Practical Optimization Algorithms
实用优化算法

徐 翔

数学科学学院
浙江大学

Mar 29, 2021

第四讲：共轭梯度法（Conjugate Gradient）

THE LINEAR CG METHOD (线性共轭梯度法)

- 求解线性系统

$$Ax = b \tag{4.1}$$

  等价于求解如下的二次极小化问题:

$$\min f(x) = \frac{1}{2}x^T Ax - b^T x \tag{4.2}$$

  (4.1) and (4.2) 具有相同的唯一解.

- 记号: 梯度∇f 记为线性方程组的残量(residual),

$$\nabla f(x) = Ax - b \equiv r(x), \tag{4.3}$$

  在 $x = x_k$处有,

$$g_k = \nabla f(x_k) = r_k = Ax_k - b \tag{4.4}$$

Conjugate Direction Methods

Definition : 共轭方向

A set of nonzero vectors $\{p_0, p_1, \cdots, p_t\}$ is said to be conjugate with respect to the symmetric positive definite matrix A (SPD) if

$$p_i^T A p_j = 0, \quad \forall i \neq j. \tag{4.5}$$

Remark

- It is easy to show that any set of vectors satisfying this property is also linear independent. (给出证明)
- The importance of conjugacy lies in the fact that we can minimize $f(\cdot)$ in n steps by successively minimizing it along the individual directions in a conjugate set.

Conjugate Direction Methods

- Given a starting point $x_0 \in \mathcal{R}^n$ and a set of conjugate directions $\{p_0, p_1, \cdots, p_{n-1}\}$
- Generate the sequence $\{x_k\}$ by setting

$$x_{k+1} = x_k + \alpha_k p_k \tag{4.6}$$

where α_k is the one-dimensional minimizer of the quadratic function $f(\cdot)$ along $x_k + \alpha p_k$, given explicitly by

$$\alpha_k = -\frac{r_k^T p_k}{p_k^T A p_k} \tag{4.7}$$

- We have the following theorem

Theorem

For any $x_0 \in \mathcal{R}^n$, the sequence $\{x_k\}$ generated by the conjugate direction algorithm (4.6)-(4.7) converges to the solution x^* of the linear system (4.1) at most n steps. 二次终止性

Conjugate Direction Methods

Proof the theorem

- Since $\{p_i\}$ are linearly independent, they span the whole space $\mathcal{R}^n$. Hence, we can write the difference between x_0 and the solution x^* in the following way:
$$x^* - x_0 = \sigma_0 p_0 + \sigma_1 p_1 + \cdots + \sigma_{n-1} p_{n-1}$$
- By premultiplying this expression by $p_k^T A$ and using the conjugacy property (4.5), we obtain
$$\sigma_k = \frac{p_k^T A(x^* - x_0)}{p_k^T A p_k} \quad (? = \alpha_k) = -\frac{p_k^T r_k}{p_k^T A p_k}$$
- To prove σ_k coincide with the step lengths α_k by showing $p_k^T A(x^* - x_0) = -p_k^T r_k$.

Conjugate Direction Methods

Proof

- 目的：$p_k^T A(x^* - x_0) = -p_k^T r_k$
- Suppose x_k is generated by algorithm (4.6)-(4.7), then we have

$$x_k = x_0 + \alpha_0 p_0 + \alpha_1 p_1 + \cdots + \alpha_{k-1} p_{k-1}$$

- By premultiplying this expression by $p_k^T A$ and using the conjugacy property, we have that

$$p_k^T A(x_k - x_0) = 0, \text{ i.e., } p_k^T A x_k = p_k^T A x_0$$

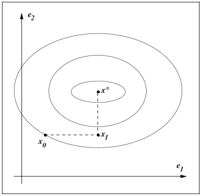
- Therefore

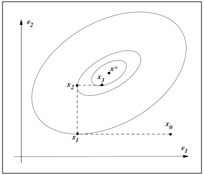
$$p_k^T A(x^* - x_0) = p_k^T A(x^* - x_k) = p_k^T (b - A x_k) = -p_k^T r_k.$$

Conjugate Direction Methods

A Simple Interpretation

- If the matrix A is diagonal, the contours of the function $\phi(\cdot)$ are ellipses whose axes are aligned with the coordinate directions.





- We can find the minimizer of this function by performing one-dimensional minimizations along the coordinate directions $e_1, e_2, \cdots, e_n$. Successive minimizations along the coordinate directions find the minimizer of a quadratic with a diagonal Hessian in n iterations.
- When A is not diagonal, its contours are still elliptical, but they are usually no longer aligned with the coordinate directions. The strategy of successive minimization along these directions in turn no longer leads to the solution in n iterations (or even in a finite number of iterations).

Conjugate Direction Methods （共轭方向法）

- We can recover the nice behavior if we transform the general problem to make A diagonal and then minimize along the coordinate directions.
- Suppose we transform the problem by defining new variables $\hat{x}$ as

$$\hat{x} = S^{-1}x \text{ where } S = [p_0, p_1, \cdots, p_{n-1}]$$

- The quadratic now becomes

$$\hat{\phi}(\hat{x}) \equiv \phi(S\hat{x}) = \frac{1}{2}\hat{x}^T(S^TAS)\hat{x} - (S^Tb)^T\hat{x}$$

- By the conjugacy property, the matrix S^TAS is diagonal, so we can find the minimizing value of $\hat{\phi}$ by performing n one-dimensional minimizations along the coordinate directions of $\hat{x}$.

Conjugate Direction Methods （共轭方向法）

Another interesting property:

- When the Hessian matrix is diagonal, each coordinate minimization correctly determines one of the components of the solution x^*.
- In other words, after k one-dimensional minimizations, the quadratic has been minimized on the subspace spanned by $e_1, e_2, \cdots, e_k$.
- The following theorem proves this important result for the general case in which the Hessian of the quadratic is not necessarily diagonal.

Conjugate Direction Methods （共轭方向法）

Theorem (Expanding Subspace Minimization)

Let $x_0 \in \mathcal{R}^n$ be any starting point and suppose that the the sequence $\{x_k\}$ is generated by the conjugate direction algorithm (4.6)-(4.7).
Then

$$r_k^T p_i = 0, \text{ for } i = 0, 1, \cdots, k-1, \tag{4.8}$$

and x_k is the minimizer of $\phi(x) = \frac{1}{2}x^T Ax - b^T x$ over the set $\mathcal{N}_k$

$$\mathcal{N}_k = \left\{ x | x = x_0 + \text{span}\{p_0, p_1, \cdots, p_{k-1}\} \right\}. \tag{4.9}$$

Proof.

- Utilizing induction (归纳法) to prove (4.8).
- Assume the minimizer of ϕ over $\mathcal{N}_k$ is $\tilde{x} = x_0 + \sigma_0 p_0 + \cdots + \sigma_{k-1}p_{k-1}$, prove $\sigma_k = \alpha_k$, i.e., to calculate $\frac{\partial \phi(\tilde{x})}{\partial \sigma_i} = 0$ to derive $\sigma_i = \alpha_i$.

□

Conjugate Direction Methods （共轭方向法）

- The fact that the current residual r_k is orthogonal to all previous search directions is a property that will be used extensively in the following (当前的残量r_k与之前所有的搜索方向p_k都是正交的).
- The discussion so far has been general, in that it applies to a conjugate direction method based on any choice of the conjugate direction set $\{p_0, p_1, \cdots, p_{n-1}\}$.
- (如何选择共轭方向？) There are many ways to choose the set of conjugate directions. For instance,
  - the eigenvectors $v_1, v_2, \cdots, v_n$ of A are mutually orthogonal as well as conjugate with respect to A.(验证)
  - the Gram-Schmidt orthogonalization process can be modified to produce a set of conjugate directions rather than a set of orthogonal directions. However, the Gram–Schmidt approach is also expensive, since it requires us to store the entire direction set.

Basic Property of the CG Method

- The conjugate gradient method(共轭梯度法) is a conjugate direction method(共轭方向法) with a very special property
- In generating its set of conjugate vectors, it can compute a new vector p_k by using only the previous vector p_{k-1}.
- It does not need to know all the previous elements $p_0, p_1, \cdots, p_{k-1}$ of the conjugate set; p_k is automatically conjugate to these vectors (自动与前面所有方向共轭).
- The CG method requires little storage and computation.

Details of the CG Method

- Each direction p_k is chosen to be a linear combination of the steepest descent direction $-\nabla f(x_k)$ (which is the same as the negative residual r_k) and the previous direction p_{k-1}.

$$p_k = -r_k + \beta_k p_{k-1} \tag{4.10}$$

- By premultiplying (4.10) by $p_{k-1}^T A$ and imposing the condition $p_{k-1}^T A p_k = 0$, we find that

$$\beta_k = \frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}}$$

- It makes intuitive sense to choose the first search direction $p_0 = -\nabla f(x_0)$, i.e., the steepest descent direction at the initial point x_0.
- As in the general conjugate direction method, we perform successive one-dimensional minimizations along each of the search directions.

Conjugate Gradient Method

Algorithm 1: (CG-Preliminary Version)

```
Given $x_0$;
Set $r_0 \leftarrow Ax_0 - b$, $p_0 \leftarrow -r_0$, $k \leftarrow 0$;
while $r_k \neq 0$, do
    $\alpha_k \leftarrow -\frac{r_k^T p_k}{p_k^T A p_k}$;
    $x_{k+1} \leftarrow x_k + \alpha_k p_k$;
    $r_{k+1} \leftarrow Ax_{k+1} - b$;
    $\beta_{k+1} \leftarrow \frac{r_{k+1}^T A p_k}{p_k^T A p_k}$;
    $p_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} p_k$;        $k \leftarrow k + 1$;
End(while)
```

Conjugate Gradient Method

Theorem

- Suppose that the k-th iterate generated by the conjugate gradient method is not the solution point x^*.
- The following four properties hold:

$$r_k^T r_i = 0, \qquad \forall i = 0, \cdots, k-1, \tag{4.11a}$$

$$\text{span}\{r_0, r_1, \cdots, r_k\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}, \tag{4.11b}$$

$$\text{span}\{p_0, p_1, \cdots, p_k\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}, \tag{4.11c}$$

$$p_k^T A p_i = 0, \qquad \forall i = 0, 1, \cdots, k-1. \tag{4.11d}$$

- Therefore, the sequence $\{x_k\}$ converges to x^* in at most n steps.

CONJUGATE GRADIENT METHOD

Remark

This theorem shows that

- the directions $p_0, p_1, \cdots p_{n-1}$ are indeed conjugate, which implies termination in n steps
- the residuals r_i are mutually orthogonal;
- each search direction p_k and residual r_k is contained in the Krylov subspace of degree k for r_0, defined as

$$\mathcal{K}(r_0; k) \equiv \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}.$$

Conjugate Gradient Method

Proof

- The proof is by induction.
- First prove $\text{span}\{r_0, r_1, \cdots, r_k\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}$ and $\text{span}\{p_0, p_1, \cdots, p_k\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}$
- $k = 0$ is trivial.
- The induction hypothesis
$$r_k \in \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}, \quad p_k \in \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}$$
- Multiplying the second expression by A, we obtain
$$Ap_k \in \text{span}\{Ar_0, A^2 r_0, \cdots, A^{k+1} r_0\}$$
- Since $r_{k+1} = A(x_k + \alpha_k p_k) - b = r_k + \alpha_k Ap_k$, then
$$r_{k+1} \in \text{span}\{r_0, Ar_0, A^2 r_0, \cdots, A^{k+1} r_0\}$$
- Thus
$$\text{span}\{r_0, r_1, \cdots, r_k, r_{k+1}\} \subset \text{span}\{r_0, Ar_0, \cdots, A^k r_0, A^{k+1} r_0\}$$

Conjugate Gradient Method

Proof

- To prove the reverse inclusion holds as well. Since $\text{span}\{p_0, p_1, \cdots, p_k\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}$, we have
$$A^{k+1} r_0 = A(A^k r_0) \in \text{span}\{Ap_0, Ap_1, \cdots, Ap_k\}$$
- Since $r_{i+1} = r_i + \alpha_i A p_i$, i.e., $Ap_i = (r_{i+1} - r_i)/\alpha_i$, we have
$$A^{k+1} r_0 \ \in \ \text{span}\{r_0, r_1, \cdots, r_k, r_{k+1}\}$$
- Since the induction hypothesis $\text{span}\{r_0, r_1, \cdots, r_k\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0\}$,
$$\text{span}\{r_0, Ar_0, \cdots, A^k r_0, A^{k+1} r_0\} \subset \text{span}\{r_0, r_1, \cdots, r_k, r_{k+1}\}$$
- Finally, we have
$$\text{span}\{r_0, r_1, \cdots, r_k, r_{k+1}\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0, A^{k+1} r_0\}$$

Conjugate Gradient Method

Proof

- Next we prove $\text{span}\{p_0, p_1, \cdots, p_k, p_{k+1}\} = \text{span}\{r_0, Ar_0, \cdots, A^k r_0, A^{k+1} r_0\}$.
- $$\begin{aligned}\text{span}\{p_0, p_1, \cdots, p_k, p_{k+1}\} &= \text{span}\{p_0, p_1, \cdots, p_k, r_{k+1}\}\\ &= \text{span}\{r_0, Ar_0, \cdots, A^k r_0, r_{k+1}\}\\ &= \text{span}\{r_0, r_1, \cdots, r_k, r_{k+1}\}\\ &= \text{span}\{r_0, Ar_0, \cdots, A^k r_0, A^{k+1} r_0\}\end{aligned}$$

Conjugate Gradient Method

Proof

- Next we prove the conjugacy, i.e., $p_k^T A p_i = 0, \ \forall i = 0, 1, \cdots, k-1$.
- From the algorithm, $p_{k+1} = -r_{k+1} + \beta_{k+1} p_k$. Multiplying by Ap_i, we have

$$p_{k+1}^T A p_i = -r_{k+1}^T A p_i + \beta_{k+1} p_k^T A p_i$$

- When $i = k$, since $\beta_{k+1} = \frac{r_{k+1}^T A p_k}{p_k^T A p_k}$, we have $p_{k+1}^T A p_k = 0$ immediately.
- When $i \leq k-1$, since we have the induction hypothesis, i.e., $p_k^T A p_i = 0, \text{ for } i = 0, \cdots, k-1$. Thus, according to the expanding subspace minimization theorem,

$$r_{k+1}^T p_i = 0, \text{ for } i = 0, \cdots, k.$$

- Since span$\{p_0, p_1, \cdots, p_i\}$ $=$ span$\{r_0, Ar_0, \cdots, A^i r_0\}$, we have

$$Ap_i \in A\text{span}\{r_0, \cdots, A^i r_0\} = \text{span}\{Ar_0, \cdots, A^{k+1} r_0\} \subset \text{span}\{p_0, \cdots, p_{i+1}\}$$

- Hence we have $r_{k+1}^T A p_i = 0$, for $i = 0, \cdots, k-1$.
- Finally, we have $p_{k+1}^T A p_i = -r_{k+1}^T A p_i + \beta_{k+1} p_k^T A p_i = 0 + 0 = 0$.

Conjugate Gradient Method

Proof

- Finally we prove $r_k^T r_i = 0,$ for $i = 0, \cdots, k-1$ by noninductive argument.
- Becasue the direction set is conjugate, we have $r_k^T p_i = 0,$ for $i = 0, \cdots, k-1$ and $k = 1, \cdots n-1$.
- Since $p_i = -r_i + \beta_i p_{i-1}$, so that $r_i \in \text{span}\{p_{i-1}, p_i\}$ for $i = 1, \cdots, k-1$, we conclude that $r_k^T r_i = 0$ for $i = 0, \cdots, k-1$.
- Note that $r_k^T r_0 = -r_k^T p_0 = 0$.
- We have the final argument $r_k^T r_i = 0,$ for $i = 0, \cdots, k-1$.

Conjugate Gradient Method

- The proof of this theorem relies on the fact that the first direction p_0 is the steepest descent direction $-r_0$.
- In fact, the result does not hold for other choices of p_0.
- Since the gradients r_k are mutually orthogonal, the term conjugate gradient method is actually a misnomer.
- It is the search directions, not the gradients, that are conjugate with respect to A.

A Practical Form of the CG Method

- We can derive a slightly more economical form of the conjugate gradient method by using the results of above theorems.
- Since $r_k^T p_k = r_k^T(-r_k + \beta_k p_{k-1}) = -r_k^T r_k$ and $r_{k+1} = r_k + \alpha_k A p_k$, we have

$$\alpha_k = -\frac{r_k^T p_k}{p_k^T A p_k} = \frac{r_k^T r_k}{p_k^T A p_k}, \tag{4.12}$$

$$\beta_{k+1} = \frac{r_{k+1}^T A p_k}{p_k^T A p_k} = \frac{r_{k+1}^T (r_{k+1} - r_k)/\alpha_k}{r_k^T r_k/\alpha_k} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \tag{4.13}$$

- We can have a more practical form of the CG method

A Practical Form of the CG Method

Algorithm 2: (CG)

```
Given $x_0$;
Set $r_0 \leftarrow Ax_0 - b$, $p_0 \leftarrow -r_0$, $k \leftarrow 0$;
while $r_k \neq 0$, do
    $\alpha_k \leftarrow -\frac{r_k^T p_k}{p_k^T A p_k}$;
    $x_{k+1} \leftarrow x_k + \alpha_k p_k$;
    $r_{k+1} \leftarrow r_k + \alpha_k A p_k$,        $(Ax_{k+1} - b)$;
    $\beta_{k+1} \leftarrow \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$,        $(\frac{r_{k+1}^T A p_k}{p_k^T A p_k})$;
    $p_{k+1} \leftarrow -r_{k+1} + \beta_{k+1} p_k$;        $k \leftarrow k+1$;
End(while)
```

Rate of Convergence

Theorem

If A has only r distinct eigenvalues, the the CG iteration will terminate at the solution in at most r iterations.

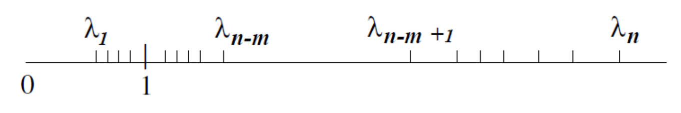
Theorem

If A has eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$, we have that

$$\|x_{k+1} - x^*\|_A^2 \leq \left(\frac{\lambda_{n-k} - \lambda_1}{\lambda_{n-k} + \lambda_1}\right)^2 \|x_0 - x^*\|_A^2. \tag{4.14}$$

Rate of Convergence

- Above theorem can be used to predict the behavior of the CG method on specific problems.
- Suppose we have the situation: the eigenvalues of A consist of m large values, with the remaining $n-m$ smaller eigenvalues clustered around 1.

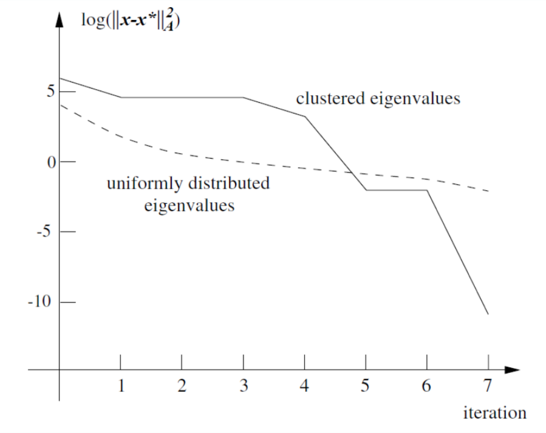


- If we define $\epsilon = \lambda_{n-m} - \lambda_1$, the theorem tells us that after $m+1$ steps of the conjugate gradient algorithm, we have

$$\|x_{m+1} - x^*\| \approx \epsilon \|x_0 - x^*\|_A$$

- For a small value of ϵ, we conclude that the CG iterates will provide a good estimate of the solution after only $m+1$ steps.

Demo



- Performance of the conjugate gradient method on a problem in which five of the eigenvalues are large and the remainder are clustered near 1
- Performance of the conjugate gradient method on a matrix with uniformly distributed eigenvalues.

Rate of Convergence

- Another convergence expression for CG is based on the Euclidean condition number of A, which is defined by

$$\kappa(A) = \|A\|_2 \|A^{-1}\|_2 = \lambda_n / \lambda_1.$$

-

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

Preconditioning

- We can accelerate the conjugate gradient method by transforming the linear system to improve the eigenvalue distribution of A.
- The key to this process, which is known as *preconditioning*, is a change of variables from x to $\hat{x}$ via a nonsingular matrix C, that is,

$$\hat{x} = Cx$$

- The quadratic ϕ is transformed accordingly to

$$\hat{\phi}(\hat{x}) = \frac{1}{2}\hat{x}^T(C^{-T}AC^{-1})^{-1}\hat{x} - (C^{-T}b)^T\hat{x}$$

- If we use CG algorithm to minimize $\hat{\phi}$ or, equivalently, to solve the linear system

$$(C^{-T}AC^{-1})\hat{x} = C^{-T}b$$

  then the convergence rate will depend on the eigenvalues of the matrix $C^{-T}AC^{-1}$ rather than those of A.
- Therefore, we aim to choose C such that the eigenvalues of $C^{-T}AC^{-1}$ are more favorable for the convergence theory discussed above.

Preconditioning

Algorithm 3: (Preconditioned CG)

```
Given $x_0$, preconditioner $M$;
Set $r_0 \leftarrow Ax_0 - b$,
Solve $My_0 = r_0$, $p_0 \leftarrow -y_0$, $k \leftarrow 0$;
while $r_k \neq 0$, do
    $\alpha_k \leftarrow -\frac{r_k^T y_k}{p_k^T A p_k}$;
    $x_{k+1} \leftarrow x_k + \alpha_k p_k$;
    $r_{k+1} \leftarrow r_k + \alpha_k A p_k$;
    Solve $My_{k+1} = r_{k+1}$ ;
    $\beta_{k+1} \leftarrow \frac{r_{k+1}^T y_{k+1}}{r_k^T y_k}$;
    $p_{k+1} \leftarrow -y_{k+1} + \beta_{k+1} p_k$;
    $k \leftarrow k + 1$;
End(while)
```

Preconditioning

Remark

- The above algorithm does not make use of C explicitly, but rather the matrix $M = C^T C$, which is symmetric and positive definite by construction.
- If we set $M = I$ in above algorithm, we recover the standard CG method.
- The orthogonality property of the successive residuals becomes

$$r_k^T M^{-1} r_j = 0, \quad \forall k \neq j.$$

- In terms of computational effort, the main difference between the preconditioned and unpreconditioned CG methods is the need to solve systems of the form $My = r$.

Nonlinear Conjugate Gradient Methods

- In place of the choice for the step length α_k (which minimizes ϕ along the search direction p_k), we need to perform a line search that identifies an approximate minimum of the nonlinear function f along p_k.
- The residual r , which is $\nabla\phi$ in linear CG algorithm, must be replaced by the gradient of the nonlinear objective f .

The Fletcher-Reeves Method

Algorithm 4: (FR: Fletcher-Reeves)

```
Given $x_0$;
Evaluate $f_0 = f(x_0)$, $\nabla f_0 = \nabla f(x_0)$;
Set $p_0 \leftarrow -\nabla f_0$, $k \leftarrow 0$;
while $\nabla f_k \neq 0$, do
    Compute $\alpha_k$;
    $x_{k+1} \leftarrow x_k + \alpha_k p_k$;
    Evaluate $\nabla f_{k+1}$;
    $\beta_{k+1}^{FR} \leftarrow \frac{\nabla f_{k+1}^T \nabla f_{k+1}}{\nabla f_k^T \nabla f_k}$;
    $p_{k+1} \leftarrow -\nabla f_{k+1} + \beta_{k+1}^{FR} p_k$;
    $k \leftarrow k + 1$;
End(while)
```

The Fletcher-Reeves Method

- If the line search is exact or any inexact line search procedure that yields an α_k satisfying the following strong wolfe conditions

$$f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k (\nabla f(x_k))^T p_k \tag{4.15a}$$
$$|(\nabla f(x_k + \alpha_k p_k))^T p_k| \leq c_2 |\nabla f_k^T p_k|, \tag{4.15b}$$

  with $0 < c_1 < c_2 < \frac{1}{2}$ will ensure that all directions p_k are descent directions for the function f.

- Goldstein conditions are not suitable for CG method.

Behavior of the Fletcher-Reeves Method

Theorem

- Suppose that FR algorithm is implemented with a step length α_k that satisfies the strong Wolfe conditions (4.15) with $0 < c_2 < \frac{1}{2}$.
- Then the method generates descent directions p_k that satisfy the following inequalities:

$$-\frac{1}{1-c_2} \leq \frac{\nabla f_k^T p_k}{\|\nabla f_k\|^2} \leq \frac{2c_2-1}{1-c_2}, \quad \forall k = 0, 1, \cdots$$

- This theorem can be used to explain a *weakness* of the Fletcher-Reeves method.
- We will argue that if the method generates a *bad direction* and a tiny step, then the next direction and next step are also likely to be*poor*.

Behavior of the Fletcher-Reeves Method

- Let θ denote the angle between p_k and $-\nabla f_k$, defined by

$$\cos\theta_k = \frac{-\nabla f_k^T p_k}{\|\nabla f_k\| \|p_k\|}$$

- Suppose that p_k is a poor search direction, in the sense that it makes an angle of nearly $\frac{\pi}{2}$, that is, $\cos\theta_k \approx 0$.
- By multiplying both sides of the relationship in the above theorem by $\|\nabla f_k\|/\|p_k\|$, we obtain

$$\frac{1-2c_2}{1-c_2}\frac{\|\nabla f_k\|}{\|p_k\|} \le \cos\theta_k \le \frac{1}{1-c_2}\frac{\|\nabla f_k\|}{\|p_k\|}, \quad \forall k = 0, 1, \cdots .$$

- From these inequalities, we deduce that $\cos\theta_k \approx 0$ if and only if

$$\|\nabla f_k\| \ll \|p_k\|.$$

Behavior of the Fletcher-Reeves Method

- Since p_k is almost orthogonal to the gradient, it is likely that the step from x_k to x_{k+1} is tiny, that is, $x_{k+1} \approx x_k$.
- If so, we have $\nabla f_{k+1} \approx \nabla f_k$, and therefore
$$\beta_{k+1}^{FR} \approx 1$$
- By using this approximation together with $\|\nabla f_{k+1} \approx \|\nabla f_k\| \ll \|p_k\|$, we conclude that
$$p_{k+1} \approx p_k$$
- Therefore the new search direction will improve little (if at all) on the previous one.
- It follows that if the condition $\cos\theta_k \approx 0$ holds at some iteration k and if the subsequent step is small, a long sequence of unproductive iterates will follow.

The Polak-Ribière Method

- There are many variants of the Fletcher-Reeves method that differ from each other mainly in the choice of the parameter β_k.
- An important variant, proposed by Polak and Ribière, defines this parameter as follows:

$$\beta_{k+1}^{PR} = \frac{\nabla f_{k+1}^T(\nabla f_{k+1} - \nabla f_k)}{\|\nabla f_k\|^2} \tag{4.16}$$

- We refer the algorithm in which β_{k+1}^{PR} replaces β_{k+1}^{FR} in Algorithm 4 as Algorithm PR.
- Algorithm FR and Algorithm PR are identical when f is a strongly convex quadratic function and the line search is exact, since the gradients are mutually orthogonal, i.e., $(\nabla f_{k+1})^T \nabla f_k = 0$, and so $\beta_{k+1}^{PR} = \beta_{k+1}^{FR}$.
- When applied to general nonlinear functions with inexact line searches, however, the behavior of the two algorithms differs markedly.
- Numerical experience indicates that Algorithm PR-CG tends to be the more robust and efficient of the two.

Behavior of the Polak-Ribière Method

- If the search direction p_k satisfies $\cos\theta_k \approx 0$ for some k, and if the subsequent step is small, it follows by substituting $\nabla f_k \approx \nabla f_{k+1}$ into the PR formula, we get
$$\beta_{k+1}^{PR} \approx 0.$$
- So the new search direction p_{k+1} will be close to the steepest descent direction ∇f_{k+1}, and $\cos\theta_{k+1}$ will be close to 1.
- Therefore, Algorithm PR-CG essentially performs a restart after it encounters a bad direction.

The Polak-Ribière Method's Variant

- For Algorithm PR-CG, the strong Wolfe conditions (4.15) do not guarantee that p_k is always a descent direction.

Theorem

- Consider the Polak-Ribière method (4.16) with an ideal line search.
- There exists a twice continuously differential objective function $f : \mathcal{R}^3 \to \mathcal{R}$ and a starting point $x_0 \in \mathcal{R}^3$
- such that the sequence of gradients $\{\nabla f(x_k)\}$ is bounded away from zero.

Remark

- If we define the β parameter as

$$\beta_{k+1}^{+} = \max\{\beta_{k+1}^{PR}, 0\} \tag{4.17}$$

  which gives an algorithm called Algorithm PR+,
- then a simple adaptation of the strong Wolfe conditions ensures that the descent property holds.

The FR-PR Formula

- It is possible to guarantee global convergence for any parameter β_k satisfying the bound

$$|\beta_k| \leq \beta_k^{FR}$$

  for all $k \geq 2$.

- This fact suggest the following modification of the PR method. For all $k \geq 2$ let

$$\beta_k = \begin{cases} -\beta_k^{FR}, & \text{if } \beta_k^{PR} \leq \beta_k^{FR}; \\ \beta_k^{PR}, & \text{if } |\beta_k^{PR}| \leq \beta_k^{FR}; \\ \beta_k^{FR}, & \text{if } \beta_k^{PR} > \beta_k^{FR}. \end{cases}$$

- The algorithm based on this strategies will be denoted by FR-PR.

The Hestenes-Stiefel Formula

- When the objective is quadratic and the line search is exact, There are many other choices for β_{k+1} that coincide with the Fletcher-Reeves formula β_{k+1}.
- The Hestenes-Stiefel formula, which defines

$$\beta_{k+1}^{HS} = \frac{(\nabla f_{k+1})^T(\nabla f(x_{k+1}) - \nabla f(x_k))}{(\nabla f(x_{k+1}) - \nabla f(x_k))^T p_k} \tag{4.18}$$

- This algorithm (called Algorithm HS) is similar to Algorithm PR, both in terms of its theoretical convergence properties and in its practical performance.

Dai-Yuan Formula

- Other variants of the CG method have recently been proposed.
-
$$\beta_{DY}^{k+1} = \frac{\nabla f_{k+1}^T \nabla f_{k+1}}{(\nabla f_{k+1} - \nabla f_k)^T p_k}, \tag{4.19}$$
- This algorithm possess attractive theoretical and computational properties.
    - This choice guarantee that p_k is a descent direction, provided the step length α_k satisfies the Wolfe conditions.
    - The CG algorithm based on (4.19) appear to be competitive with the PR method.

Quadratic Termination and Restarts

- A modification that is often used in nonlinear conjugate gradient procedures is to *restart the iteration* at every n steps by setting $\beta_k = 0$, that is, by taking a steepest descent step.
- Restarting serves to periodically refresh the algorithm, erasing old information that may not be beneficial.
- We can even prove a strong theoretical result about restarting: It leads to n-step quadratic convergence, that is,

$$\|x_{k+n} - x^*\| = \mathcal{O}(\|x_k - x^*\|^2) \tag{4.20}$$

- It may not be relevant in a practical context, because nonlinear conjugate gradient methods can be recommended only for solving problems with large n.
- In such problems restarts may never occur, since an approximate solution is often located in fewer than n steps.

Quadratic Termination and Restarts

- Since the gradients are mutually orthogonal when f is a quadratic function.
- A restart is performed whenever two consecutive gradients are far from orthogonal, as measured by the test

$$\frac{|\nabla f_k^T \nabla f_{k-1}|}{\|\nabla f_k\|^2} \geq \nu \tag{4.21}$$

- A typical value for the parameter $\nu = 0.1$.

Global Convergence

Assumptions

1. The level set $\mathcal{L} := \{x | f(x) \leq f(x_0)\}$ is bounded.
2. In some open neighborhood $\mathcal{N}$ of $\mathcal{L}$, the objective function f is Lipschitz continuously differentiable.

These assumption imply that there is a constant $\bar{\gamma}$ such that

$$\|\nabla f(x)\| \leq \bar{\gamma}, \quad \forall x \in \mathcal{L}. \tag{4.22}$$

Theorem(AI-Baali)

- Suppose that Assumptions holds, and that Algorithm 4 is implemented with a line search that satisfies the strong Wolfe conditions (4.15) with $0 < c_1 < c_2 < \frac{1}{2}$.
- Then

$$\lim_{k\to\infty} \inf \|\nabla f_k\| = 0. \tag{4.23}$$

Thanks For Your Attention