Севастопольский государственный университет
Институт информационных технологий

Дополнительная профессиональная программа профессиональной переподготовки «Глубокие нейросети в компьютерном зрении»

Основы нейронных сетей

Лекция 3
ЦЕЛЕВЫЕ ФУНКЦИИ И АЛГОРИТМЫ ОПТИМИЗАЦИИ

Бондарев Владимир Николаевич

Целевая функция и условия оптимумов

1. Стационарные точки и условия оптимумов
2. Квадратичная целевая функция и интерпретация собственных значений матрицы Гессе

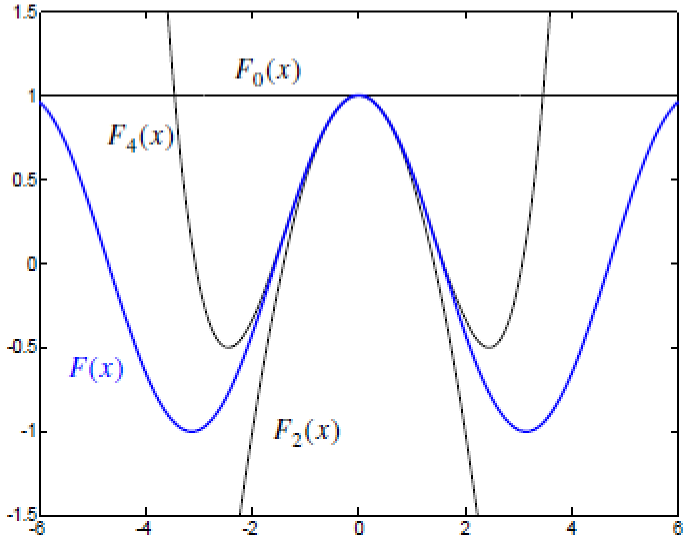
Напоминание: ряд Тейлора

Будем полагать, что $F(\mathbf{x})$ является **аналитической функцией**, т.е существуют все её производные . Тогда её можно аппроксимировать рядом Тейлора в окрестности некоторой точки x^*.

1. Одномерный случай

$$F(x) = F(x^*) + \frac{d}{dx}F(x)\Big|_{x = x^*}(x - x^*) + \frac{1}{2}\frac{d^2}{dx^2}F(x)\Big|_{x = x^*}(x - x^*)^2 + \cdots + \frac{1}{n!}\frac{d^n}{dx^n}F(x)\Big|_{x = x^*}(x - x^*)^n + \cdots$$

Пример: $F(x)=\cos(x)$,



Разложение в ряд по 4-м членам при $x^*=0$:

$$F(x) \approx F_4(x) = 1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 .$$

Напоминание: ряд Тейлора

2. Многомерный случай, **x** – вектор (n x1)

$$F(\mathbf{x}) = F(\mathbf{x}^*) + \nabla F(\mathbf{x})^T\Big|_{\mathbf{x}=\mathbf{x}^*}(\mathbf{x}-\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T \nabla^2 F(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}^*}(\mathbf{x}-\mathbf{x}^*) + \cdots$$

$\nabla F(\mathbf{x})$ - вектор градиента

$$\nabla F(\mathbf{x}) = \left[\frac{\partial}{\partial x_1}F(\mathbf{x}) \; \frac{\partial}{\partial x_2}F(\mathbf{x}) \; \ldots \; \frac{\partial}{\partial x_n}F(\mathbf{x})\right]^T$$

$\nabla^2 F(\mathbf{x})$ - матрица Гессе (матрица вторых производных, гессиан)

$$\nabla^2 F(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2}{\partial x_1^2}F(\mathbf{x}) & \frac{\partial^2}{\partial x_1 \partial x_2}F(\mathbf{x}) & \ldots & \frac{\partial^2}{\partial x_1 \partial x_n}F(\mathbf{x}) \\ \frac{\partial^2}{\partial x_2 \partial x_1}F(\mathbf{x}) & \frac{\partial^2}{\partial x_2^2}F(\mathbf{x}) & \ldots & \frac{\partial^2}{\partial x_2 \partial x_n}F(\mathbf{x}) \\ \vdots & \vdots & & \vdots \\ \frac{\partial^2}{\partial x_n \partial x_1}F(\mathbf{x}) & \frac{\partial^2}{\partial x_n \partial x_2}F(\mathbf{x}) & \ldots & \frac{\partial^2}{\partial x_n^2}F(\mathbf{x}) \end{bmatrix}.$$

Напоминание: производные по направлению

i-ый элемент вектора градиента $\nabla F(\mathbf{x})$ соответствует **производной по направлению вдоль оси x_i.**
i-ый элемент матрицы Гессе $\nabla^2 F(\mathbf{x})$ на главной диагонали соответствует второй производной по направлению вдоль оси $x_{i.}$

Производные вдоль произвольного направления, заданного вектором **p**:
1) первая производная $\dfrac{\mathbf{p}^T \nabla F(\mathbf{x})}{\|\mathbf{p}\|}$ **;** **2) вторая производная** $\dfrac{\mathbf{p}^T \nabla^2 F(\mathbf{x}) \mathbf{p}}{\|\mathbf{p}\|^2}$.

1-ая производная - *это нормированная проекция градиента на направление* **p.**
Пример: $F(\mathbf{x}) = x_1^2 + 2x_2^2$.
Производная по направлению $\mathbf{p} = \begin{bmatrix} 2 & -1 \end{bmatrix}^T$ в точке $\mathbf{x}^* = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}^T$

$$\nabla F(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}^*} = \begin{bmatrix} \dfrac{\partial}{\partial x_1} F(\mathbf{x}) \\ \dfrac{\partial}{\partial x_2} F(\mathbf{x}) \end{bmatrix}\Bigg|_{\mathbf{x}=\mathbf{x}^*} = \begin{bmatrix} 2x_1 \\ 4x_2 \end{bmatrix}\Bigg|_{\mathbf{x}=\mathbf{x}^*} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}. \qquad \frac{\mathbf{p}^T \nabla F(\mathbf{x})}{\|\mathbf{p}\|} = \frac{\begin{bmatrix} 2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix}}{\left\| \begin{bmatrix} 2 \\ -1 \end{bmatrix} \right\|} = \frac{[0]}{\sqrt{5}} = 0.$$

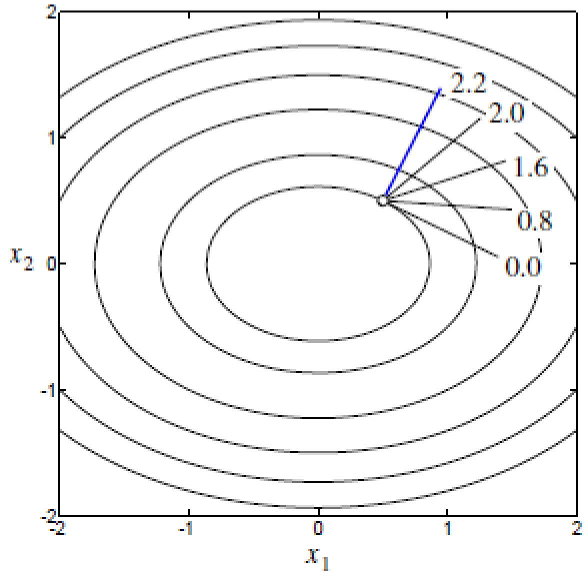
Любое направление ортогональное градиенту имеет нулевую направленную производную, т.е нулевую скорость убывания функции $F(\mathbf{x})$.

В.Бондарев

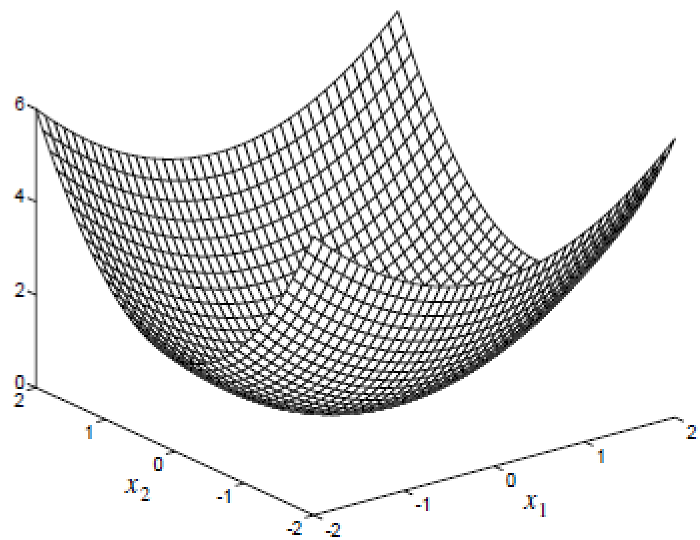
Квадратичная функция и её направленные производные

Какое направление характеризуется максимальной направленной производной (уклоном, скоростью изменения)? Максимум будет, когда скалярное произведение градиента на направление будет максимальным, т.е. когда направление $\mathbf{p} = \nabla F(\mathbf{x})$.

Контуры $F(\mathbf{x})$ (сечения)



3D



На контурном графике 5 направлений и значения направленных производных. Нулевая производная по направлению, ортогональна градиенту (касательная к контурной линии)

В.Бондарев

Минимумы целевой функции

x* является **строгим минимумом** $F(\mathbf{x})$, если для всех $\Delta\mathbf{x}$ таких, что $\delta > \|\Delta\mathbf{x}\| > 0$

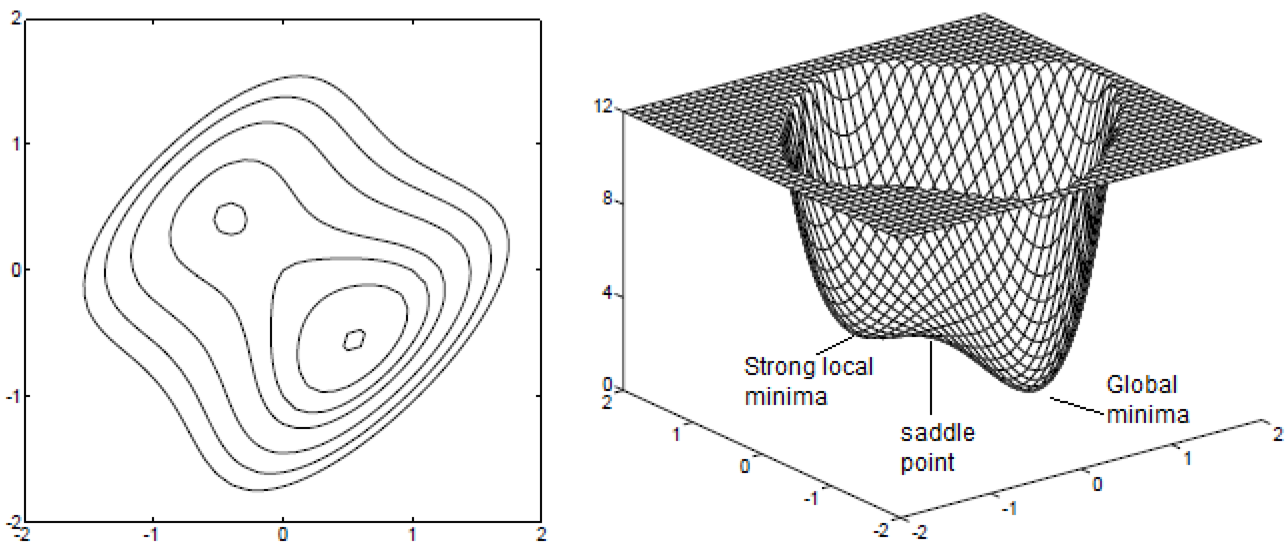
$$F(\mathbf{x}^*) < F(\mathbf{x}^* + \Delta\mathbf{x})$$

x* является **слабым минимумом** $F(\mathbf{x})$, если для всех $\Delta\mathbf{x}$ таких, что $\delta > \|\Delta\mathbf{x}\| > 0$

$$F(\mathbf{x}^*) \le F(\mathbf{x}^* + \Delta\mathbf{x})$$

x* является **глобальным минимумом** $F(\mathbf{x})$, если всех $\Delta\mathbf{x} \neq \mathbf{0}$

$$F(\mathbf{x}^*) < F(\mathbf{x}^* + \Delta\mathbf{x})$$



Седловая точка: вдоль линии $x_1=-x_2$– максимум, вдоль ортогонального направления – минимум. Координаты седловой т. (-0.13, +0.13)

Условия оптимальности

Рассмотрим ряд Тейлора. Если $\|\Delta\mathbf{x}\|$ мало, то можно пренебречь членами высоких порядков, тогда

$$F(\mathbf{x}^* + \Delta\mathbf{x}) \cong F(\mathbf{x}^*) + \nabla F(\mathbf{x})^T\Big|_{\mathbf{x}=\mathbf{x}^*}\Delta\mathbf{x}$$

точка **x*** будет минимумом, если для любых $\Delta\mathbf{x}$

$$\nabla F(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}^*} = \mathbf{0}$$ - **необходимое** *условие 1-го порядка*

В точке минимума x* градиент равен нулю. Точки, удовлетворяющие этому условию называются **стационарными.**

Если **x*** стационарная точка, то с учетом условия 1-го порядка

$$F(\mathbf{x}^* + \Delta\mathbf{x}) = F(\mathbf{x}^*) + \frac{1}{2}\Delta\mathbf{x}^T\nabla^2 F(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}^*}\Delta\mathbf{x} + \cdots .$$

В точке **x*** будет существовать строгий минимум, если

$$\Delta\mathbf{x}^T\nabla^2 F(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}^*}\Delta\mathbf{x} > 0.$$ - *условие2-го порядка*

Чтобы это условие выполнялось для любых $\Delta\mathbf{x} \neq \mathbf{0}$ **достаточно**, чтобы матрица Гессе была **положительно определена ($\mathbf{z}^T\mathbf{A}\mathbf{z} > 0$)** . Для того чтобы в т. **x*** находился минимум (строгий или слабый) достаточно, чтобы матрица Гессе была **положительно полуопределена ($\mathbf{z}^T\mathbf{A}\mathbf{z} \geq 0$).** Если все собственные числа матрицы положительны, то матрица п.о. Если все собственные числа не отрицательны, то матрица п.п.о.

В.Бондарев

Собственные векторы и собственные значения матриц

Рассмотрим линейное преобразование

$$\mathbf{y}=\mathbf{A}\mathbf{z}, \qquad (1)$$

где **z**- вектор, **y**-вектор, **A** – квадратная матрица. Это преобразование отображает произвольный вектор **z** пространства Z в вектор **y** того же пространства.

Вектор $\mathbf{z}\neq 0$, удовлетворяющий соотношению

$$\mathbf{A}\mathbf{z}=\lambda\mathbf{z}\,, \qquad (2)$$

называют **собственным вектором** матрицы **A**, число λ – **собственным значением матрицы A**.

Для определения собственных чисел (2) переписывают в виде

$$(\mathbf{A} - \lambda\, \mathbf{I})\mathbf{z}=0 \qquad (3)$$

системы линейных уравнений и находят решения относительно λ и **z.**

Сначала приравнивают нулю определитель

$$|(\mathbf{A} - \lambda\, \mathbf{I})| =0$$

и находят λ. Затем при заданных λ определяют собственные векторы.

Собственные векторы и собственные значения матриц

Пример:

$$A=\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix}; \quad |(A - \lambda\, I)|=0\,; \quad \left|\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix} - \lambda\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\right| = \left|\begin{bmatrix}2-\lambda & 1\\ 1 & 2-\lambda\end{bmatrix}\right| =0$$

$$(2- \lambda)^2 -1 =0; \quad \lambda^2\text{-}4\,\lambda+3=0; \qquad \lambda_1=1,\, \lambda_2=3$$

$$(A - \lambda_1\, I)\mathbf{z}=\begin{bmatrix}2-\lambda_1 & 1\\ 1 & 2-\lambda_1\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix}=\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix}= z_1+z_2=0;\, z_1=-z_2;\, \mathbf{z_1}=\begin{bmatrix}1\\ -1\end{bmatrix}$$

$$(A - \lambda_2\, I)\mathbf{z}=\begin{bmatrix}2-\lambda_2 & 1\\ 1 & 2-\lambda_2\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix}=\begin{bmatrix}-1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix}= -z_1+z_2=0;\, z_1=z_2;\, \mathbf{z_2}=\begin{bmatrix}1\\ 1\end{bmatrix}$$

.

$$\lambda_1=1,\, \lambda_2=3,\, \mathbf{z_1}=\begin{bmatrix}1\\ -1\end{bmatrix}, \mathbf{z_2}=\begin{bmatrix}1\\ 1\end{bmatrix}$$

Квадратичная целевая функция

Общая форма **квадратичной целевой функции**

$$F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{d}^T\mathbf{x} + c,$$

где **A** - симметричная матрица.

Градиент и гессиан квадратичной функции $F(\mathbf{x})$ соответственно равны:

$$\nabla F(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{d}, \qquad \nabla^2 F(\mathbf{x}) = \mathbf{A}.$$

Все остальные производные квадратичной функции равны нулю.

Рассмотрим квадратичную функцию со стационарной точкой в начале координат $F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x}$. Мы хотим использовать собственные векторы матрицы **A** в качестве новых базисных векторов. Так как **A** симметрична, то её собственные векторы взаимно ортогональны.

Введем матрицу $\mathbf{B} = \begin{bmatrix}\mathbf{z}_1 & \mathbf{z}_2 & \dots & \mathbf{z}_n\end{bmatrix}$, составленную из собственных векторов **A** .

Тогда новая матрица с базисом по собственным векторам будет равна

$$\mathbf{A}' = [\mathbf{B}^T\mathbf{A}\mathbf{B}] = \begin{bmatrix}\lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & \lambda_n\end{bmatrix} = \Lambda$$

Здесь λ_i - собственные числа матрицы **A.** Также верно, что $\mathbf{B}^{-1} = \mathbf{B}^T$ и $\mathbf{A} = \mathbf{B}\Lambda\mathbf{B}^T$.

Интерпретация собственных значений матрицы Гессе

Вторая производная по направлению для квадратичной $F(\mathbf{x})$

$$\frac{\mathbf{p}^T \nabla^2 F(\mathbf{x}) \mathbf{p}}{\|\mathbf{p}\|^2} = \frac{\mathbf{p}^T \mathbf{A} \mathbf{p}}{\|\mathbf{p}\|^2}.$$

Рассмотрим направление $\mathbf{p} = \mathbf{B}\mathbf{c}$, где вектор $\mathbf{c}$ представляет направление $\mathbf{p}$ в координатной системе собственных векторов. Тогда

$$\frac{\mathbf{p}^T \mathbf{A} \mathbf{p}}{\|\mathbf{p}\|^2} = \frac{\mathbf{c}^T \mathbf{B}^T (\mathbf{B} \Lambda \mathbf{B}^T) \mathbf{B} \mathbf{c}}{\mathbf{c}^T \mathbf{B}^T \mathbf{B} \mathbf{c}} = \frac{\mathbf{c}^T \Lambda \mathbf{c}}{\mathbf{c}^T \mathbf{c}} = \frac{\sum_{i=1}^{n} \lambda_i c_i^2}{\sum_{i=1}^{n} c_i^2}.$$

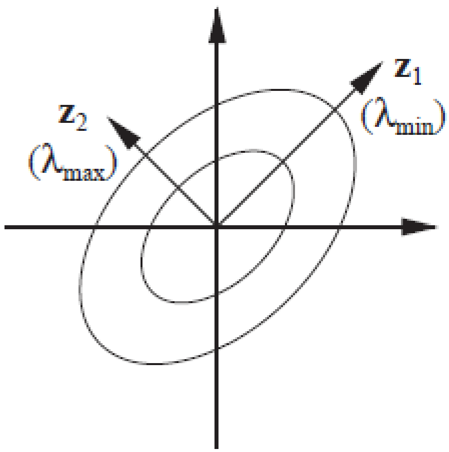
Выводы: 1) вторая производная по направлению – это взвешенное среднее собственных значений и она не может быть больше , наибольшего λ_{max} собственного значения, и меньше наименьшего собственного значения λ_{min} ;
2) она будет иметь наибольшее значение , равное λ_{max} , по направлению собственного вектора $\mathbf{p} = \mathbf{z}_{max}$ с наибольшим собственным значением;
3) в каждом собственном направлении вторые производные будут равны собственным значениям.

Собственные значения – это вторые производные квадратичной $F(\mathbf{x})$ в направлении собственных векторов.

В.Бондарев

Координатная система собственных векторов матрицы Гессе

Собственные векторы матрицы Гессе являются главными осями, функций, представляющих контуры равных уровней $F(\mathbf{x})$. Для двумерного случая



Здесь первое собственное значение меньше второго. Соответственно поверхность $F(\mathbf{x})$ в первом направлении будет менее крутая, а во втором – более крутая. Поэтому во втором направлении мы будем пересекать линии равных контуров быстрее.

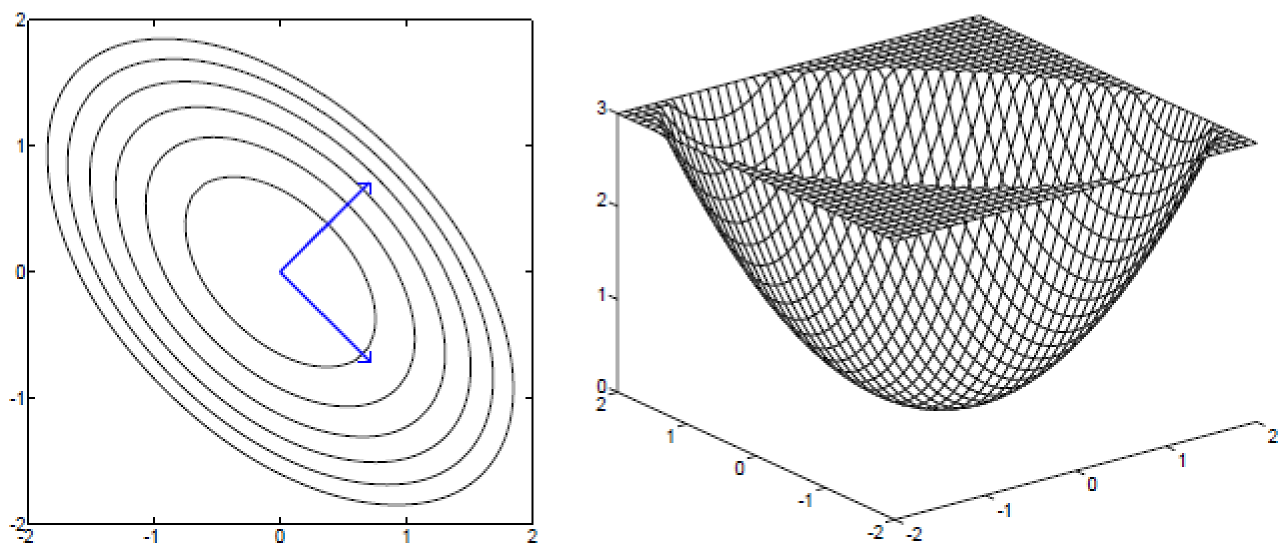
Пример

Рассмотрим следующую квадратичную функцию:

$$F(\mathbf{x}) = x_1^2 + x_1 x_2 + x_2^2 = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \mathbf{x}$$

Гессиан, собственные значения и собственные векторы:

$$\nabla^2 F(\mathbf{x}) = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix},\ \lambda_1 = 1,\ \mathbf{z}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix},\ \lambda_2 = 3,\ \mathbf{z}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$



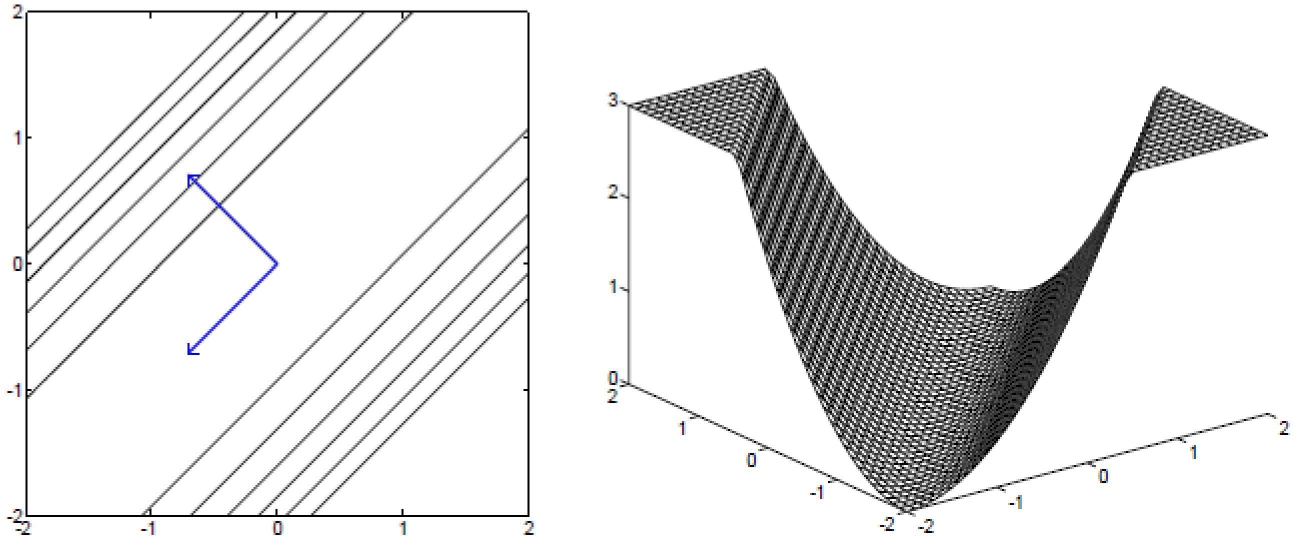
Пример

Рассмотрим следующую квадратичную функцию:

$$F(\mathbf{x}) = \frac{1}{2}x_1^2 - x_1x_2 + \frac{1}{2}x_2^2 = \frac{1}{2}\mathbf{x}^T\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\mathbf{x}\,.$$

Гессиан, собственные значения и собственные векторы:

$$\nabla^2 F(\mathbf{x}) = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix},\ \lambda_1 = 2\,,\ \mathbf{z}_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix},\ \lambda_2 = 0\,,\ \mathbf{z}_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}.$$



АЛГОРИТМЫ ОПТИМИЗАЦИИ ЦЕЛЕВОЙ ФУНКЦИИ

План

1. Итеративные алгоритмы оптимизации
2. Алгоритм наискорейшего спуска (SDA)
   - SDA с фиксированной скоростью обучения
   - SDA с минимизацией вдоль направления
3. Метод Ньютона
4. Метод сопряженных градиентов

Итеративные алгоритмы оптимизации

Цель оптимизации заключается в поиске вектора $\mathbf{x}$, который минимизирует целевую функцию $F(\mathbf{x})$.

Будем использовать для этого алгоритмы последовательного приближения – **итеративные алгоритмы**. Поиск начинается с начального значения $\mathbf{x}_0$ и на каждом шаге

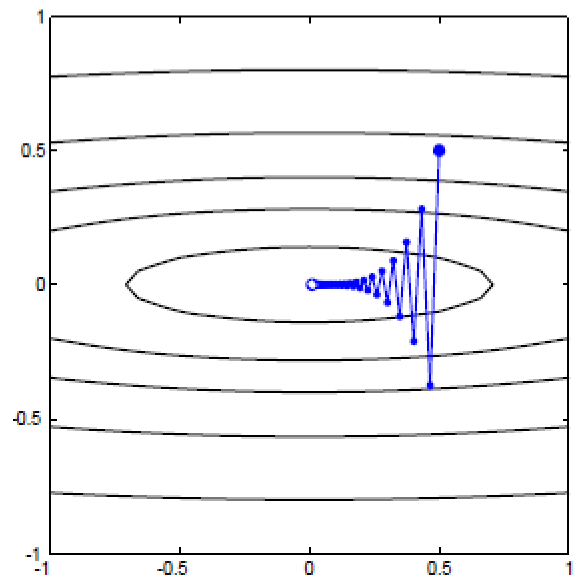
$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k ,$$

или

$$\Delta \mathbf{x}_k = (\mathbf{x}_{k+1} - \mathbf{x}_k) = \alpha_k \mathbf{p}_k ,$$

где $\mathbf{p}_k$ – вектор, определяющий направление поиска; α_k - скорость обучения, определяющая длину шага.

Алгоритмы отличаются выбором вектора направления поиска $\mathbf{p}_k$ и способами вычисления значений скорости обучения



В.Бондарев

Алгоритм наискорейшего спуска (steepest descent algorithm – SDA)

При поиске минимума должно выполняться условие : $F(\mathbf{x}_{k+1}) < F(\mathbf{x}_k)$. (1)

Рассмотрим ряд Тейлора для ЦФ в районе т. $\mathbf{x}_{k+1}$

$$F(\mathbf{x}_{k+1}) = F(\mathbf{x}_k + \Delta\mathbf{x}_k) \approx F(\mathbf{x}_k) + \mathbf{g}_k^T \Delta\mathbf{x}_k , \quad \mathbf{g}_k \equiv \nabla F(\mathbf{x})\Big|_{\mathbf{x} = \mathbf{x}_k} . \quad (2)$$

где $\mathbf{g}_k$ - градиент .

Чтобы выполнялось условие (1), второй член (2) должен быть отрицательным:

$$\mathbf{g}_k^T \Delta\mathbf{x}_k = \alpha_k \mathbf{g}_k^T \mathbf{p}_k < 0 \qquad \text{или} \qquad \mathbf{g}_k^T \mathbf{p}_k < 0 .$$

Любой вектор $\mathbf{p}_k$, удовлетворяющий этому условию, соответствует **направлению спуска. Направление наискорейшего спуска :**

$$\mathbf{p}_k = - \mathbf{g}_k .$$

Соответственно процедура **SDA** определяется выражением:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k .$$

Есть **два подхода определения** α_k :

1) использовать фиксированное значение, например $\alpha_k = 0.01$ или переменное, но предопределенное, например $\alpha_k = 1/k$.

2) минимизировать на каждом шаге $F(\mathbf{x})$ по отношению к α_k вдоль $\mathbf{p}_k$

Отметим, что направление наискорейшего спуска ортогонально линиям равных контуров целевой функции.

Устойчивость SDA с фиксированной скоростью обучения

Пусть целевая функция является **квадратичной**. Тогда $\nabla F(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{d}$.
Подставив это значение в SDA, получим

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \mathbf{g}_k = \mathbf{x}_k - \alpha(\mathbf{A}\mathbf{x}_k + \mathbf{d}) \quad \text{или} \quad \mathbf{x}_{k+1} = [\mathbf{I} - \alpha\mathbf{A}]\mathbf{x}_k - \alpha\mathbf{d}.$$

Это уравнение линейной динамической системы, которая будет устойчивой, если собственные значения матрицы $[\mathbf{I} - \alpha\mathbf{A}]$ меньше 1.
Пусть $\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ и $\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n\}$ собственные значения и собственные векторы **A**.
Тогда

$$[\mathbf{I} - \alpha\mathbf{A}]\mathbf{z}_i = \mathbf{z}_i - \alpha\mathbf{A}\mathbf{z}_i = \mathbf{z}_i - \alpha\lambda_i\mathbf{z}_i = (1 - \alpha\lambda_i)\mathbf{z}_i.$$

Т.е собственные векторы матрицы $[\mathbf{I} - \alpha\mathbf{A}]$ совпадают с собственными векторами **A,** а собственные значения равны $(1 - \alpha\lambda_i)$. Тогда **условие устойчивости SDA** запишется в виде $|(1 - \alpha\lambda_i)| < 1$.
Если квадратичная $F(\mathbf{x})$ имеет сильный минимум, то все собственные значения должны быть положительными. Тогда условие устойчивости

$$\alpha < \frac{2}{\lambda_i}.$$

Поскольку оно должно выполняться для всех собственных чисел, то, очевидно,

$$\alpha < \frac{2}{\lambda_{max}}.$$

Т.е. скорость обучения обратно пропорциональна кривизне квадратичной функции.

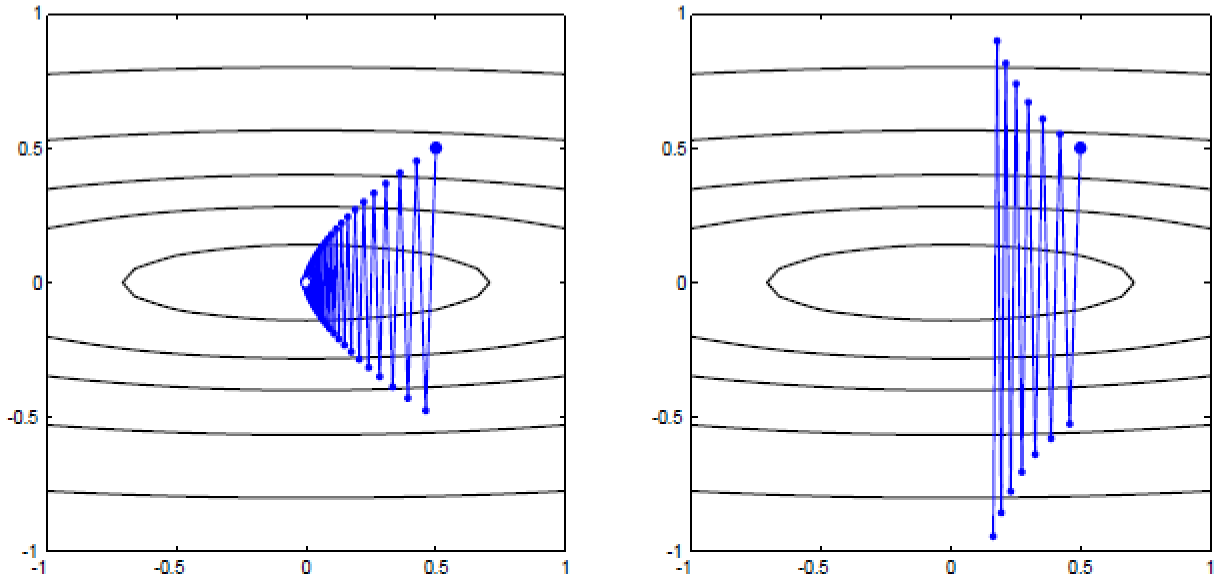
В.Бондарев

Пример SDA

$$F(\mathbf{x}) = x_1^2 + 25x_2^2 \qquad \mathbf{A} = \begin{bmatrix} 2 & 0 \\ 0 & 50 \end{bmatrix} \qquad \left\{(\lambda_1 = 2), \left(\mathbf{z}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right)\right\}, \left\{(\lambda_2 = 50), \left(\mathbf{z}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right)\right\}.$$

$$\alpha < \frac{2}{\lambda_{max}} = \frac{2}{50} = 0.04 . \qquad \mathbf{x}_0 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} . \qquad \nabla F(\mathbf{x}) = \begin{bmatrix} \frac{\partial}{\partial x_1} F(\mathbf{x}) \\ \frac{\partial}{\partial x_2} F(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} 2x_1 \\ 50x_2 \end{bmatrix} . \qquad \mathbf{g}_0 = \nabla F(\mathbf{x})\Big|_{\mathbf{x} = \mathbf{x}_0} = \begin{bmatrix} 1 \\ 25 \end{bmatrix} .$$

$\alpha = 0.039$ $\qquad$ $\alpha = 0.041$



SDA с оптимизацией целевой функции по скорости обучения

В этом случае на каждом шаге алгоритма выполняется поиск α_k которое **минимизирует** $F(\mathbf{x}_k + \alpha_k \mathbf{p}_k)$ **вдоль направления** $\mathbf{p}_k$.

Для произвольных $F(\mathbf{x})$ требуется выполнять линейный поиск, но для квадратичной $F(\mathbf{x})$ решение можно найти аналитически.

Можно показать, используя ряд Тейлора, что производная $F(\mathbf{x})$ по скорости обучения

$$\frac{d}{d\alpha_k} F(\mathbf{x}_k + \alpha_k \mathbf{p}_k) = \nabla F(\mathbf{x})^T \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k + \alpha_k \mathbf{p}_k^T \nabla^2 F(\mathbf{x}) \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k .$$

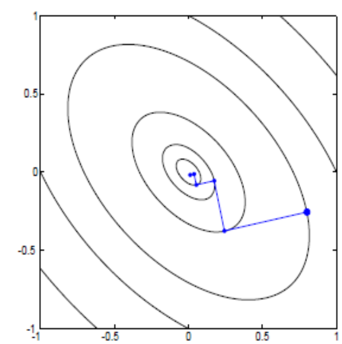
Приравняв эту производную нулю, получим оптимальное α_k

$$\alpha_k = -\frac{\nabla F(\mathbf{x})^T \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k}{\mathbf{p}_k^T \nabla^2 F(\mathbf{x}) \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k} = -\frac{\mathbf{g}_k^T \mathbf{p}_k}{\mathbf{p}_k^T \mathbf{A}_k \mathbf{p}_k} .$$

Для квадратичной функции гессиан $\mathbf{A}$ не зависит от k.

Последовательные шаги алгоритма выполняются вдоль взаимно ортогональных направлений, т.к.

$$\frac{d}{d\alpha_k} F(\mathbf{x}_{k+1}) = \nabla F(\mathbf{x})^T \Big|_{\mathbf{x} = \mathbf{x}_{k+1}} \frac{d}{d\alpha_k} [\mathbf{x}_k + \alpha_k \mathbf{p}_k] = \nabla F(\mathbf{x})^T \Big|_{\mathbf{x} = \mathbf{x}_{k+1}} \mathbf{p}_k = \mathbf{g}_{k+1}^T \mathbf{p}_k = 0$$



Следовательно в т. минимума по α_k градиент ортогонален предыдущему направлению поиска.

Метод Ньютона

Алгоритм SDA основан на использовании первых производных. Метод Ньютона основан на поиске стационарной точки квадратичной аппроксимации ЦФ $F(\mathbf{x})$:

$$F(\mathbf{x}_{k+1}) = F(\mathbf{x}_k + \Delta\mathbf{x}_k) \approx F(\mathbf{x}_k) + \mathbf{g}_k^T \Delta\mathbf{x}_k + \frac{1}{2}\Delta\mathbf{x}_k^T \mathbf{A}_k \Delta\mathbf{x}_k . \qquad (1)$$

Найдем градиент (1) по отношению к $\Delta\mathbf{x}_k$ и приравняем его нулю

$$\mathbf{g}_k + \mathbf{A}_k \Delta\mathbf{x}_k = \mathbf{0} .$$

Отсюда оптимальный шаг равен $\Delta\mathbf{x}_k = -\mathbf{A}_k^{-1}\mathbf{g}_k$.

Метод Ньютона

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{A}_k^{-1}\mathbf{g}_k .$$

Пример:

$$F(\mathbf{x}) = x_1^2 + 25x_2^2 \qquad \mathbf{A} = \begin{bmatrix} 2 & 0 \\ 0 & 50 \end{bmatrix} \qquad \mathbf{x}_0 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} .$$

На первом шаге метода Ньютона получаем:

$$\mathbf{x}_1 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} - \begin{bmatrix} 2 & 0 \\ 0 & 50 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 25 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} - \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} .$$

Метод всегда находит минимум квадратичной функции за один шаг. В общем случае метод Ньютона не сходится за один шаг и не гарантирует схождение. Это зависит от вида $F(\mathbf{x})$ и начальных условий поиска.

Метод сопряженных градиентов

Метод обладает **квадратичным окончанием**, если он минимизирует квадратичную функцию за конечное число шагов. Метод Ньютона требует всего лишь одного шага. Но он требует вычисления 2-х производных (n^2) . Желательно иметь методы, которые используют только первую производную, но обладают свойством квадратичного окончания.

Пусть $F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{d}^T\mathbf{x} + c$.

Векторы являются **взаимно сопряженными** для п.о. **A**, если $\mathbf{p}_k^T\mathbf{A}\mathbf{p}_j = 0 \qquad k \neq j$.

Например, сопряженными будут собственные векторы **A**:

$$\mathbf{z}_k^T\mathbf{A}\mathbf{z}_j = \lambda_j\mathbf{z}_k^T\mathbf{z}_j = 0 \qquad k \neq j ,$$

Доказано: ***Если для поиска используются сопряженные направления, то любая квадратичная функция n переменных, имеющая минимум, может быть минимизирована за n шагов.***

Как построить эти направления ? Для этого нужно переопределить условия сопряженности без использования матрицы Гессе.

Метод сопряженных градиентов

Для квадратичной целевой функции: $\nabla F(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{d}$, и $\nabla^2 F(\mathbf{x}) = \mathbf{A}$.
Тогда изменения градиента на k+1 итерации

$$\Delta\mathbf{g}_k = \mathbf{g}_{k+1} - \mathbf{g}_k = (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{d}) - (\mathbf{A}\mathbf{x}_k + \mathbf{d}) = \mathbf{A}\Delta\mathbf{x}_k \quad \text{и} \quad \Delta\mathbf{x}_k = \alpha_k \mathbf{p}_k ,$$

где скорость обучения оптимизируется вдоль направления $\mathbf{p}_k$.
Выполняя подстановки, получим **новое условие сопряженности направлений**

$$\alpha_k \mathbf{p}_k^T \mathbf{A}\mathbf{p}_j = \Delta\mathbf{x}_k^T \mathbf{A}\mathbf{p}_j = \Delta\mathbf{g}_k^T \mathbf{p}_j = 0 \qquad k \neq j \cdot$$

Т.о. , направления поиска будут сопряженными, если они ортогональны направлениям изменения градиента.
Обычно поиск начинают с направления наискорейшего спуска $\mathbf{p}_0 = -\mathbf{g}_0$. Затем на каждой итерации конструируют вектор $\mathbf{p}_k$, ортогональный к $\{\Delta\mathbf{g}_0, \Delta\mathbf{g}_1, \ldots, \Delta\mathbf{g}_{k-1}\}$. Эта процедура может быть представлена формулой

$$\mathbf{p}_k = -\mathbf{g}_k + \beta_k \mathbf{p}_{k-1} \cdot$$

Скалярный вес β_k выбирается так, чтобы $\mathbf{p}_k$ и $\mathbf{p}_{k-1}$ были сопряженными.
Т.е. новое направление поиска является линейной комбинацией текущего направления наискорейшего спуска и предыдущего направления поиска.

Метод сопряженных градиентов

Для вычисления весовых коэффициентов β_k используется только текущий градиент и предпоследний градиент:

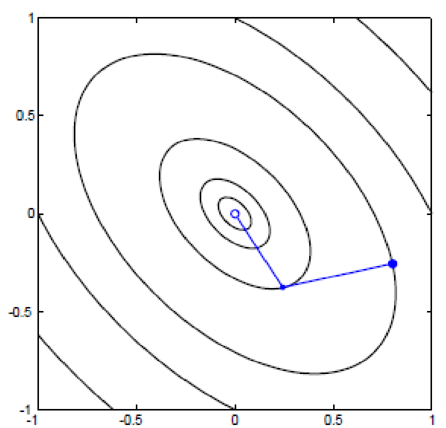
$$\beta_k = \frac{\Delta \mathbf{g}_{k-1}^T \mathbf{g}_k}{\Delta \mathbf{g}_{k-1}^T \mathbf{p}_{k-1}} \text{ или } \beta_k = \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}} \text{ или } \beta_k = \frac{\Delta \mathbf{g}_{k-1}^T \mathbf{g}_k}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}$$

Формулы соответствуют методам Хестенсона-Штифеля, Флетчера-Ривса , Полака-Рибейры.

Алгоритм сопряженных градиентов (conjugate gradient algorithm - CGA)

1. Выбрать в качестве начального направления $\mathbf{p}_0 = -\mathbf{g}_0$;
2. Выполнить шаг в соответствии $\Delta \mathbf{x}_k = \alpha_k \mathbf{p}_k$, выбирая α_k , которое минимизирует $F(\mathbf{x}_k + \alpha_k \mathbf{p}_k)$ вдоль направления $\mathbf{p}_k$;
3. Выбрать следующее направление в соответствии $\mathbf{p}_k = -\mathbf{g}_k + \beta_k \mathbf{p}_{k-1}$.
4. Если не достигнута точность $\|\mathbf{p}_k\| < e$ вернуться к п.2

$$F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \mathbf{x}$$



Как и предсказано алгоритм сходится за 2 шага