метод сопряженных градиентов

Conjugate gradient method

квадрат. 2

$$\|x^{k+1} - x^*\|_2 \le \left(\frac{\sqrt{\varkappa} - 1}{\sqrt{\varkappa} + 1}\right)^k \|x^0 - x^*\|$$

Daniil Merkulov

Optimization methods. MIPT

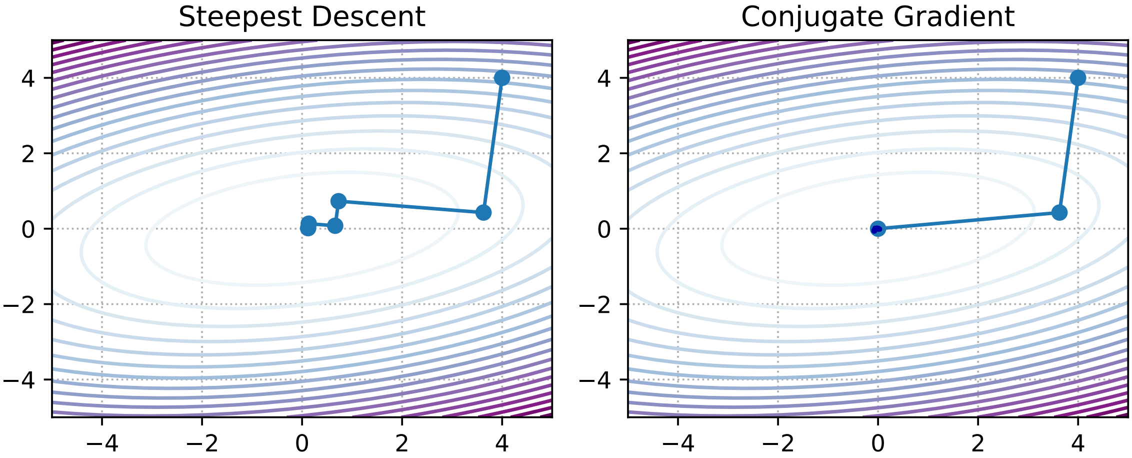
Strongly convex quadratics

Consider the following quadratic optimization problem:

$$\min_{x \in \mathbb{R}^d} f(x) = \min_{x \in \mathbb{R}^d} \frac{1}{2} x^\top A x - b^\top x + c, \text{ where } A \in \mathbb{S}^d_{++}.$$

Optimality conditions

$$Ax^* = b$$



Exact line search aka steepest descent

$$\alpha_k = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_{k+1}) = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

More theoretical than practical approach. It also allows you to analyze the convergence, but often exact line search can be difficult if the function calculation takes too long or costs a lot. An interesting theoretical property of this method is that each following iteration is orthogonal to the previous one:

$$\alpha_k = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Exact line search aka steepest descent

$$\alpha_k = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_{k+1}) = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

$\nabla f(x) = Ax - b$

More theoretical than practical approach. It also allows you to analyze the convergence, but often exact line search can be difficult if the function calculation takes too long or costs a lot. An interesting theoretical property of this method is that each following iteration is orthogonal to the previous one:

$$\alpha_k = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

$\nabla f(x^k)^T \nabla f(x^{k+1}) = 0$

Optimality conditions:

$\nabla f(x^k)^T \cdot (Ax^{k+1} - b) = 0$

$g^k = \nabla f(x^k)$

$\nabla f(x^k)^T (A(x^k - \alpha^k \nabla f(x^k)) - b) = 0$

$g^{k^T} g^k - \alpha^k g^{k^T} A g^k = 0$

Exact line search aka steepest descent

$$\alpha_k = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_{k+1}) = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

More theoretical than practical approach. It also allows you to analyze the convergence, but often exact line search can be difficult if the function calculation takes too long or costs a lot. An interesting theoretical property of this method is that each following iteration is orthogonal to the previous one:

$$\alpha_k = \text{arg}\min_{\alpha \in \mathbb{R}^+} f(x_k - \alpha \nabla f(x_k))$$

Optimality conditions:

$$\nabla f(x_k)^T \nabla f(x_{k+1}) = 0$$

Optimal value for quadratics

$$\nabla f(x_k)^\top A(x_k - \alpha \nabla f(x_k)) - \nabla f(x_k)^\top b = 0 \qquad \alpha_k = \frac{\nabla f(x_k)^T \nabla f(x_k)}{\nabla f(x_k)^T A \nabla f(x_k)}$$

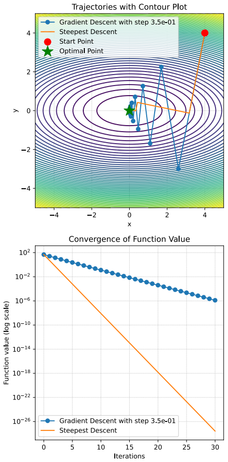


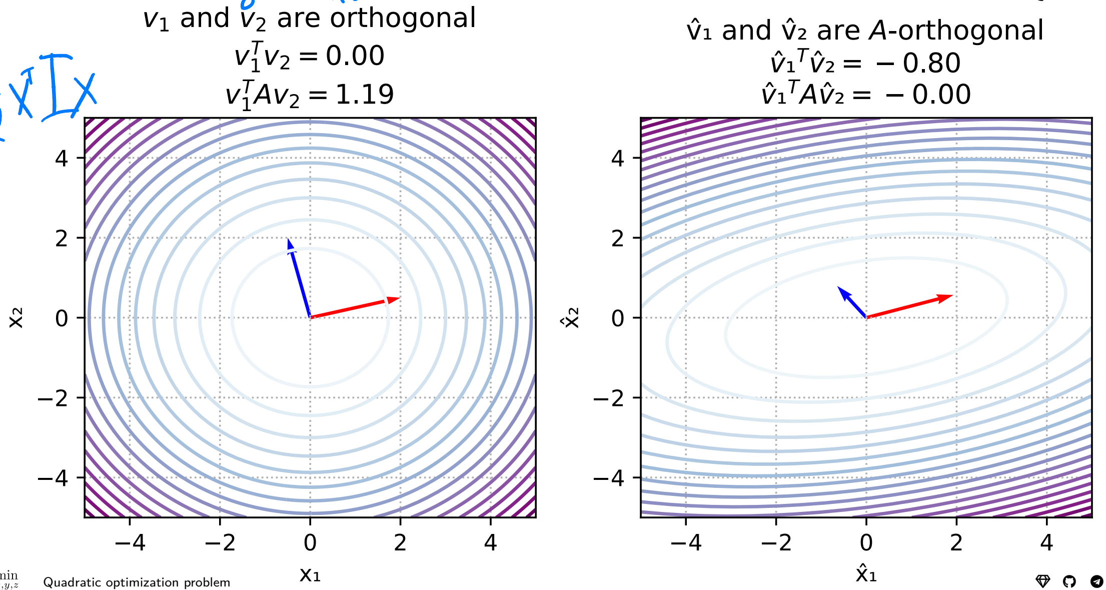
Figure 1: Steepest Descent

Open In Colab ♣

Conjugate directions. A-orthogonality.

Идеальный $\frac{1}{2}x^T I x$

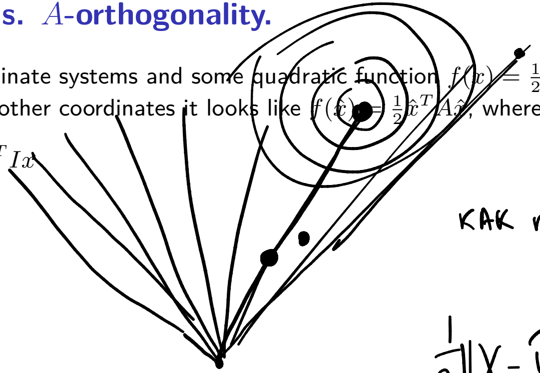
Реальный $\frac{1}{2}x^T A x$



Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A \hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A \hat{x}$$



Since $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A \hat{x}$$

как преобразоваться на эллипс ?

$$\frac{1}{2}\|x - \tilde{y}\|_2^2 \to \min$$

$$x^T A x \leq 1$$

$$L = \frac{1}{2}\|x-y\|_2^2 + \lambda(x^T A x - 1) \Rightarrow$$

$$\frac{\partial L}{\partial x} = x - y + 2\lambda A x = 0$$

$$x^T A x = 1 \qquad x(I + 2\lambda A) = y$$

$$x^* = y(2\lambda A + I)^{-1}$$

Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A \hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A \hat{x}$$

Since $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A \hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T \hat{x}$$

Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A \hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A \hat{x}$$

$\Lambda^{\frac{1}{2}} = \text{diag}(\sqrt{\lambda_i})$

Since $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A \hat{x} = \frac{1}{2}\hat{x}^T Q \Lambda Q^T \hat{x} = \frac{1}{2}\underbrace{\hat{x}^T Q \Lambda^{\frac{1}{2}}}_{x^T}\,\underbrace{}_{I}\,\underbrace{\Lambda^{\frac{1}{2}} Q^T \hat{x}}_{x}$$

Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A \hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A \hat{x}$$

Since $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A \hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T \hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^T \hat{x} = \frac{1}{2}x^T I x$$

Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A \hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A \hat{x}$$

Since $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A \hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T \hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}} \boxed{\Lambda^{\frac{1}{2}} Q^T \hat{x}} = \frac{1}{2}x^T I x \qquad \text{if } x = \Lambda^{\frac{1}{2}} Q^T \hat{x}$$

Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A \hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A \hat{x}$$

Since $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A \hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T \hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}} Q^T \hat{x} = \frac{1}{2}x^T I x \qquad \text{if } x = \Lambda^{\frac{1}{2}} Q^T \hat{x} \text{ and } \hat{x} = Q\Lambda^{-\frac{1}{2}} x$$

Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A\hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A\hat{x}$$

Since $A = Q\Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda Q^T\hat{x} = \frac{1}{2}\hat{x}^T Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^T\hat{x} = \frac{1}{2}x^T I x \qquad \text{if } x = \Lambda^{\frac{1}{2}}Q^T\hat{x} \text{ and } \hat{x} = Q\Lambda^{-\frac{1}{2}}x$$

Conjugate directions. A-orthogonality.

Suppose, we have two coordinate systems and some quadratic function $f(x) = \frac{1}{2}x^T I x$ looks just like on the left part of Figure 2, while in another coordinates it looks like $f(\hat{x}) = \frac{1}{2}\hat{x}^T A \hat{x}$, where $A \in \mathbb{S}^d_{++}$.

$$\frac{1}{2}x^T I x \qquad\qquad \frac{1}{2}\hat{x}^T A \hat{x}$$

Since $A = Q \Lambda Q^T$:

$$\frac{1}{2}\hat{x}^T A \hat{x} = \frac{1}{2}\hat{x}^T Q \Lambda Q^T \hat{x} = \frac{1}{2}\hat{x}^T Q \Lambda^{\frac{1}{2}} \Lambda^{\frac{1}{2}} Q^T \hat{x} = \frac{1}{2}x^T I x \qquad \text{if } x = \Lambda^{\frac{1}{2}} Q^T \hat{x} \text{ and } \hat{x} = Q \Lambda^{-\frac{1}{2}} x$$

A-orthogonal vectors

$$\|x\|_A^2 = x^T A x$$

Vectors $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^d$ are called A-orthogonal (or A-conjugate) if

$$x^T A y = 0 \qquad \Leftrightarrow \qquad x \perp_A y$$

When $A = I$, A-orthogonality becomes orthogonality.

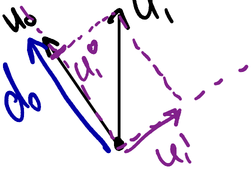
Gram–Schmidt process

ортогонализация

Вход: $u_0, \ldots, u_{n-1}$ — n ЛНЗ векторов

Выход: $d_0, \ldots, d_{n-1}$ — n ЛНЗ $\perp$ попарно векторов

пример:



$$d_0 = u_0$$

$$d_1 = u_1 - \pi_{d_0}(u_1)$$

"u_1'"

$$d_2 = u_2 - \pi_{d_0}(u_2) - \pi_{d_1}(u_2)$$

$$\pi_{d_i}(u_k) = \frac{d_i^T u_k}{d_i^T d_i} d_i$$

$$d_k = u_k + \sum_{i=0}^{k-1} \beta_{ik} \cdot d_i$$

$$\beta_{ik} = - \frac{d_i^T u_k}{d_i^T d_i}$$

Gram–Schmidt process

Идея метода сопр напр.
РАЗЛОЖИТЬ вектор

$$x^0 - x^* = \sum_{i=0}^{d-1} \alpha_i \cdot d_i$$

d_i – попарно A-ортогональны

α_i наискор. спуск
d_i Грам-Шмидт в смысле $\perp_A$

Idea of the method of conjugate directions

Thus, we formulate an algorithm:

1. Let $k = 0$ and $x_k = x_0$, count $d_k = d_0 = -\nabla f(x_0)$.

Idea of the method of conjugate directions

Thus, we formulate an algorithm:

1. Let $k = 0$ and $x_k = x_0$, count $d_k = d_0 = -\nabla f(x_0)$.
2. By the procedure of line search we find the optimal length of step. Calculate α minimizing $f(x_k + \alpha_k d_k)$ by the formula
$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k}$$

Idea of the method of conjugate directions

Thus, we formulate an algorithm:

1. Let $k = 0$ and $x_k = x_0$, count $d_k = d_0 = -\nabla f(x_0)$.
2. By the procedure of line search we find the optimal length of step. Calculate α minimizing $f(x_k + \alpha_k d_k)$ by the formula

$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k}$$

3. We're doing an algorithm step:

$$x_{k+1} = x_k + \alpha_k d_k$$

Idea of the method of conjugate directions

$$d_{k+1}^\top A d_k = 0$$

$$d_k^\top A(-\nabla f(x^{k+1}) + \beta_k d_k) = 0$$

$$-d_k^\top A \nabla f(x^{k+1}) + \beta_k d_k^\top A d_k = 0$$

Thus, we formulate an algorithm:

1. Let $k = 0$ and $x_k = x_0$, count $d_k = d_0 = -\nabla f(x_0)$.
2. By the procedure of line search we find the optimal length of step. Calculate α minimizing $f(x_k + \alpha_k d_k)$ by the formula

$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k}$$

3. We're doing an algorithm step:

$$x_{k+1} = x_k + \alpha_k d_k$$

4. update the direction: $d_{k+1} = -\nabla f(x_{k+1}) + \beta_k d_k$, where β_k is calculated by the formula:

$$\beta_k = \frac{\nabla f(x_{k+1})^\top A d_k}{d_k^\top A d_k}.$$

$$d_{k+1} \perp_A d_k$$

Idea of the method of conjugate directions

Thus, we formulate an algorithm:

1. Let $k = 0$ and $x_k = x_0$, count $d_k = d_0 = -\nabla f(x_0)$.
2. By the procedure of line search we find the optimal length of step. Calculate α minimizing $f(x_k + \alpha_k d_k)$ by the formula
$$\alpha_k = -\frac{d_k^\top (Ax_k - b)}{d_k^\top A d_k}$$
3. We're doing an algorithm step:
$$x_{k+1} = x_k + \alpha_k d_k$$
4. update the direction: $d_{k+1} = -\nabla f(x_{k+1}) + \beta_k d_k$, where β_k is calculated by the formula:
$$\beta_k = \frac{\nabla f(x_{k+1})^\top A d_k}{d_k^\top A d_k}.$$
5. Repeat steps 2-4 until n directions are built, where n is the dimension of space (dimension of x).

Method of Conjugate Directions

If a set of vectors $d_1, \ldots, d_k$ - are A-conjugate (each pair of vectors is A-conjugate), these vectors are linearly independent. $A \in \mathbb{S}^n_{++}$.

Proof

We'll show, that if $\sum\limits_{i=1}^{n} \alpha_k d_k = 0$, than all coefficients should be equal to zero:

$$
\begin{aligned}
0 &= \sum_{i=1}^{n} \alpha_k d_k \\
&= d_j^\top A \left(\sum_{i=1}^{n} \alpha_k d_k \right) \\
&= \sum_{i=1}^{n} \alpha_k d_j^\top A d_k \\
&= \alpha_j d_j^\top A d_j + 0 + \ldots + 0
\end{aligned}
$$

Thus, $\alpha_j = 0$, for all other indices one have perform the same process

Conjugate Gradients

Обозначения: $r^k = b - Ax^k$ – невязка (residual)

$\nabla f = Ax - b$

$$r^k = -A e^k$$

$e^k = x^k - x^*$ – ошибка (error)

потому что $Ax^* = b$

Лемма 1 процедура сх-ся ровно за n шагов (n-размерное пр-ва)

$$\min_{x \in \mathbb{R}^n} f(x)$$

пусть есть n ⊥A напр.:

$d_0, \ldots, d_{n-1}$ $\quad x^{k+1} = x^0 + \sum_{i=0}^{k} \alpha_i d_i$,

где α^i подбирается из Line Search: $\quad \alpha_i = \dfrac{d_i^T r_i}{d_i^T A d_i}$

То есть: $e^0 = x^0 - x^* = \sum_{i=0}^{n-1} \delta_i d_i$ $\quad (*)$

Док-во

1) пусть $\delta_i = -\alpha_i$

$x^0 + \sum_{i=0}^{n-1} \alpha_i d_i = x^*$

2) Фиксируем индекс k

$d_k^T A \, (*)$: $\quad d_k^T A e^0 = \sum_{i=0}^{n-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$

(из-за A орт.) $\Rightarrow$

Conjugate Gradients

$$d_k^T A e^0 = \sum_{i=0}^{k-1} \delta_i d_k^T A d_i = \delta_k d_k^T A d_k$$

(из-за A орт.) $\Rightarrow$

$$d_k^T A \Big(\underbrace{e^0 + \sum_{i=0}^{k-1} \alpha_i d_i}_{e^k} \Big) = \delta_k d_k^T A d_k$$

e^k — отмечаем 0

$$\delta_k = \frac{d_k^T A e^k}{d_k^T A d_k} = -\frac{d_k^T r_k}{d_k^T A d_k} = -\alpha_k$$

ч.т.д.

Conjugate Gradients

conjugate directions, где

в качестве векторов $d_0, \dots, d_{n-1}$

выбираем

$$GS^{\perp_A}(r_0, \dots, r_{n-1})$$

Conjugate Gradients

$GS_{\perp_A}$: Базис: $u_0 \ldots\ldots u_{n-1}$

Новый базис: $d_0, \ldots\ldots d_{n-1}$

$$d_i = u_i + \sum_{j=0}^{i-1} \beta_{ij} d_j \quad (GS), \qquad \beta_{ij} = -\frac{u_i^T A d_j}{d_j^T A d_j} \quad (B)$$

Лемма 2

$$e^i = e^0 + \sum_{j=0}^{i-1} \alpha_j d_j \ominus \quad \Bigg| \quad e^0 = x^0 - x^* = -\sum_{j=0}^{n-1} \alpha_j d_j$$

$$\ominus -\sum_{j=0}^{n-1} \alpha_j d_j + \sum_{j=0}^{i-1} \alpha_j d_j = \sum_{j=i}^{n-1} -\alpha_j d_j \quad (ER)$$

Conjugate Gradients

Conjugate Gradients

Лемма 3

(ER) для фикс. k: $e^k = -\sum_{j=k}^{n-1} \alpha_j d_j \; (E_k)$

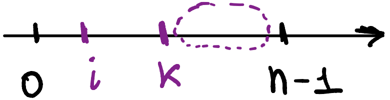
для некоторого i $\quad -d_i^T A \cdot (E_k)$

$$-d_i^T A e^k = +\sum_{j=k}^{n-1} \alpha_j d_i^T A d_j$$

если $i < k$

$$-d_i^T A e^k = 0$$

$$\boxed{d_i^T r^k = 0}$$



Таким образом, r^k перпендикулярен всем предыдущим направлениям d_i

Conjugate Gradients

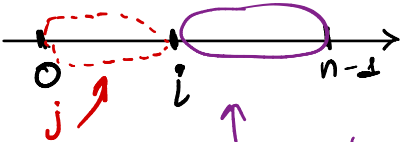
Лемма 4

$r^{k^T} \cdot$ (GS)

$$d_i = u_i + \sum_{j=0}^{i-1} \beta_{ij} d_j \quad (GS)$$

$$r^{k^T} d_i = r^{k^T} u_i + \sum_{j=0}^{i-1} \beta_{ij} r^{k^T} d_j$$

$j < k$



если k: $r^{k^T} d_j = 0$ (лемма 3)

Пусть $k > i$: $(i < k)$

$$\underbrace{r^{k^T} d_i}_{0} = r^{k^T} u_i$$

0 лемма 3

$$\boxed{u_i^T r^k = 0} \quad i < k$$

В CG:

$$u_i = r^i \Rightarrow \boxed{r^{i^T} r^k = 0 \quad i < k}$$

невязка $\perp$ всем предыдущим невязкам

Conjugate Gradients

Conjugate Gradients

пусть $k = i$

$$r^{k^T} d_i = r^{k^T} u_i + \sum_{j=0}^{i-1} \beta_{ij} r^{k^T} d_j$$

$$r^{k^T} d_k = r^{k^T} u_k + 0 \Rightarrow \boxed{u_k^T r^k = d_k^T r^k}$$

Лемма 5

$$r^{i+1} = -Ae^{i+1} = -A(e^i + \alpha_i d_i) = -Ae^i - \alpha_i A d_i = r^i - \alpha_i A d_i$$

$$\boxed{r^{i+1} = r^i - \alpha_i A d_i}$$

Conjugate Gradients

Рассмотрим β_{ij} в (GS)

$$\beta_{ij} = -\frac{u_i^T A d_j}{d_j^T A d_j} = -\frac{r_i^T A d_j}{d_j^T A d_j}$$

В случае CG: $u_i = r^i$

ОКАЗЫВАЕТСЯ, ЧТО β_{ij} ПОЧТИ ВСЕГДА $= 0$, кроме случаев соседних i, j

Для этого рассмотрим:

$$\langle r^i, r^{j+1} \rangle = \langle r^i, r^j - \alpha_j A d_j \rangle = \langle r^i, r^j \rangle - \alpha_j \langle r^i, A d_j \rangle$$

$$\Rightarrow \quad \alpha_j \langle r^i, A d_j \rangle = \langle r^i, r^j \rangle - \langle r^i, r^{j+1} \rangle$$

если $i = j$

$$\alpha_i \langle r^i, A d_i \rangle = \langle r^i, r^i \rangle - \langle r^i, r^{i+1} \rangle \to 0$$

если $i = j+1$

$$\alpha_j \langle r^i, A d_j \rangle = -\langle r^i, r^i \rangle$$

ИНАЧЕ $\Rightarrow \langle r^i, A d_j \rangle = 0$

Conjugate Gradients

Conjugate Gradients

Концепт:

Вспоминаем

$j < i$

$j = i - 1$

$x^0, \ d_0 = -\nabla f(x^0) = r^0$

$x^{k+1} = x^k + \alpha^k \cdot d^k$

$d^{k+1} = GS(r_0, \ldots)$

$\beta_{ij} = -\frac{r_i^T A d_j}{d_j^T A d_j} = $

$= +\frac{1}{\alpha_j} \frac{\langle r^i, r^i \rangle}{d_j^T A d_j} = + \frac{d_j^T A d_j}{d_j^T r_j} \cdot \frac{\langle r^i, r^i \rangle}{d_j^T A d_j} = \frac{\langle r^i, r^i \rangle}{\langle r^j, d^j \rangle} =$

$= \frac{\langle r^i, r^i \rangle}{\langle r^{i-1}, r^{i-1} \rangle}$

$\alpha_j = \frac{d_j^T r_j}{d_j^T A d_j}$

Conjugate gradient method

Conjugate Gradient = Conjugate Directions

+ Residuals as starting vectors for Gram–Schmidt

$\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$

if $\mathbf{r}_0$ is sufficiently small, then return $\mathbf{x}_0$ as the result

$\mathbf{d}_0 := \mathbf{r}_0$

$k := 0$

repeat

$$\alpha_k := \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{d}_k^\top \mathbf{A}\mathbf{d}_k}$$

наиск. спуск

$$\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{d}_k$$

$$\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{d}_k$$

if $\mathbf{r}_{k+1}$ is sufficiently small, then exit loop

$$\beta_k := \frac{\mathbf{r}_{k+1}^\top \mathbf{r}_{k+1}}{\mathbf{r}_k^\top \mathbf{r}_k}$$

$$\mathbf{d}_{k+1} := \mathbf{r}_{k+1} + \beta_k \mathbf{d}_k$$

GS

$k := k + 1$

end repeat

return $\mathbf{x}_{k+1}$ as the result

Convergence

Theorem 1. If matrix A has only r different eigenvalues, then the conjugate gradient method converges in r iterations.

Theorem 2. The following convergence bound holds

$$\|x_k - x^*\|_A \leq 2\left(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\right)^k \|x_0 - x^*\|_A,$$

where $\|x\|_A^2 = x^\top Ax$ and $\kappa(A) = \frac{\lambda_1(A)}{\lambda_n(A)}$ is the conditioning number of matrix A, $\lambda_1(A) \geq ... \geq \lambda_n(A)$ are the eigenvalues of matrix A

Note: compare the coefficient of the geometric progression with its analog in gradient descent.

Non-linear conjugate gradient method

In case we do not have an analytic expression for a function or its gradient, we will most likely not be able to solve the one-dimensional minimization problem analytically. Therefore, step 2 of the algorithm is replaced by the usual line search procedure. But there is the following mathematical trick for the fourth point:

For two iterations, it is fair:

$$x_{k+1} - x_k = cd_k,$$

where c is some kind of constant. Then for the quadratic case, we have:

$$\nabla f(x_{k+1}) - \nabla f(x_k) = (Ax_{k+1} - b) - (Ax_k - b) = A(x_{k+1} - x_k) = cAd_k$$

Expressing from this equation the work $Ad_k = \dfrac{1}{c}\left(\nabla f(x_{k+1}) - \nabla f(x_k)\right)$, we get rid of the “knowledge” of the function in step definition β_k, then point 4 will be rewritten as:

$$\beta_k = \frac{\nabla f(x_{k+1})^\top (\nabla f(x_{k+1}) - \nabla f(x_k))}{d_k^\top (\nabla f(x_{k+1}) - \nabla f(x_k))}.$$

This method is called the Polack - Ribier method.

Preconditioned conjugate gradient method