Minimization of a function.

In many cases we are interested in finding the minimum of a function \( f(x) \) where \( x \) is a vector of coordinates. We shall motivate the study of this problem by considering the linear problem:

\[
Ax = b
\]

where \( A \) is a matrix and \( b \) is a vector. This is widely known and used task. A formal solution to this problem can be written as \( x = A^{-1}b \). The inversion of the matrix \( A \) is required. Inversion of a matrix can be an expensive operation if the size of the matrix is large and is (in general) proportional to \( N^3 \) where \( N \) is the dimensionality of the matrix. Clearly, for large matrices the computations become formidable.

It is possible to rewrite the above linear problem as a minimization problem in which we seek the minimum of the function

\[
f(x) = (Ax - b)^T (Ax - b)
\]

a minimum of \( f(x) \) is when \( Ax - b = 0 \), which is what we are looking for. The gradient of \( f(x) \) - \( \nabla f(x) \) is the vector of derivatives of \( f(x) \) with respect to all the components of \( x \), \( \nabla f(x) = \left( \frac{\partial f}{\partial x^{(1)}}, \frac{\partial f}{\partial x^{(2)}}, \ldots, \frac{\partial f}{\partial x^{(N)}} \right) \), it defines the direction of maximum change for \( f(x) \). In the simplest approach we are performing a search for a minimum along one dimension defined by the direction of \( \nabla f(x) \).

If we come back to the specific example above we have

\[
g_0 \equiv \nabla f(x_0) = A^T \left[ Ax_0 - b \right]
\]

The new, partially optimized coordinate is searched along the \( g_0 \) direction

\[
x_1 = x_0 - \lambda g_0
\]

We need to determine the single unknown \( \lambda \) such that the function \( f(x_0) \) will be at a minimum along the line defined by \( g_0 \) and \( x_0 \).

At the minimum along that line the scalar product of the gradient of the function and \( g_0 \) is zero. We therefore have

\[
\bar{g} = A^T \left[ A(x_0 + \lambda g_0) - b \right]
\]

\[
g_0^T \bar{g} = g_0^T g_0 + \lambda A^T A g_0 = 0
\]

\[
\lambda = -g_0^T g_0 \left( g_0^T A g_0 \right)^{-1} = \frac{-\|g_0 g_0\|}{\|g_0 A\|}
\]
Exactly the same procedure can be repeated at the new point \( x_i = x_0 - \lambda g_0 \). We will have \( g_1 \) and can search for \( \lambda_2 \) such that \( x_2 = x_1 + \lambda_2 g_1 \) is a minimum along the line defined by \( g_1 \) and \( x_1 \). Such a search defined by the local gradient is referred to steepest descent search. This search is not efficient since the minimization along the \( g_1 \) direction may bring us into a new point with a component of the gradient along \( g_0 \). This means that at some point we will have to go back and minimize along \( g_0 \). It would be nice if we could set the search direction in such a way that if we minimize along the \( g_0 \) direction, we would never required to minimize along that direction again. If we have such a wonderful algorithm, a system of N dimensions will minimize to the global minimum after N linear optimizations. The computational complexity of the above problem is to operate with a matrix on a coordinate vector N times. If the matrix is sparse, this can be more efficient than matrix inversion.

Conjugate Gradient (CG) algorithm is doing exactly that for a quadratic system. The system we have above is indeed quadratic. There is no such guarantee for systems that are not quadratic in the variables. However close to minima any system is indeed quadratic and the CG seems to work quite well in general.

How is CG doing it?

Rather than minimizing along \( g_1 \), we minimize along a new direction \( h_1 \). The direction \( h_1 \) is set in such a way that at the minimum along the line \( x_2 = x_1 + \lambda_2 h_1 \) the function is minimized with respect to the \( g_0 \) direction as well. \( h_1 \) is determined as a mix of the previous direction and the current gradient. \( h_1 = g_1 + \gamma g_0 \). The unknown parameter \( \gamma \) is determined from the requirement that the gradient of the function at \( x_2 \) will be orthogonal to \( g_0 \). Hence we have two unknowns - \( \lambda_2, \gamma \) and two conditions to satisfy \( g_0 \nabla f (x_2) = 0 \quad g_1 \nabla f (x_2) = 0 \). Solving for the conditions we have

\[
h_1 = g_1 - \frac{g_1 A g_0}{g_0 A g_0} g_0 \quad \text{and} \quad \lambda_2 = -\frac{\|g_1 h_1\|}{\|A h_1\|^2}
\]

The next low point is \( x_3 = x_2 + \lambda_2 h_2 \)

\[
h_2 = g_2 - \frac{g_2 A h_1}{h_1 A h_1} h_1
\]

\[
\lambda_3 = -\frac{\|g_2 h_1\|}{\|A h_1\|^2}
\]

and so on…