MATH 519/619 Optimization
Winter quarter 2006

Homework set #1
Miguel Á. Carreira-Perpiñán

This set covers chapters 1 to 4 of the book *Numerical Optimization* by Nocedal and Wright.

From the book, the following exercises: 2.2, 2.3, 2.6, 2.7, 2.8, 2.12, 2.14, 3.1, 3.3, 3.4, 4.1, 4.10. In addition, the exercises below. The Matlab programming exercises are 3.1 and **4**–**7**. For exercises 2.2 and 2.8, plot your results with Matlab (for your own information); for exercise 3.1, estimate the convergence rate with the `convseq` function of exercise **4**.

Exercise 1. This is an extension of exercise 2.1. Consider the function $f(\mathbf{x}) = a(x_2 - x_1^2)^2 + (1 - x_1)^2$. Compute the gradient $\nabla f(\mathbf{x})$ and the Hessian $\nabla^2 f(\mathbf{x})$. Find and classify (as maxima, minima and saddle points) the stationary points of f. Compute the condition number of the Hessian at the stationary points. Plot the contours of f in the rectangle $[-2, 2] \times [-1, 3]$ for $a = 4,\ 10,\ 0,\ -1,\ -4$ (you may need to select the contours manually to view the stationary point).

Exercise 2. Consider the function $f(\mathbf{x}) = \frac{1}{2}(x_1^2 - x_2^2)$.

- Sketch the contours of f around its stationary point.
- Compute the steepest descent direction and the Newton direction at $\mathbf{x}$. Verify the steepest descent direction is a descent direction for any $\mathbf{x}$. For what points $\mathbf{x}$ is the Newton direction a descent direction?
- Compute explicitly $\mathbf{x}_{k+1}$ given $\mathbf{x}_0$ for the steepest descent method using constant step size $\alpha_k = \alpha > 0$. What does it tend to for $k \gg 1$? Study the geometry of the problem for $\alpha < 1$, $\alpha = 0$ and $\alpha > 1$. Test the following starting points $\mathbf{x}_0$: $\binom{0}{1}$, $\binom{1}{1}$, $\binom{2}{1}$, $\binom{1}{0}$. What happens with $\mathbf{x}_0 = \mathbf{0}$? What happens if using an exact line search?

Repeat for the function $f(\mathbf{x}) = x_1^2 + 2x_2^2$.

Exercise 3. Prove that the convergence of the sequences (2^{-k}), (k^{-k}) and (2^{-2^k}) is linear, superlinear and quadratic, respectively. Tabulate them for a few values of k.

Exercise 4. Consider a sequence $\mathbf{x}_0, \dots, \mathbf{x}_K \in \mathbb{R}^n$ produced by an optimization method where the optimizer is $\mathbf{x}^*$. We can empirically estimate the rate of the method by fitting a line to the consecutive distances: $\log d_{k+1} = a + b \log d_k$ where $d_k = \|\mathbf{x}_k - \mathbf{x}^*\|$; b will be the order of the method and $M = e^a$ the rate constant. Write a Matlab function `convseq` that takes as input a $(K+1) \times n$ matrix $\mathbf{X}$ (containing the sequence, rowwise) and a $1 \times n$ vector $\mathbf{x}^*$ and plots the K pairs of consecutive log-distances and the least-squares line, and gives the value of the order b and the constant M. Apply it to the sequences of the previous exercise.

Exercise 5. Program in Matlab the coordinate descent method using backtracking line search. Test it as in exercise 3.1. Estimate the convergence rate with the `convseq` function of exercise **4**.

Exercise 6. Program in Matlab the steepest descent method using exact line search for a quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x}$ (use the result of exercise 3.3). Test it for $\mathbf{x} \in \mathbb{R}^2$ with matrices $\mathbf{A}$ having condition numbers $\kappa(\mathbf{A}) = 2,\ 10, 100$ and plot the sequence of iterates as in fig. 3.7. Estimate the convergence rate with `convseq`.

Exercise 7. Like the previous exercise but for the coordinate descent method.