CG algorithm

Jinchen

November 2, 2021

1 Motivation

Conjugate gradient method is an algorithm for optimization problems. Here we just take an easy example:

$$f(x) = \frac{1}{2}x^T Qx - x^T b,$$

in which x is a vector in $\mathbb{R}^n$, Q is a $n \times n$ symmetric (IF Q is hermitian instead of symmetric, change all T to †) and positive-definite matrix.

We would like to find the x^* at where $f(x^*)$ has the minimum, so $f'(x^*) = Qx - b = 0$.

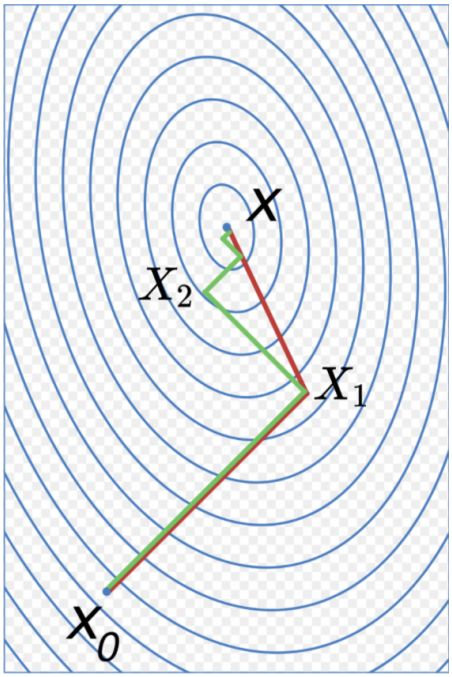

Figure 1

2 Gradient Descent

There is a direct way to find the minimum point of $f(x)$ which is shown by the green lines in the figure above:

- Start from a random initial point X_0
- Find the negative gradient direction d_1
- Find the minimum point of $f(x)$ along that direction d_1 as X_1
- The next direction d_2 should be orthognal to the previous direction d_1
- Find the minimum point of $f(x)$ along the direction d_2 as X_2
- Iterate to get closer to X^*

There are some obvious shortages of this method:

- Zig-zag route
- Sepetitious steps
- ...

So we need another better solution...

3 Conjugate Gradient Method

Firstly, let's give a specific definition:

$$\mathrm{d_i \text{ and } d_j \text{ are } Q-orthognal} :< \mathrm{d_i, d_j} >= \mathrm{d_i^T Q d_j} = \mathrm{constant} \cdot \delta_{\mathrm{ij}},$$

then we would like to find a group of Q-orthognal basis of $\mathbb{R}^n$: $\{d_1, d_2, \dots d_n\}$, so that we can approach x^* in each direction independently.

Assume $x^* = \sum_i \alpha_i^* d_i$, then we have $b = Qx^* = \sum_i \alpha_i^* Q d_i$, multiply the equation by d_k^T, we select the α_k^* out as $\alpha_k^* = \frac{d_k^T b}{d_k^T Q d_k}$.

Now let's try to find $\{d_1, d_2 \dots d_n\}$ and to get x^*.

We have an initial guess x_1, the negative gradient at x_1 is $d_1 = b - Qx_1$, calculate

$$\alpha_1 = \frac{d_1^T d_1}{d_1^T Q d_1},$$

then go to the next point $x_2 = x_1 + \alpha_1 d_1$.

The gradient at x_2 is $g_2 = Qx_2 - b$, calculate

$$\beta_1 = \frac{g_2^T Q d_1}{d_1^T Q d_1},$$

and the next direction is $d_2 = -g_2 + \beta_1 d_1$, which is Q-orthognal to d_1

$$d_2^T Q d_1 = -g_2^T Q d_1 + \beta_1 d_1^T Q d_1 = 0,$$

so, just follow the steps above to get d_n,

$$\beta_k = \frac{g_{k+1}^T Q d_k}{d_k^T Q d_k},$$

and

$$d_{k+1} = -g_{k+1} + \beta_k d_k,$$

notice here

$$g_k = Q x_k - b.$$

And how long should x_k to go along d_k to get x_{k+1}?
Destination:

$$x^* = \sum_i \frac{d_i^T b}{d_i^T Q d_i} \cdot d_i = \sum_i \frac{d_i^T Q x^*}{d_i^T Q d_i} \cdot d_i$$

Where we are:

$$x_k = \sum_i \frac{x_k^T Q d_i}{d_i^T Q d_i} \cdot d_i$$

So,

$$\alpha_k = \frac{d_k^T b - x_k^T Q d_k}{d_k^T Q d_k} = \frac{-g_k^T d_k}{d_k^T Q d_k}$$

and

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

Iterate to get closer to x^*.

Why we need to define Q-orthognal?

If we choose a group of $\{d_1, d_2, \ldots d_n\}$, which satisfy $d_i^T d_j = constant \cdot \delta_{ij}$, then

$$b = \sum_i \beta_i d_i$$

$$Qx = \sum_i \alpha_i Q d_i = \sum_i \alpha_i' d_i$$

then when we change α_i to move x in one direction, all α_i' will be affected.

4 For Lattice

For example, we know $S_F = Dslash^{-1}$, $Dslash$ is so big that is hard to inverse, but it is symmetric and positive-definite. Therefore, we can use CG algorithm to solve

$$x = S_F b, b = (Dslash)x, \ for \ \forall b$$

just like for a random b, find x^* s.t. $b = Qx^*$. But in order to satisfy the positive-definite condition, usually we change to solve

$$D^\dagger b = (D^\dagger D)x$$

CG

Compute $r^{(0)} = b - Ax^{(0)}$ for some initial guess $x^{(0)}$
for $i = 1, 2, \ldots$
  solve $M z^{(i-1)} = r^{(i-1)}$
  $\rho_{i-1} = r^{(i-1)^T} z^{(i-1)}$
  if $i = 1$
    $p^{(1)} = z^{(0)}$
  else
    $\beta_{i-1} = \rho_{i-1}/\rho_{i-2}$
    $p^{(i)} = z^{(i-1)} + \beta_{i-1} p^{(i-1)}$
  endif
  $q^{(i)} = Ap^{(i)}$
  $\alpha_i = \rho_{i-1}/p^{(i)^T} q^{(i)}$
  $x^{(i)} = x^{(i-1)} + \alpha_i p^{(i)}$
  $r^{(i)} = r^{(i-1)} - \alpha_i q^{(i)}$
  check convergence; continue if necessary
end

here M=1

Figure 2: CG in lattice camp, from Peng Sun

Look at the sudo-code in the Fig.2, just adopt replacement of variables as below,

- $\alpha_k \to \alpha_i$
- $d_k \to p^{(i)}$
- $-g_{k+1} \to r^{(i)}$
- $\beta_k \to \beta_i$

This CG algorithm has some advantages, although the length of scequences can become large, only small number of vectors need to be kept in memory. For the simple CG algorithm convergence is guaranteed for positive definite symmetric matrices, at least after N steps for matrices of dimension N, but usually significantly faster.

5 Optimization

5.1 Even-Odd Precondition

$$M = \begin{bmatrix} M_{ee} & M_{eo} \\ M_{oe} & M_{oo} \end{bmatrix} \tag{1}$$

$$\begin{aligned} M &= \begin{bmatrix} 1 & 0 \\ M_{oe}M_{ee}^{-1} & 1 \end{bmatrix} \cdot \begin{bmatrix} M_{ee} & 0 \\ 0 & M_{oo} - M_{oe}M_{ee}^{-1}M_{eo} \end{bmatrix} \cdot \begin{bmatrix} 1 & M_{ee}^{-1}M_{eo} \\ 0 & 1 \end{bmatrix} \\ &= L\tilde{M}U \end{aligned} \tag{2}$$

Therefore, $M\phi = x$ becomes $\tilde{M}\phi' = x'$, in which $\phi = U^{-1}\phi'$ and $x' = L^{-1}x$. And we know,

$$\begin{bmatrix} 1 & 0 \\ A & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ -A & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{3}$$

5.2 Preconditioning methods

So-called preconditioning methods sometimes allow one to accelerate the convergence, depending on the Dirac operator. A preconditioner is a suitable matrix M which we use to transform the system to

$$M^{-1}A\boldsymbol{x} = M^{-1}\boldsymbol{b}.$$

Let us assume that we have a matrix M, which is numerically cheap to invert, i.e., to solve $M\hat{\boldsymbol{s}} = \boldsymbol{s}$ for $\hat{\boldsymbol{s}}$, and which approximates A in some way. Then the spectral properties of $M^{-1}A$ may be more favorable, allowing faster convergence of the iterative solution. This may be true in particular, if the small eigenvalues of M agree with those of A. For a more complete discussion of such methods please check:

- R. Barrett et al.: Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods (SIAM, Philadelphia 1994) 139, 140, 141
- I. Montvay and G. Münster: Quantum Fields on a Lattice (Cambridge University Press, Cambridge, New York 1994) 141