Non-linear conjugate gradient method(s):
Fletcher–Reeves
Polak–Ribièrè
Hestenes–Stiefel

January 29, 2014
Extend the linear CG method to non-quadratic functions $f : \mathbb{R}^n \to \mathbb{R}$:

$$\text{minimize } f(x).$$

**Reminder:** CG for quadratic functions

1. $r_0 = Ax_0 - b$, $p_0 = -r_0$, $k = 0$
2. **while** $\|r_k\| > \epsilon$ **do**
3. $\alpha_k = -(r_k^T p_k)/(p_k^T A p_k)$
4. $x_{k+1} = x_k + \alpha_k p_k$
5. $r_{k+1} = Ax_{k+1} - b = r_k + \alpha_k A p_k$
6. $\beta_{k+1} = (r_{k+1}^T A p_k)/(p_k^T A p_k)$
7. $p_{k+1} = -r_{k+1} + \beta_{k+1} p_k$
8. $k = k + 1$
9. **end while**
Problem

Extend the linear CG method to non-quadratic functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$:

$$\text{minimize} \quad f(x) \quad \text{subject to} \quad x \in \mathbb{R}^n.$$ 

Almost done:

1. $r_0 = \nabla f(x_0)$, $p_0 = -r_0$, $k = 0$
2. while $\|r_k\| > \epsilon$ do
3. $\alpha_k =$ linesearch for $f$ along $p_k$
4. $x_{k+1} = x_k + \alpha_k p_k$
5. $r_{k+1} = \nabla f(x_{k+1})$
6. $\beta_{k+1} =$ ???
7. $p_{k+1} = -r_{k+1} + \beta_{k+1} p_k$
8. $k = k + 1$
9. end while
Algebraic manipulations with linear CG:

\[ p_k = -r_k + \beta_k p_{k-1} \]

\[ \Downarrow \]

\[ -r_k^T p_k = -r_k^T [-r_k + \beta_k p_{k-1}] = r_k^T r_k - \beta_k \underbrace{r_k^T p_{k-1}}_{=0} = r_k^T r_k \]

\[ \Downarrow \]

\[ \alpha_k = -\frac{r_k^T p_k}{p_k^T A p_k} = \frac{r_k^T r_k}{p_k^T A p_k} \]
Algebraic manipulations with linear CG:

\[ r_{k+1} = r_k + \alpha_k A p_k \]
\[ \Downarrow \]
\[ A p_k = \alpha_k^{-1} [r_{k+1} - r_k] \]
\[ \Downarrow \]
\[ \beta_{k+1} = \frac{r_{k+1}^T A p_k}{p_k^T A p_k} = \alpha_k^{-1} \frac{r_{k+1}^T [r_{k+1} - r_k]}{p_k^T A p_k} \]
\[ = \frac{p_k^T A p_k}{r_k^T r_k} \frac{r_{k+1}^T [r_{k+1} - r_k]}{p_k^T A p_k} \]
\[ = \frac{r_{k+1}^T [r_{k+1} - r_k]}{r_k^T r_k} \]

leads to Polak–Ribiére

\[ \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \]

leads to Fletcher–Reeves
Fletcher–Reeves algorithm:

1: \( p_0 = -\nabla f(x_0), \ k = 0 \)
2: \( \textbf{while} \ |\nabla f(x_k)| > \epsilon \ \textbf{do} \)
3: \( \alpha_k = \text{linesearch for } f \text{ along } p_k \)
4: \( x_{k+1} = x_k + \alpha_k p_k \)
5: \( \beta_{k+1}^{\text{FR}} = |\nabla f(x_{k+1})|^2 / |\nabla f(x_k)|^2 \)
6: \( p_{k+1} = -\nabla f(x_{k+1}) + \beta_{k+1}^{\text{FR}} p_k \)
7: \( k = k + 1 \)
8: \( \textbf{end while} \)

Big question:

Under which conditions on \( f \), \( \alpha_{k-1} \) is \( p_k \) a descent direction at \( x_k \) for \( f \)?
Fletcher–Reeves algorithm:

Big question:
Under which conditions on \( f \), \( \alpha_{k-1} \) is \( p_k \) a descent direction at \( x_k \) for \( f \)?

\[
\nabla f(x_k)^T p_k = -\|\nabla f(x_k)\|^2 + \beta_k^{\text{FR}} \underbrace{\nabla f(x_k)^T p_{k-1}}_{=0 \text{ for exact linesearch}}
\]

Conclusion: linesearch should be “accurate enough” to guarantee descent.
Analysis of Fletcher–Reeves:

Lemma 5.6
FR CG algorithm with strong Wolfe linesearch; $0 < c_1 < c_2 < 1/2$. Then

$$-\frac{1}{1-c_2} \leq \frac{p_k^T \nabla f(x_k)}{\|\nabla f(x_k)\|^2} \leq \frac{2c_2 - 1}{1-c_2} < 0.$$  

Proof: Induction in $k$.  

Global convergence of Fletcher–Reeves:

**Theorem 5.7**

Assume:
1. \( f \) is bounded from below and is Lipschitz continuously differentiable (prerequisites for Zoutendijk’s);
2. \( \alpha_k \) satisfies strong Wolfe’s, \( 0 < c_1 < c_2 < 1/2 \).

Then \( \liminf_{k \to \infty} \|\nabla f(x_k)\| = 0 \).
Main problem of Fletcher–Reeves:

Suppose

$$\cos \theta_k = -\frac{p_k^T \nabla f(x_k)}{\|p_k\| \|\nabla f(x_k)\|} \approx 0$$

Then

$$\cos \theta_{k+1} \approx 0$$

If we (for any reason) end up with a bad direction $p_k$ then FR continues to generate poor directions.
Polak–Ribiére algorithm:

\[ \beta_{k+1}^{PR} = \frac{\nabla f(x_{k+1})^T [\nabla f(x_{k+1}) - \nabla f(x_k)]}{\| \nabla f(x_k) \|^2} \]

**Warning:**

Strong Wolfe conditions do not guarantee that \( p_k \) - descent direction [but the algorithm often works better than FR in practice]!

\[ \cos \theta_k \approx 0 \implies \cos \theta_{k+1} \approx 1 \]

Even better:

\[ \beta_{k+1}^{PR+} = \max\{0, \beta_{k+1}^{PR}\} \]
Polak–Ribiére algorithm:

\[ \beta_{k+1}^{\text{PR}} = \frac{\nabla f(x_{k+1})^T [\nabla f(x_{k+1}) - \nabla f(x_k)]}{\|\nabla f(x_k)\|^2} \]

**Theorem**

*Even with exact line-search* in the Polak–Ribiére CG algorithm, there are examples \( f : \mathbb{R}^3 \to \mathbb{R} \), \( f \in C^2 \): for some \( x_0 \in \mathbb{R}^3 \) it holds that

\[
\lim \inf_{k \to \infty} \|nabla f(x_k)\| > 0.
\]
Other choices of $\beta$:

$$\hat{y}_k = \nabla f(x_{k+1}) - \nabla f(x_k)$$

- Hestenes–Stiefel:
  $$\beta_{k+1} = \frac{\hat{y}_k^T \nabla f(x_{k+1})}{\hat{y}_k^T p_k}$$

- Dai–Yuan, 1999 [works with strong Wolfe]:
  $$\beta_{k+1} = \frac{\|\nabla f(x_{k+1})\|^2}{\hat{y}_k^T p_k}$$

- Hager–Zhang, 2005 [works with strong Wolfe]:
  $$\beta_{k+1} = \frac{\hat{y}_k \nabla f(x_{k+1})}{\hat{y}_k^T p_k} - 2 \frac{\|\hat{y}_k\|^2 p_k^T \nabla f(x_{k+1})}{(\hat{y}_k^T p_k)^2}$$
Convergence rate:

Linear!
Though one can show faster convergence rate with restarts after every $N$ iterations – in practice we want to stop the algorithm long before $N$ iterations!