10-725: Optimization **Fall 2012**

Lecture 10: September 26

Lecturer: Barnabas Poczos/Ryan Tibshirani *Scribes: Yipei Wang, Zhiguang Huo*

This lecture's notes illustrate some uses of various LaTeX macros. Take a look at this and imitate.

10.1 Motivation

From previous lectures, we know that steepest decent method is very slow. The Newton method is fast. But the computation of inverse of the Hessian matrix is very expensive. Can we find a method between these two?

Conjugate direction method can be regarded as being between the method of steepest decent and Newton's method.

10.2 Conjugate Direction Methods

The goal includes two folds. (1)Accelarate the convergence rate of steepest decent. (2)Avoiding the high computational cost Newton's method

10.2.1 Definition[Q-conjugate directions]

Let Q be a symmetric matrix
$d_1, d_2, ..., d_k$ vectors ($d_i \in R^n, d_i \neq 0$) are Q-orthogonal (conjugate) w.e.r Q, if

$$d_i^T Q d_j = 0,\ \forall i \neq j$$

note In the application we consider, the matrix Q will be positive definite. But this is not inherent in the basc definition.

If Q=0, any two vectors are conjugate.
If Q=I, conjugacy is equivalent to the usual notion of orthogonality.

10.2.2 Linear Independency Lemma

lemma Let Q be positive definite.
If $d_1, d_2, ..., d_k$ vectors are Q-conjugate, then they are linearl independent.

proof by contradiction If $d_k = \alpha_1 d_1 + ... + \alpha_{k-1}$, then

$$0 < d_k^T Q d_k = d_k^T Q(\alpha_1 d_1 + ... + \alpha_{k-1} d_{k-1}) = \alpha_1 d_k^T Q d_1 + ... \alpha_{k-1} d_k^T Q d_{k-1} = 0$$

10.2.3 The importance of Q-conjugancy

For quadratic problem, our goal is to solve

$$\arg\min_{x \in R^n} \frac{1}{2} x^T Q x - b^T x$$

Assume Q is positive definite
The first order differiential of the objective function equals to 0 for the minimizer.
So the unique solution to this problem is the solution to

$$Qx = b, x \in R^n$$

Let x^* denote the solution. Let $d_0, d_1, ..., d_{n-1}$ vectors be Q-conjugate. Since $d_0, d_1, ..., d_{n-1}$ vectors are independent,

$$x^* = \alpha_0 d_0 + ... + \alpha_{n-1} d_{n-1}$$

Therefore, $d_i^T Q x^* = d_i^T Q(\alpha_0 d_0 + ... + \alpha_{n-1} d_{n-1}) = \alpha_i d_i^T Q d_i$

$$\Rightarrow \alpha_i = \frac{d_i^T Q x^*}{d_i^T Q d_i} = \frac{d_i^T b}{d_i^T Q d_i}$$

we don't need to know x^* to get α_i

$$x^* = \sum_{i=0}^{d-1} \alpha_i d_i = \sum_{i=0}^{d-1} \frac{d_i^T b}{d_i^T Q d_i} d_i$$

We can see that there is no need to do matrix inversion. We only need to calculate inner product.

10.2.4 Conjugate Direction Theorem

The expansion for x^* can be considered to be the result of an iterative process of n steps where at the ith steps $\alpha_i d_i$ is added. This can be generalized further such a way that the starting point of the iteration can be arbitery x_0

We introduce the conjugate direction theorem here.

Theorem[Conjugate Direction Theorem]

Let $d_0, d_1, ..., d_{n-1}$ vectors be Q-conjugate.
$x_0 \in R^n$ be an arbitery starting point.

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

$$g_k = Qx_k - b$$

$$\alpha_k = -\frac{-g_k^T d_k}{d_k^T Q d_k} = -\frac{(Qx_k - b)^T d_k}{d_k^T Q d_k}$$

Then after n steps, $x_n = x^*$

Proof
Since $\{d_0, d_1, ..., d_{n-1}\}$ vectors are independent.
$\Rightarrow x^* - x_0 = \alpha_0 d_0 + ... + \alpha_{n-1} d_{n-1}$ for some $\alpha_0, ..., \alpha_{n-1}$

Using the $x_{k+1} = x_k + \alpha_k d_k$ update rules, we have

$$\begin{aligned} x_1 &= x_0 + \alpha_0 d_0 \\ x_2 &= x_0 + \alpha_0 d_0 + \alpha_1 \\ x_k &= x_0 + \alpha_0 d_0 + \alpha_1 + ...\alpha_{k-1} d_{k-1} \\ x_n &= x_0 + \alpha_0 d_0 + \alpha_1 d_1 + ... + \alpha_{n-1} d_{n-1} = x^* \end{aligned} \qquad \text{span}$$

Therefore, it's enough to prove that with these α_k values we have

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

We already know

$$\begin{aligned} x^* - x_0 &= \alpha_0 d_0 + ... + \alpha_{n-1} d_{n-1} \\ x_k - x_0 &= \alpha_0 d_0 + ... + \alpha_{k-1} d_{k-1} \end{aligned} \qquad \text{span}$$

Therefore,

$$d_k^T Q(x^* - x_0) = d_k^T Q(\alpha_0 d_0 + ... + \alpha_{n-1} d_{n-1} = \alpha_k d_k^T Q d_k$$

$$\Rightarrow \alpha_k = \frac{d_k^T Q(x^* - x_0)}{d_k^T Q d_k}$$

$$d_k^T Q(x_k - x_0) = d_k^T Q(\alpha_0 d_0 + \alpha_1 d_1 + ... + \alpha_{k-1} d_{k-1}) = 0$$

$$d_k Q(x^* - x_0) = d_k^T Q(x^* - x_k + x_k - x_0) = d_k^T Q(x^* - x_k)$$

$$\alpha_k = \frac{d_k^T Q(x^* - x_0)}{d_k^T Q d_k} = \frac{d_k^T Q(x^* - x_k)}{d_k^T Q d_k} = -\frac{d_k^T g_k}{d_k^T Q d_k}$$

Another motivation for Q-conjugacy
Our goal is:

$$\arg\min_{x\in R^n} \frac{1}{2}x^TQx - b^Tx$$

$x - x_0 = \alpha_0 d_0 + ... + \alpha_{n-1}d_{n-1}$ for some $\{\alpha_i\}_{i=0}^{n-1} \in R$ Therefore,

$$f(x) = \frac{1}{2}[x_0+\sum_{j=0}^{n-1}\alpha_j d_j]^TQ[x_0+\sum_{j=0}^{n-1}\alpha_j d_j]-b^T[x_0+\sum_{j=0}^{n-1}\alpha_j d_j]f(x) = c+\sum_{j=0}^{n-1}\frac{1}{2}[x_0+\alpha_j d_j]^TQ[x_0+\alpha_j d_j]-b^T[x_0+\alpha_j d_j]$$

The optimization problem is transformed into a n separate 1-dimensional optimization problems

10.2.5 Expanding Subspace Theorem

Let $B_k = span(d_0, ..., d_{k-1}) \subset R^n$

We will show as the method of conjugate directions progresses each x_k minimizes the objective $f(x) = \frac{1}{2}x^TQx - b^Tx$ both over $x_0 + B_k$ and $x_{k-1} + \alpha d_{k-1}, \alpha \in R$.That is

$$x_k = \arg\min_{x=x_{k-1}+\alpha d_{k-1},\alpha\in R} \frac{1}{2}x^TQx - b^Txx_k = \arg\min_{x=x_0+B_k} \frac{1}{2}x^TQx - b^Tx$$

Theorem[Expanding Subspace Theorem]
Let $\{d_i\}_{i=0}^{n-1}$ be a sequence of Q-conjugate vectors in R^n, $x_0 \in R^n$ arbitary

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

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

$$\Rightarrow x_k = \arg\min_{x=x_{k-1}+\alpha d_{k-1},\alpha\in R} \frac{1}{2}x^TQx - b^Tx$$

$$x_k = \arg\min_{x=x_0+B_k} \frac{1}{2}x^TQx - b^Tx$$

Proof
It's enough to show that x_k minimizes f on $x = x_0 + B_k$ since it contains the line:$x = x_{k-1} + \alpha d_{k-1}$ (by the definition of B_k)

Since f is strictly convex, it's enough to show that $g_k = f'(x_k)$ is orthogonal to B_k

We prove $g_k \perp B_k$ by induction
k=0: B_0 is empty set.
Assume that $g_k \perp B_k$ and prove that $g_{k+1} \perp B_{k+1}$
By definition,

$$x_{k+1} = x_k + \alpha_k d_k g_k = Qx_k - b$$

Therefore,

$$
\begin{aligned}
g_{k+1} &= Qx_{k+1} - b \\
&= Q(x_k + \alpha_k d_k) - b \\
&= (Qx_k - b) + \alpha_k Q d_k \\
&= g_k + \alpha_k Q d_k
\end{aligned}
$$

span

First, let us prove that $g_{k+1} \perp d_k$ We have proved $g_{k+1} = g_k + \alpha_k Q d_k$
By definition,

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

Therefore,

$$d_k^T g_{k+1} = d_k^T (g_k + \alpha_k Q d_k) = d_k^T - d_k^T g_k = 0$$

$$g_{k+1} \perp d_k$$

Now let us prove that $g_{k+1} \perp d_i, i < k$
Since

$$g_{k+1} = g_k + \alpha_k Q d_k$$

$$g_k \perp B_k = span(d_0, ..., d_{k-1})$$

Therefore,

$$d_i^T g_{k+1} = d_i^T (g_k + \alpha_k Q d_k) = d_i^T g_k + \alpha_k d_i^T Q d_k = 0$$

$$g_{k+1} \perp d_i, \forall i < k$$

So far we have proved that $g_{k+1} \perp B_{k+1}$, where $B_{k+1} = span(d_0, ..., d_k)$

Corollary of Exponential subspace theorem
corollary

$$
\begin{aligned}
g_k \perp B_k &= span(d_0, ..., d_{k-1}) \\
g_k^T d_i &= 0, \forall 0 \le i \le k \\
\emptyset &= B_0 \subset ... \subset B_k \subset B_n = R^n
\end{aligned}
$$

span

since x_k minimizes f over $x_0 + B_k$. $\Rightarrow x_n$ is the minimum of f in R^n

10.3 Conjugate gradient method

The conjugate gradient method isa conjugate direction method

Selects the successive direction vectors as a conjugate version of the successive gradients obtained as the method progress

The conjugate directions are not specified beforehandm but rather are determined sequentially at each step of the iteration

10.3.1 Description and advantage

So far given $d_0, \dots, d_{n-1}$, we already have an update rule for α_k.

$$\alpha_k = -\frac{g_K^T d_k}{d_K^T Q d_k}$$

The problem is how should we choose vectors $d_0, \dots, d_{n-1}$.

The conjugate gradient method is the conjugate direction method that is obtained by selecting the successive direction vectors as a conjugate version of the successive gradients obtained as the method progresses. Thus, the directions are not specified beforehand, but rather are determined sequentially at each step of the iteration.

There are three primary advantages to this method of direction selection.

1. Unless the solution is attained in less than n steps, the gradient is always nonzero and linearly independent of all previous direction vectors.
2. A more important advantage of the conjugate gradient method is the especially simple formula that is used to determine the new direction vector. This simplicity makes the method only slightly more complicated than steepest descent.
3. Because the directions are based on the gradients, the process makes good uniform progress toward the solution at every step. This is in contrast to the situation for arbitrary sequences of conjugate directions in which progress may be slight until the final few steps.

10.3.2 Algorithm

Let $x_0 \in \mathbb{R}^n$ be arbitrary.

$$d_0 = -g_0 = b - Qx_0$$

$$\alpha_k = -\frac{g^T d_k}{d_k^T Q d_k}$$

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

$$g_k = Qx_k - b$$

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

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

In the algorithm the first step is identical to a steepest descent step; each succeeding step moves in a direction that is a linear combination of the current gradient and the preceding direction vector. The attractive feature of the algorithm is the simple formula for updating in the direction vector. The method is only slightly more complicated to implement than the method of steepest descent but converges in a finite number of steps.

Theorem 10.1 *The conjugate gradient algorithm is a conjugate direction method. If it does not terminate at x_k, then*

a) $span(g_0, g_1, \dots, g_k) = span(g_0, Qg_0, \dots, Q^k g_0)$

b) $span(d_0, d_1, \dots, d_k) = span(g_0, Qg_0, \dots, Q^k g_0)$

c) $d_k^T Q d_i = 0, \forall i < k$

d) $\alpha_k = \frac{g_k^T g_k}{d_k^T Q d_k}$

e) $\beta_k = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k}$

Proof: We first prove a), b) and c) simultaneously by induction. Clearly, they are true for $k = 0$. Now suppose they are true for k, we show that they are true for $k + 1$. We have

$$g_{k+1} = g_k + \alpha_k Q d_k$$

By the induction hypothesis both g_k and Qd_k belong to $\text{span}(g_0, Qg_0, \dots, Q^{k+1} g_0)$, the first by a) and the second by b). Thus $g_{k+1} \in \text{span}(g_0, Qg_0, \dots, Q^{k+1} g_0)$. Furthermore $g_{k+1} \notin \text{span}(d_0, d_1, \dots, d_k) = \text{span}(g_0, Qg_0, \dots, Q^k g_0)$ since otherwise $g_{k+1} = 0$, because for any conjugate direction method g_{k+1} is orthogonal to $\text{span}(d_0, d_1, \dots, d_k)$]. (The induction hypothesis on c) guarantees that the method is a conjugate direction method up to x_{k+1}.) Thus, finally we conclude that

$$\text{span}(g_0, g_1, \dots, g_k) = \text{span}(g_0, Qg_0, \dots, Q^k g_0)$$

which proves a).

To prove b) we write

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

and b) immediately follows form a) and the induction hypothesis on b).

Next, to prove c we have

$$d_{k+1}^T Q d_i = -g_{k+1}^T Q d_i + \beta_k d_k^T Q d_i.$$

For $i = k$ the right side is zero by definition of β_k. For $i < k$ both terms vanish. The first term vanishes since $Qd_i \in \text{span}(d_0, d_1, \dots, d_{k+1})$, the induction hypothesis which guarantees the method is a conjugate direction method up to x_{k+1}, and by the Expanding Subspace Theorem that guarantees that g_{k+1} is orthogonal to $\text{span}(d_0, d_1, \dots, d_{k+1})$. The second term vanishes by the induction hypothesis on c). This proves c), which also proves that the method is a conjugate direction method.

To prove d) we have

$$-g_k^T d_k = g_k^T g_k - \beta_{k-1} g_k^T d_{k-1},$$

and the second term is zero by the Expanding Subspace Theorem.

Finally, to prove e) we note that $g_{k+1}^T g_k = 0$, because $g_k \in \text{span}(d_0, d_1, \ldots, d_k)$ and $g_{k+1} \perp \text{span}(d_0, d_1, \ldots, d_k)$. Thus since

$$Qd_k = \frac{1}{\alpha_k}(g_{k+1} - g_k).$$

We have

$$g_{k+1}^T Q d_k = \frac{1}{\alpha_k} g_{k+1}^T g_{k+1}.$$

■

10.4 Extension to non-quadratic problem

10.4.1 Description

Our goal is to

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

We will make quadratic approximation $g_k = \nabla f(x_k)$ and $Q = \nabla^2 f(x_k)$. using these associations, reevaluated at each step, all quantities necessary to implement the basic conjugate gradient algorithm can be evaluated. If f is quadratic, these associations are identities, so that the general algorithm obtained by using them is a generalization of the conjugate gradient scheme. This is similar to the philosophy underlying Newtons method where at each step the solution of a general problem is approximated by the solution of a purely quadratic problem through these same associations.

When applied to nonquadratic problems, conjugate gradient methods will not usually terminate within n steps. It is possible therefore simply to continue finding new directions according to the algorithm and terminate only when some termination criterion is met. Therefore after n steps, we can restart the process from this point and run the algorithm for another n steps.

10.4.2 Algorithm

1. Starting at x_0 and compute $g_0 = \nabla f(x_0)$ and set $d_0 = -g_0$.
2. For $k = 0, 1, \ldots, n-1$:
   a) Set $x_{k+1} = \alpha_k d_k$ where $\alpha_k = -\frac{g_k^T d_k}{d_k^T [\nabla^2 f(x_k)] d_k}$
   b) Compute $g_{k+1} = \nabla f(x_{k+1})$
   c) Unless $k = n-1$, set $d_{k+1} = -g_{k+1} + \beta_k d_k$ where $\beta_k = \frac{g_{k+1}^T [\nabla f(x_k)] d_k}{d_k^T [\nabla^2 f(x_k)] d_k}$
3. Replace x_0 by x_n and go back to step 1. Until converge.

10.4.3 Properties of CGA

An attractive feature of the algorithm is that, just as in the pure form of Newtons method, no line searching is required at any stage. Also, the algorithm converges in a finite number of steps for a quadratic problem. The undesirable features are that $F(x_k)$ must be evaluated at each point, which is often impractical, and that the algorithm is not, in this form, globally convergent.

10.5 Line search method

Two of the line search method are FletcherReeves method and Ploak-Ribiere method. This algorithm is quite similar to the Conjugate gradient method. The slight differences are in step 2 a) and 2 c) above.

10.5.1 Algorithm for Fletcher Reeves method

1. Starting at x_0 and compute $g_0 = \nabla f(x_0)$ and set $d_0 = -g_0$.
2. For $k = 0, 1, \ldots, n-1$:
   a) Set $x_{k+1} = \alpha_k d_k$ where $\alpha_k = \arg\min_\alpha f(x_k + \alpha d_k)$
   b) Compute $g_{k+1} = \nabla f(x_{k+1})$
   c) Unless $k = n-1$, set $d_{k+1} = -g_{k+1} + \beta_k d_k$ where $\beta_k = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k}$
3. Replace x_0 by x_n and go back to step 1. Until converge.

Firstly Fletcher-Reeves method is line search method when evaluating α and there is no close form solution normally. Secondly the Hessian is not used in the algorithm. Also in the quadratic case it is identical to the original conjugate direction algorithm.

10.5.2 Algorithm for Polak Ribiere method

Polak-Ribiere method is quite similar to Fletcher-Reeves method. The only difference occurs when evaluating β_k.

1. Starting at x_0 and compute $g_0 = \nabla f(x_0)$ and set $d_0 = -g_0$.
2. For $k = 0, 1, \ldots, n-1$:
   a) Set $x_{k+1} = \alpha_k d_k$ where $\alpha_k = \arg\min_\alpha f(x_k + \alpha d_k)$
   b) Compute $g_{k+1} = \nabla f(x_{k+1})$
   c) Unless $k = n-1$, set $d_{k+1} = -g_{k+1} + \beta_k d_k$ where $\beta_k = \frac{(g_{k+1} - g_k)^T g_{k+1}}{g_k^T g_k}$
3. Replace x_0 by x_n and go back to step 1. Until converge.

Again this leads to a value identical to the standard formula in the quadratic case. Experimental evidence seems to favor the PolakRibiere method over other methods of this general type.

10.6 Convergence

Global convergence of the line search methods is established by noting that a pure steepest descent step is taken every n steps and serves as a spacer step. Since the other steps do not increase the objective, and in fact hopefully they decrease it, global convergence is assured. Thus the restarting aspect of the algorithm is important for global convergence analysis, since in general one cannot guarantee that the directions d_k generated by the method are descent directions.

The local convergence properties of both of the above, and most other, nonquadratic extensions of the conjugate gradient method can be inferred from the quadratic analysis. Since one complete cycle solves a quadratic problem exactly just as Newtons method does in one step, we expect that for general nonquadratic problems there will hold $\|x_{k+n} - x^*\| \leq c\|x_k - x^*\|^2$ for some c and $k = 0, n, 2n, 3n, \ldots$. This can indeed be proved, and of course underlies the original motivation for the method. For problems with large n, however, a result of this type is in itself of little comfort, since we probably hope to terminate in fewer than n steps.

10.7 summary

- Conjugate Direction Methods.
  - conjugate directions
- Minimizing quadratic functions
- Conjugate Gradient Methods for nonquadratic functions
  - Line search methods

    1. FletcherReeves method
    2. PolakRibiere method

References

[CW87] Luenberger, David G and Ye, Yinyu, "Linear and nonlinear programming," *Springer*, 2008, vol. 116.