Numerical Methods I

The Gradient and Conjugate Gradient Method

Marcel Oliver

1 Reformulation as an Optimization Problem

Let $A \in \mathbb{R}^{n \times n}$ be a symmetric, positive definite matrix. Then $A\boldsymbol{x} = \boldsymbol{b}$ if and only if $\boldsymbol{x}$ minimizes the function

$$\Phi(\boldsymbol{x}) = \tfrac{1}{2}\,\boldsymbol{x}^T A \boldsymbol{x} - \boldsymbol{x}^T \boldsymbol{b}\,. \tag{1}$$

Proof. Assume first that $\boldsymbol{x}$ is a minimizer of Φ. Thus, $\Phi(\boldsymbol{x} + \lambda\,\boldsymbol{v})$ must have a minimum at $\lambda = 0$ for any fixed vector $\boldsymbol{v} \in \mathbb{R}^n$. In other words,

$$\frac{d}{d\lambda}\Phi(\boldsymbol{x} + \lambda\,\boldsymbol{v})\Big|_{\lambda=0} = 0\,. \tag{2}$$

We compute

$$\begin{aligned} \frac{d}{d\lambda}\Phi(\boldsymbol{x} + \lambda\,\boldsymbol{v}) &= \tfrac{1}{2}\,(\boldsymbol{x} + \lambda\,\boldsymbol{v})^T A\boldsymbol{v} + \tfrac{1}{2}\,\boldsymbol{v}^T A(\boldsymbol{x} + \lambda\,\boldsymbol{v}) - \boldsymbol{v}^T\boldsymbol{b} \\ &= \boldsymbol{v}^T\,\big(A(\boldsymbol{x} + \lambda\,\boldsymbol{v}) - \boldsymbol{b}\big)\,. \end{aligned} \tag{3}$$

Setting $\lambda = 0$, we see that we must require

$$\boldsymbol{v}^T\,(A\boldsymbol{x} - \boldsymbol{b}) = 0\,. \tag{4}$$

Since $\boldsymbol{v}$ is arbitrary, this implies that $A\boldsymbol{x} = \boldsymbol{b}$.

Vice versa, assume that $A\boldsymbol{x} = \boldsymbol{b}$. Then, for any $\boldsymbol{y} \in \mathbb{R}^n$,

$$\begin{aligned} \Phi(\boldsymbol{y}) - \Phi(\boldsymbol{x}) &= \tfrac{1}{2}\,\boldsymbol{y}^T A\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{b} - \tfrac{1}{2}\,\boldsymbol{x}^T A\boldsymbol{x} + \boldsymbol{x}^T\boldsymbol{b} \\ &= \tfrac{1}{2}\,\boldsymbol{y}^T A\boldsymbol{y} - \boldsymbol{y}^T A\boldsymbol{x} - \tfrac{1}{2}\,\boldsymbol{x}^T A\boldsymbol{x} + \boldsymbol{x}^T A\boldsymbol{x} \\ &= \tfrac{1}{2}\,(\boldsymbol{y} - \boldsymbol{x})^T A(\boldsymbol{y} - \boldsymbol{x}) \\ &\geq 0 \end{aligned} \tag{5}$$

with equality if and only if $\boldsymbol{x} = \boldsymbol{y}$ since A is positive definite. We conclude that $\boldsymbol{x}$ is the unique minimizer of Φ. □

2 The Gradient Method

The algorithm works works as follows:

- Choose a descent direction $\boldsymbol{d}_k$;
- Walk in the descent direction until you reach a minimum along the line;
- Repeat.

There are many ways of choosing descent directions. The simplest is to take the direction of steepest descent,

$$\boldsymbol{d}_k = -\boldsymbol{\nabla}\Phi(\boldsymbol{x}_k) = \boldsymbol{b} - A\boldsymbol{x}_k = \boldsymbol{r}_k \,. \tag{6}$$

Given $\boldsymbol{x}_k$, compute the next iterate via

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \, \boldsymbol{d}_k \tag{7}$$

where α_k is chosen such that

$$\frac{d}{d\alpha_k}\Phi(\boldsymbol{x}_k + \alpha_k \, \boldsymbol{d}_k) = 0 \,. \tag{8}$$

Following the computation leading up to (3), we find that

$$0 = \boldsymbol{d}_k^T \left(A(\boldsymbol{x}_k + \alpha_k \, \boldsymbol{d}_k) - \boldsymbol{b} \right) , \tag{9}$$

and therefore

$$\alpha_k = \frac{\boldsymbol{d}_k^T \boldsymbol{r}_k}{\boldsymbol{d}_k^T A \boldsymbol{d}_k} \,. \tag{10}$$

In summary, one iteration of the gradient method consists of the steps

$$\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k \, , \tag{11}$$

$$\alpha_k = \frac{\boldsymbol{r}_k^T \boldsymbol{r}_k}{\boldsymbol{r}_k^T A \boldsymbol{r}_k} \, , \tag{12}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \, \boldsymbol{r}_k \,. \tag{13}$$

3 The Conjugate Gradient Method

The conjugate gradient method is based on the concept of optimality with respect to a set of search directions. Once the algorithm has reached optimality in some direction, we allow only changes that are in a certain sense orthogonal, thereby preserving optimality under iteration.

Specifically, we say that a point $\boldsymbol{x} \in \mathbb{R}^n$ is optimal with respect to a subspace $V \subset \mathbb{R}^n$ if Φ has a minimum at $\boldsymbol{x}$ along each line passing through $\boldsymbol{x}$ in a direction $\boldsymbol{v} \in V$.

Repeating the calculation that lead up to (4), we find that $\boldsymbol{x}$ is optimal with respect to V if

$$\boldsymbol{r}^T \boldsymbol{v} = 0 \quad \text{for all } \boldsymbol{v} \in V \,, \tag{14}$$

where $\boldsymbol{r} = \boldsymbol{b} - A\boldsymbol{x}$ is the corresponding residual.

Now assume that $\boldsymbol{x}_k$ is optimal with respect to some subspace V_k. We would like to find a condition on a new descent direction $\boldsymbol{d}_k$ so that the next iterate under

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \, \boldsymbol{d}_k \tag{15}$$

is optimal not only with respect to the new descent direction $\boldsymbol{d}_k$, as in the gradient method, but also with respect to all old descent directions V_k. Optimality with respect to the new direction implies, as for the gradient method,

$$\alpha_k = \frac{\boldsymbol{d}_k^T \boldsymbol{r}_k}{\boldsymbol{d}_k^T A \boldsymbol{d}_k} \,. \tag{16}$$

To get optimality conditions with respect to V_k, we multiply (15) by $-A$ from the left and add $\boldsymbol{b}$, so that

$$\boldsymbol{r}_{k+1} = \boldsymbol{r}_k - \alpha_k \, A\boldsymbol{d}_k \,. \tag{17}$$

Optimality with respect to V_k for $\boldsymbol{x}_{k+1}$ and $\boldsymbol{x}_k$ means that

$$\boldsymbol{r}_{k+1}^T \boldsymbol{v} = \boldsymbol{r}_k^T \boldsymbol{v} = 0 \quad \text{for all } \boldsymbol{v} \in V_k \,, \tag{18}$$

and therefore

$$\boldsymbol{v}^T A \boldsymbol{d}_k = 0 \quad \text{for all } \boldsymbol{v} \in V_k \,. \tag{19}$$

In other words, $\boldsymbol{d}_k$ must be A-orthogonal to all directions in V_k.

The conjugate gradient method now works as follows. The first descent direction is chosen as for the gradient method, namely $\boldsymbol{d}_1 = \boldsymbol{r}_1$. Each subsequent descent direction is the A-orthogonalization of $\boldsymbol{r}_k$ with respect to the space of old descent directions

$$V_k = \operatorname{Span}\{\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{k-1}\} \,. \tag{20}$$

Following this construction, each new descent direction $\boldsymbol{d}_k$ is ultimately a linear combination of all previous residuals $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_k$. In particular, we see that

$$V_k = \operatorname{Span}\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{k-1}\} \,. \tag{21}$$

Thus, equation (17) gives a recursion relation for the spaces V_k, namely[1]

$$V_{k+1} \subset V_k \oplus AV_k \,. \tag{22}$$

[1] In fact, it is easy to see that equality holds unless the algorithm terminates with the exact solution, and that

$$\begin{aligned} V_{k+1} &= \operatorname{Span}\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{k-1}, A\boldsymbol{d}_{k-1}\} \\ &= \operatorname{Span}\{A^0\boldsymbol{r}_1, A^1\boldsymbol{r}_1, \ldots, A^{k-1}\boldsymbol{r}_1\} \\ &= \operatorname{Span}\{A^0\boldsymbol{d}_1, A^1\boldsymbol{d}_1, \ldots, A^{k-1}\boldsymbol{d}_1\} \,. \end{aligned}$$

We now analyze the A-orthogonality condition when stepping from $\boldsymbol{x}_k$ to $\boldsymbol{x}_{k+1}$ in detail. First, $\boldsymbol{x}_k$ is already optimal with respect to V_k—this has been achieved in the previous step of the iteration—so that

$$\boldsymbol{r}_k^T \boldsymbol{v} = 0 \quad \text{for all } \boldsymbol{v} \in V_k \,. \tag{23}$$

Due to (22), this implies, in particular, that

$$\boldsymbol{r}_k^T A \boldsymbol{v} = 0 \quad \text{for all } \boldsymbol{v} \in V_{k-1} \,, \tag{24}$$

which is to say that

$$\boldsymbol{r}_k^T A \boldsymbol{d}_j = 0 \quad \text{for all } j = 1, \dots, k-2 \,. \tag{25}$$

In other words, $\boldsymbol{r}_k$ is already A-orthogonal to all but the $(k-1)$-st previous descent direction. This leads to a tremendous simplification of the orthogonalization step; the Gram–Schmidt-procedure needs only one projection, so that

$$\boldsymbol{d}_k = \boldsymbol{r}_k - \frac{\boldsymbol{r}_k^T A \boldsymbol{d}_{k-1}}{\boldsymbol{d}_{k-1}^T A \boldsymbol{d}_{k-1}} \, \boldsymbol{d}_{k-1} \tag{26}$$

for $k \geq 2$.

We summarize the conjugate gradient method:

$$\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k \,, \tag{27}$$

$$\boldsymbol{d}_k = \begin{cases} \boldsymbol{r}_1 & \text{for } k = 1 \\ \boldsymbol{r}_k - \dfrac{\boldsymbol{r}_k^T A \boldsymbol{d}_{k-1}}{\boldsymbol{d}_{k-1}^T A \boldsymbol{d}_{k-1}} \, \boldsymbol{d}_{k-1} & \text{for } k \geq 2 \,, \end{cases} \tag{28}$$

$$\alpha_k = \frac{\boldsymbol{d}_k^T \boldsymbol{r}_k}{\boldsymbol{d}_k^T A \boldsymbol{d}_k} \,, \tag{29}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \, \boldsymbol{d}_k \,. \tag{30}$$

In exact arithmetic, one can show that either the dimension of V_k increases by one each iteration, or the exact solution is reached. This implies that the search space is exhausted after at most n iteration and the algorithm must terminate with the exact answer.