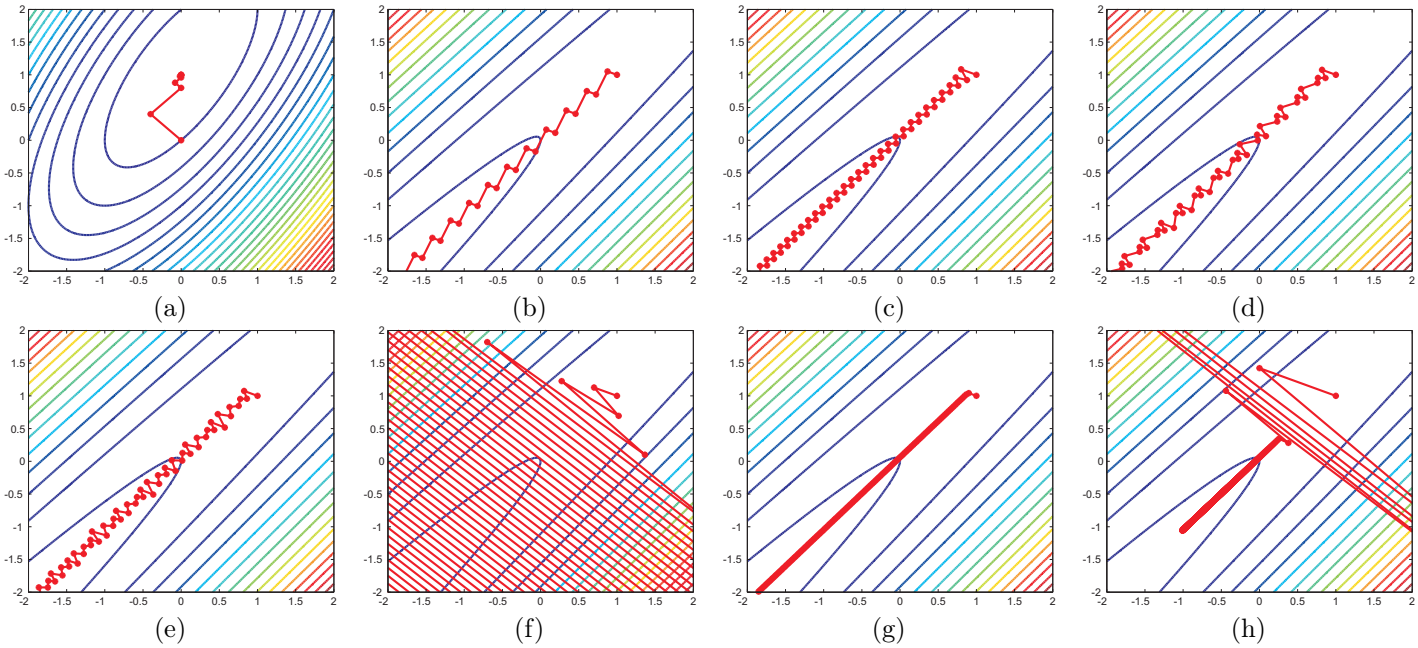


Рис. 2: Примеры работы метода градиентного спуска. Случай (a): быстрая сходимость метода (всего 14 итераций для точности по аргументу 10^{-5}), случай (b): медленная сходимость метода с точной одномерной оптимизацией (всего 529 итераций для точности по аргументу 10^{-5}, в среднем 15 обращений к оракулу на одной итерации), случай (c): работа метода с оптимизацией по методу Флетчера (3051 итерация, 4 обращения к оракулу на итерацию), случай (d): работа метода с использованием backtracking (2350 итераций, 2 обращения к оракулу на итерацию), случай (e): работа метода с адаптивной стратегией выбора длины шага (2878 итераций, 2 обращения к оракулу на итерацию), случай (f): расходимость метода при использовании фиксированной длины шага 0.15, случай (g): сходимость метода при использовании фиксированной длины шага 0.05 (5923 итерации), случай (h): фактическое застопоривание метода при использовании уменьшающейся длины шага.

Вопрос диагонального движения решается в методе градиентного спуска. Известно, что градиент $\nabla f(\boldsymbol{x})$ определяет направление наибольшего роста функции в точке $\boldsymbol{x}$. Поэтому в качестве стратегии выбора направления $\boldsymbol{d}_k$ разумно выбрать $-\nabla f(\boldsymbol{x}_k)$ (так как функция минимизируется). Если на каждой итерации задача одномерной оптимизации решается точно, то такой метод называется методом наискорейшего спуска. Нетрудно убедиться, что в этом методе соседние направления $\boldsymbol{d}_{k+1}, \boldsymbol{d}_k$ всегда ортогональны. Действительно,

$$0 = \frac{\partial}{\partial \alpha} f(\boldsymbol{x}_k + \alpha \boldsymbol{d}_k)\Big|_{\alpha=\alpha_k} = \nabla f(\boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k)^T \boldsymbol{d}_k = \boldsymbol{d}_{k+1}^T \boldsymbol{d}_k.$$

Пример работы метода наискорейшего спуска показан на рис. 2,a. Заметим, что в данном случае для минимизации функции понадобилось 14 итераций против 34 итераций, которые требовались методу покоординатного спуска для решения той же задачи (рис. 1,a).

Наряду с точной оптимизацией по α на каждом шаге, в методе градиентного спуска можно пользоваться различными стратегиями неточной одномерной оптимизации. Рассмотрим некоторые из них:

- Метод Флетчера, в котором выбирается значение α, удовлетворяющее двум условиям:

1. $f(\boldsymbol{x}_k - \alpha \boldsymbol{d}_k) \leq f(\boldsymbol{x}_k) - \rho\alpha\nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k$, $\rho \in (0, 1/2)$,
2. $0 \geq \nabla f(\boldsymbol{x}_k - \alpha \boldsymbol{d}_k) \geq \sigma \nabla f(\boldsymbol{x}_k)$, $\sigma \in (0, 1)$.

Пример работы метода градиентного спуска с методом Флетчера показан на рис. 2,c. Заметим, что по сравнению с наискорейшим спуском количество обращений к оракулу на каждой итерации снижается до среднего значения 4 (против 15 у наискорейшего спуска). Однако, общее количество итераций увеличивается.

- Backtracking. В этом методе выбирается некоторое фиксированное заранее значение α, которое затем сокращается в цикле на некоторую величину $v > 1$ до тех пор, пока не будет выполнено первое из условий для метода Флетчера. Такой подход позволяет еще больше сократить среднее число обращений к оракулу на каждой итерации (см. рис. 2,d).

- Адаптивная коррекция. В этом методе на текущей итерации из точки $\boldsymbol{x}_k$ делается два шага: с длиной α и с длиной αv, где $v > 1$. Если удлинённый шаг приводит к лучшему значению f, то $\alpha := \alpha v$. Если лучшим является шаг с текущей длиной, то α не меняется. Наконец, если оба шага приводят к увеличению f по сравнению с $f(\boldsymbol{x}_k)$, то α в цикле уменьшается на v до тех пор, пока новое значение f не станет меньше, чем $f(\boldsymbol{x}_k)$. При таком подходе к выбору α в штатном режиме требуется всего два обращения к оракулу на каждой итерации. Пример работы метода градиентного спуска с адаптивной коррекцией показан на рис. 2,e.

- Фиксированный шаг. Здесь $\alpha = \alpha_0$, где α_0 задается пользователем заранее. На рис. 2,f показан пример работы метода градиентного спуска с шагом $\alpha = 0.15$. Легко заметить, что в этом случае метод расходится. При сокращении длины шага до $\alpha = 0.05$ (см. рис. 2,g) метод начинает стабильно сходится, но количество требуемых итераций при этом доходит до 6 тыс.

- Уменьшающийся шаг. Здесь $\alpha_k = \alpha_0/k$, где k – номер итерации. На рис. 2,h показан пример работы метода для $\alpha_0 = 0.15$. Заметим, что на первых итерациях метод ведет себя нестабильно по аналогии с предыдущим случаем. Однако, за счет редукции длины шага, с некоторой итерации метод стабилируется. При этом, однако, длина шага становится настолько малой, что в итоге метод не сходится даже за 20 тыс. итераций.

В результате можно заключить, что последние две стратегии выбора длины шага являются не слишком удачными, т.к. в них не проводится проверок на уменьшение значения функции в итерациях. Первые три подхода, а также стратегия точной одномерной оптимизации могут применяться в зависимости от ограничений на допустимое количество обращений к оракулу.

Для метода наискорейшего спуска можно доказать следующую теорему о скорости сходимости [2]:

Теорема. Предположим, что $f \in C_M^{2,2}$ и гессиан f в оптимальной точке $\boldsymbol{x}_*$ удовлетворяет соотношению $lI \preceq \nabla^2 f(\boldsymbol{x}_*) \preceq LI$ для некоторых констант $l, L > 0$. Тогда из начальной точки $\boldsymbol{x}_0$:

$$r_0 = \|\boldsymbol{x}_0 - \boldsymbol{x}_*\| < \bar{r} = \frac{2l}{M}$$

метод наискорейшего спуска сходится с линейной скоростью:

$$\|\boldsymbol{x}_k - \boldsymbol{x}_*\| \leq \frac{\bar{r} r_0}{\bar{r} - r_0} \left(\frac{L}{L+l}\right)^k.$$

Класс функций $C_M^{2,2}$ состоит из дважды непрерывно-дифференцируемых функций, для которых вторая производная является Липшиц-непрерывной с константой M, т.е.

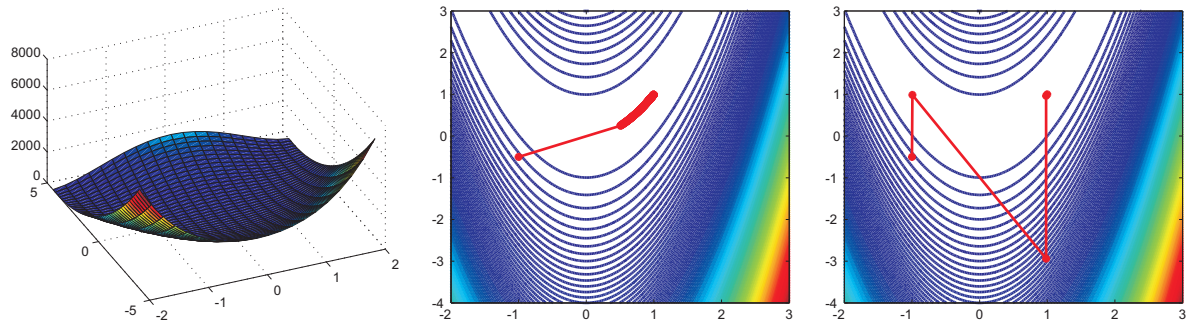
$$\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\| \leq M\|\boldsymbol{x} - \boldsymbol{y}\|, \ \forall \boldsymbol{x}, \boldsymbol{y}.$$

Условие $lI \preceq \nabla^2 f(\boldsymbol{x}_*) \preceq LI$ означает, что $l = \lambda_{min}(\nabla^2 f(\boldsymbol{x}_*))$, $L = \lambda_{max}(\nabla^2 f(\boldsymbol{x}_*))$, где $\lambda_{min}, \lambda_{max}$ – соответственно минимальное и максимальное собственное значение матрицы гессиана.

Для квадратичных функций $M = 0$. Поэтому при дополнительном требовании строгой положительной определённости гессиана квадратичной функции метод наискорейшего спуска сходится с линейной скоростью из любого начального приближения. Однако, количество итераций метода зависит от константы $L/(L+l)$. Если l и L близки между собой (линии уровня квадратичной функции близки к окружностям), то метод наискорейшего спуска сходится максимально быстро. Для примера на рис. 2,a $L \simeq 5l$, $L/(L+l) \simeq 0.87$ и метод сходится за 14 итераций. С другой стороны, для примера на рис. 2,b $L \simeq 600l$, $L/(L+l) \simeq 0.9985$. В результате метод сходится только за 529 итераций. Таким образом, изменение шкалы отдельных координат $\boldsymbol{x}$ может существенно влиять на скорость сходимости метода наискорейшего спуска.

Рассмотрим функцию Розенблока $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ (см. рис. 3,a), которая достигает минимума в точке $\boldsymbol{x}_* = [1, 1]^T$. Рассмотрим небольшую окрестность оптимальной точки $x_1 \in [1/2, 3/2], x_2 \in [1/2, 3/2]$

и оценим численно значение радиуса сходимости $\bar{r} = 2l/M$ из теоремы для метода наискорейшего спуска. Оказывается, что в этом случае $M = 363205.5, l = 0.4, \bar{r} \simeq 2.2 \cdot 10^{-6}$. Таким образом, для данного примера теорема не дает практически осмысленной оценки для радиуса сходимости. При этом, как будет показано ниже, использование шагов по градиенту при минимизации функции Розенблока является оправданным с практической точки зрения даже в точках, существенно отстоящих от оптимальной $\boldsymbol{x}_*$.



(a) (b) (c)

Рис. 3: Функция Розенблока. Случай (a) – трехмерный вид функции, случай (b) – пример работы метода наискорейшего спуска (всего 4000 итераций), случай (c) – пример работы метода Ньютона (всего 5 итераций).

Метод Ньютона

Метод Ньютона основан на использовании квадратичной аппроксимации функции f в окрестности текущей точки $\boldsymbol{x}_k$:

$$f(\boldsymbol{x}_k + \boldsymbol{d}_k) \simeq f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k + \frac{1}{2} \boldsymbol{d}_k^T \nabla^2 f(\boldsymbol{x}_k) \boldsymbol{d}_k \to \min_{\boldsymbol{d}_k}.$$

Здесь и далее будем обозначать $\boldsymbol{g}_k = \nabla f(\boldsymbol{x}_k)$, $H_k = \nabla^2 f(\boldsymbol{x}_k)$. Приравнивая к нулю градиент квадратичной аппроксимации, получаем:

$$\nabla_{\boldsymbol{d}_k} = \boldsymbol{g}_k + H_k \boldsymbol{d}_k = 0 \;\Rightarrow\; H_k \boldsymbol{d}_k = -\boldsymbol{g}_k \;\Rightarrow\; \boldsymbol{d}_k = -H_k^{-1} \boldsymbol{g}_k.$$

Таким образом, один шаг в методе Ньютона выглядит как

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - H_k^{-1} \boldsymbol{g}_k.$$

Достаточным условием локального минимума в точке $\boldsymbol{x}_*$ является строгая положительная определённость гессиана $\nabla^2 f(\boldsymbol{x}_*) \succ 0$. Для дважды непрерывно-дифференцируемой функции это означает, что гессиан является положительно определённым и в некоторой окрестности $\boldsymbol{x}_*$. Следовательно, в этой окрестности H_k является невырожденной матрицей, и направление $H_k^{-1} \boldsymbol{g}_k$ определено корректно. Значит, метод Ньютона может быть применен в окрестности $\boldsymbol{x}_*$.

Для квадратичной оптимизируемой функции f метод Ньютона сходится за одну итерацию. В частности, за одну итерацию решается задача на рис. 1,b и рис. 2,b, которая требует значительного числа итераций от методов покоординатного и градиентного спуска. Рассмотрим работу метода Ньютона для функции Розенблока из начального приближения $\boldsymbol{x}_0 = [-1, -1/2]^T$. В этом случае методу достаточно лишь 5-ти итераций для достижения точности по аргументу 10^{-5} (см. рис. 3,c). Для сравнения метод наискорейшего спуска решает эту задачу лишь за 4 тыс. итераций (см. рис. 3,b).

Можно доказать следующую теорему о скорости сходимости метода Ньютона [2]:

Теорема. Предположим, что $f \in C_M^{2,2}$ и гессиан f в оптимальной точке $\boldsymbol{x}_*$ удовлетворяет соотношению $\nabla^2 f(\boldsymbol{x}_*) \succeq lI$ для некоторой константы $l > 0$. Тогда из начальной точки $\boldsymbol{x}_0$:

$$\|\boldsymbol{x}_0 - \boldsymbol{x}_*\| < \bar{r} = \frac{2l}{3M}$$

метод Ньютона сходится с квадратичной скоростью:

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_*\| \leq \frac{M \|\boldsymbol{x}_k - \boldsymbol{x}_*\|^2}{2(l - M\|\boldsymbol{x}_k - \boldsymbol{x}_*\|)}.$$

Заметим, что область квадратичной сходимости метода Ньютона, гарантируемая теоремой, практически совпадает с областью линейной сходимости для метода наискорейшего спуска. Поэтому с практической точки зрения разумно всегда использовать метод Ньютона вместо градиентного спуска на последнем этапе процесса оптимизации (вблизи $\boldsymbol{x}_*$).

Для корректного применения метода Ньютона необходимо выполнение двух условий:

- В текущей точке $\boldsymbol{x}_k$ $H_k \succ 0$. В противном случае квадратичная аппроксимация перестает быть параболоидом с единственным минимумом, и использование точки стационарности $\boldsymbol{x}_k - H_k^{-1}\boldsymbol{g}_k$ теряет физический смысл. Требование $H_k \succ 0$ также важно для того, чтобы направление оптимизации $\boldsymbol{d}_k = -H_k\boldsymbol{g}_k$ было направлением уменьшения значения функции. Действительно, в этом случае $f'_\alpha(\boldsymbol{x}_k - \alpha H_k^{-1}\boldsymbol{g}_k)|_{\alpha=0} = -\boldsymbol{g}_k H_k^{-1}\boldsymbol{g}_k < 0$.
- Квадратичная аппроксимация должна быть адекватным приближением оптимизируемой функции f.

Во многих случаях выполнение этих двух условий не гарантируется. В результате метод Ньютона начинает работать неустойчиво и, в ряде случаев, расходится. Рассмотрим для примера функцию вида корень из функции Розенблока. Тогда метод Ньютона расходится практически из любого начального приближения.

Комбинирование градиентного спуска и метода Ньютона

На текущий момент у нас имеется метод градиентного спуска, который при выборе малой длины шага надежно сходится из любого начального приближения, но порой требует очень большого числа итераций, а также метод Ньютона, который может сходиться быстро, а может, наоборот, работать неустойчиво и в некоторых ситуациях может расходиться. По аналогии с методами одномерной оптимизации, в которых оптимальной стратегией является комбинирование надежных методов золотого сечения и деления отрезка пополам с быстрыми методами на базе квадратичных и кубических аппроксимаций, в данном случае также разумной является идея комбинирования градиентного спуска и метода Ньютона.

Выше было отмечено два основных случая, которые являются проблемными для метода Ньютона. Решение проблемы возможной неадекватности текущей квадратичной аппроксимации заключается в использовании дополнительной одномерной оптимизации вдоль направления $\boldsymbol{d}_k = -H_k^{-1}\boldsymbol{g}_k$:

$$\alpha_k = \arg\min_{\alpha \geq 0} f(\boldsymbol{x}_k - \alpha H_k^{-1}\boldsymbol{g}_k), \quad \boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k H_k^{-1}\boldsymbol{g}_k.$$

Как уже было отмечено выше, в случае $H_k \succ 0$ обратная матрица всегда существует. Кроме того, в этом случае направление $-H_k^{-1}\boldsymbol{g}_k$ является направлением с отрицательной производной. В результате одномерная оптимизация вдоль данного направления обязана приводить к неувеличению значения функции f на очередной итерации.

Рассмотрим несколько подходов к решению проблемы возможного отсутствия положительной определённости H_k:

- Собственное разложение. Матрица H_k является симметричной. Поэтому она может быть представлена как $H_k = Q_k\Lambda_k Q_k^T$, где Q_k – ортогональная матрица, состоящая из собственных векторов H_k, а $\Lambda_k = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$, где λ_i – собственное значение H_k. Произведем коррекцию отрицательных собственных значений и рассмотрим матрицу $\hat{\Lambda} = \mathrm{diag}(\hat{\lambda}_1, \ldots, \hat{\lambda}_N)$, где

  $$\hat{\lambda}_i = \begin{cases} \lambda_i, & \text{если } \lambda_i > \varepsilon, \\ \varepsilon, & \text{иначе.} \end{cases}$$

  Тогда текущее направление оптимизации может быть вычислено следующим образом:

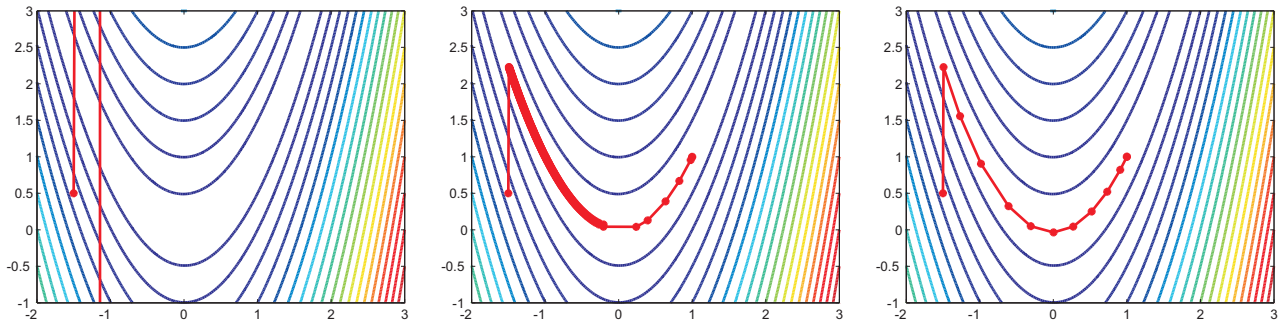
  $$Q_k\hat{\Lambda}_k Q_k^T\boldsymbol{d}_k = -\boldsymbol{g}_k \;\Rightarrow\; \boldsymbol{d}_k = -Q_k\hat{\Lambda}_k^{-1}Q_k^T\boldsymbol{g}_k.$$

  Заметим, что обращение диагональной матрицы $\hat{\Lambda}_k$ может быть выполнено аналитически.

- LDL разложение. Для больших размеров H_k поиск собственного разложения требует значительного времени. Альтернативным подходом здесь является использование LDL-разложения, т.е. представление матрицы H_k как $L_k D_k L_k^T$, где L_k – нижнетреугольная матрица, а D_k – диагональная матрица. Известно, что для положительно-определенной матрицы все элементы на диагонали D_k являются положительными[1]. Поэтому, по аналогии с предыдущим случаем, мы можем заменить отрицательные и малые положительные элементы D_k на некоторое ε. Кроме того, использование LDL-разложения позволяет находить $\boldsymbol{d}_k$ более эффективно:

  $$L_k\hat{D}_k L_k^T\boldsymbol{d}_k = -\boldsymbol{g}_k \;\Rightarrow\; \boldsymbol{d}_k = -L_k^{-T}\hat{D}_k^{-1}L_k^{-1}\boldsymbol{g}_k.$$

[1] они не соответствуют собственным значениям матрицы



(a) (b) (c)

Рис. 4: Примеры работы метода Ньютона для функции вида корень из функции Розенблока. Случай (a): расходящийся процесс для стандартного метода Ньютона, случай (b): использование одномерной оптимизации стабилизирует процесс, но требует большого числа итераций (всего 441), случай (c): использование одномерной оптимизации и коррекции собственного разложения гессиана позволяет получить стабильный итерационный процесс, сходящийся за 14 итераций.

Здесь обращение диагональной матрицы $\hat{D}_k$ может быть выполнено аналитически, а решение системы линейных уравнений с треугольной матрицей может быть эффективно получено с помощью процедуры обратного исключения неизвестных.

- Ограничение на длину шага (damped Newton). Одной из характеристик ситуации нестабильной работы метода Ньютона является большая величина шага $\boldsymbol{d}_k$. Рассмотрим задачу построения квадратичной аппроксимации функции f с дополнительным ограничением на длину $\boldsymbol{d}_k$:

$$f(\boldsymbol{x}_k) + \boldsymbol{g}_k^T \boldsymbol{d}_k + \frac{1}{2}\boldsymbol{d}_k^T H_k \boldsymbol{d}_k \to \min_{\boldsymbol{d}_k},$$
$$\|\boldsymbol{d}_k\|^2 \leq \eta.$$

Можно показать, что решение данной задачи условной оптимизации эквивалентно решению следующей задачи безусловной оптимизации:

$$L(\boldsymbol{d}_k) = f(\boldsymbol{x}_k) + \boldsymbol{g}_k^T \boldsymbol{d}_k + \frac{1}{2}\boldsymbol{d}_k^T H_k \boldsymbol{d}_k + \frac{\lambda}{2}\|\boldsymbol{d}_k\|^2 \to \min_{\boldsymbol{d}_k},$$

где между λ и η есть взаимно-однозначное соответствие[2]. Приравнивая к нулю градиент функции L, получаем:

$$\nabla L(\boldsymbol{d}_k) = \boldsymbol{g}_k + H_k \boldsymbol{d}_k + \lambda \boldsymbol{d}_k = 0 \;\Rightarrow\; \boldsymbol{d}_k = -(H_k + \lambda I)^{-1}\boldsymbol{g}_k.$$

Заметим, что матрица $H_k + \lambda I$ будет гарантированно положительно определена при $\lambda > |\lambda_{min}(H_k)|$, где λ_{min} – минимальное собственное значение гессиана. Введенное направление оптимизации $\boldsymbol{d}_k = -(H_k + \lambda I)^{-1}\boldsymbol{g}_k$ является компромиссом между градиентом и направлением Ньютона. Действительно, в случае большого значения λ $(H_k + \lambda I)^{-1} \simeq \lambda^{-1} I$ и $\boldsymbol{d}_k \simeq -\lambda^{-1}\boldsymbol{g}_k$. Если значение λ мало, то $(H_k + \lambda I)^{-1} \simeq H_k^{-1}$ и $\boldsymbol{d}_k \simeq -H_k^{-1}\boldsymbol{g}_k$. Одной из возможных стратегий выбора λ является адаптивная коррекция, описанная выше для случая градиентного спуска.

На рис. 4 показаны примеры работы стандартного метода Ньютона, метода Ньютона с использованием одномерной оптимизации и метода Ньютона с коррекцией собственного разложения.

Метод Левенберга-Марквардта (Levenberg-Marquardt)

Рассмотрим классическую задачу восстановления регрессии. Пусть имеется обучающая выборка из N объектов $(\boldsymbol{t}, X) = \{t_n, \boldsymbol{x}_n\}_{n=1}^N$, где $\boldsymbol{x}_n \in \mathbb{R}^D$ – признаковое описание объекта, а t_n – его регрессионная переменная. Требуется на основе этих данных восстановить функцию регрессии $f : \mathbb{R}^D \to \mathbb{R}$, с помощью которой можно прогнозировать значение регрессионной переменной для нового объекта, представленного своим вектором признаков $\boldsymbol{x}$. Предположим, что функция f принадлежит некоторому параметрическому семейству и определяется набором параметров $\boldsymbol{w}$[3]. Предположим далее, что наблюдаемые данные соответствуют следующей модели:

$$t_n = f(\boldsymbol{x}_n, \boldsymbol{w}) + \varepsilon_n,\; \varepsilon_n \sim \mathcal{N}(0, \sigma^2).$$

[2] функция L является функцией Лагранжа для задачи условной оптимизации с оптимальным значением двойственной переменной λ

[3] простейшим примером функции регрессии является линейная функция $\boldsymbol{w}^T\boldsymbol{x}$

Тогда обучение параметров регрессии $\boldsymbol{w}$ с помощью метода максимального правдоподобия эквивалентно минимизации следующего функционала вида «сумма квадратов»:

$$L(\boldsymbol{w}) = \sum_{n=1}^{N} (f(\boldsymbol{x}_n, \boldsymbol{w}) - t_n)^2 \to \min_{\boldsymbol{w}}.$$

Вычислим градиент и гессиан функции L:

$$\frac{\partial L}{\partial w_i} = 2 \sum_{n=1}^{N} (f(\boldsymbol{x}_n, \boldsymbol{w}) - t_n) \frac{\partial f(\boldsymbol{x}_n, \boldsymbol{w})}{\partial w_i},$$

$$\frac{\partial^2 L}{\partial w_j \partial w_i} = 2 \sum_{n=1}^{N} \frac{\partial f(\boldsymbol{x}_n, \boldsymbol{w})}{\partial w_j} \frac{\partial f(\boldsymbol{x}_n, \boldsymbol{w})}{\partial w_i} + 2 \sum_{n=1}^{N} (f(\boldsymbol{x}_n, \boldsymbol{w}) - t_n) \frac{\partial^2 f(\boldsymbol{x}_n, \boldsymbol{w})}{\partial w_j \partial w_i}.$$

Ключевое предположение методов из семейства Левенберга-Марквардта состоит в том, что второе слагаемое в гессиане L можно отбросить. Это соответствует аппроксимации функций $f(\boldsymbol{x}_n, \boldsymbol{w})$ линейными функциями в окрестности текущей точки $\boldsymbol{w}$. Другой аргумент в пользу отбрасывания слагаемого в гессиане состоит в том, что в окрестности оптимального решения $f(\boldsymbol{x}_n, \boldsymbol{w}) \simeq t_n$. Введем следующие обозначения:

$\boldsymbol{f} = \{f(\boldsymbol{x}_n, \boldsymbol{w})\}_{n=1}^{N}$ – вектор значений функции регрессии на объектах обучения,

$J = \left\{ \dfrac{\partial f(\boldsymbol{x}_n, \boldsymbol{w})}{\partial w_i} \right\}_{n,i=1}^{N,D}$ – якобиан функции f.

Тогда, с учетом отбрасывания слагаемого в гессиане, можно записать:

$$\nabla L = 2J^T(\boldsymbol{f} - \boldsymbol{t}), \quad \nabla^2 L = 2J^T J.$$

В результате шаг оптимизации по методу Ньютона в данном случае выглядит как $\boldsymbol{w}^{k+1} = \boldsymbol{w}^k - (J_k^T J_k)^{-1} J_k^T (\boldsymbol{f}_k - \boldsymbol{t})$, где $\boldsymbol{w}^k$ – значение параметров регрессии на шаге k. Такой метод получил название метода Гаусса-Ньютона. В случае линейной регрессии $f(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{w}^T \boldsymbol{x}$ второе слагаемое в гессиане не возникает, функция L является квадратичной функцией от $\boldsymbol{w}$ и метод Гаусса-Ньютона сходится за одну итерацию[4].

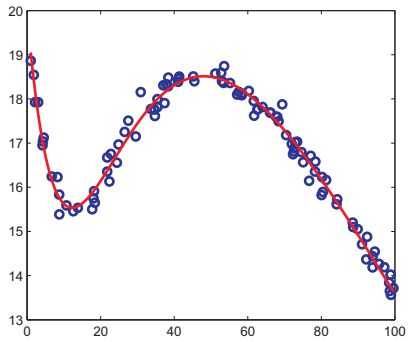


Рис. 5: Пример восстановления нелинейной регрессии вида $w_1 \exp(-x/w_2) + w_3 x \exp(-x/w_4)$ с помощью метода Левенберга-Марквардта. Истинные параметры регрессии $\boldsymbol{w} = [20, 10, 1, 50]^T$.

Матрица $J_k^T J_k$ является неотрицательно-определенной. Поэтому для обеспечения строгой положительной определённости достаточно рассмотреть матрицу вида $J_k^T J_k + \lambda I$ для любого положительного значения λ. Метод с шагом вида $\boldsymbol{w}^{k+1} = \boldsymbol{w}^k - (J_k^T J_k + \lambda I)^{-1} J_k^T (\boldsymbol{f}_k - \boldsymbol{t})$ получил название метода Левенберга.

Как было отмечено выше, скорость сходимости метода наискорейшего спуска существенно зависит от выбранной шкалы измерения компонентов оптимизируемого вектора. В частности, метод Ньютона можно интерпретировать как метод наискорейшего спуска, в котором используется невырожденное преобразование координат B^{-1}, где матрица B есть корень из гессиана. На основе этого соображения Марквардт предложил следующую модификацию метода Левенберга:

$$\boldsymbol{w}^{k+1} = \boldsymbol{w}^k - (J_k^T J_k + \lambda \operatorname{diag}(J_k^T J_k))^{-1} J_k^T (\boldsymbol{f}_k - \boldsymbol{t}).$$

Значение λ на каждой итерации может выбираться с помощью стратегии адаптивной коррекции, описанной выше для градиентного спуска.

[4] в этом случае метод Гаусса-Ньютона тождественно совпадает с методом Ньютона

Таким образом, описанные методы относятся к методам первого порядка, т.к. за счет отбрасывания второго слагаемого в гессиане удалось избавиться от необходимости вычисления второй производной функции регрессии f. При этом данные методы обладают практически квадратичной скоростью сходимости в окрестности оптимального решения.

Рассмотрим в качестве примера задачу восстановления одномерной регрессии вида $f(x, \boldsymbol{w}) = w_1 \exp(-x/w_2) + w_3 x \exp(-x/w_4)$ по выборке из 100 объектов, показанной на рис. 5. Применение метода Гаусса-Ньютона для этой задачи приводит к сходимости за 251 итерацию. Использование подхода Левенберга с адаптивной коррекцией λ уменьшает количество итераций до 25-ти. Метод Марквардта в данном случае требует всего 10 итераций. Для сравнения другой метод первого порядка – метод наискорейшего спуска – требует для решения этой задачи 350 итераций.

Список литературы

[1] Yu. Nesterov. Efficiency of coordinate descent methods for huge-scale optimization problems // CORE discussion paper, 2010/2, 2010.

[2] Yu. Nesterov. Introductory lectures on convex optimization. Kluwer, Boston, 2004.