SECTIONS C: CONTINUOUS OPTIMISATION
REVISION CLASS 2, PART 2

HONOUR SCHOOL OF MATHEMATICS, OXFORD UNIVERSITY
HILARY TERM 2006, DR RAPHAEL HAUSER

Problem 1. Consider the following problem:

$$\begin{aligned} &\min\,(x_1+1)^2+x_2^2 \\ s.t.\quad & x_2 \le x_1^{3/2}, \\ & x_2 \ge -x_1^{3/2}. \end{aligned} \tag{0.1}$$

(i) [3 pts] Sketch the feasible region and argue by inspection of the sketch that $x^* = \left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$ is the optimal solution of (0.1).
(ii) [4 pts] Write down the Lagrangian $\mathcal{L}$, its x-gradient $\nabla_x\mathcal{L}$ and the KKT conditions for problem (0.1).
(iii) [5 pts] Show that there exists no Lagrange multiplier vector λ^* such that (x^*,λ^*) satisfies the KKT conditions. Explain why the KKT conditions are not necessary at x^*.
(iv) [4 pts] Now add the extra constraint $x_1^2+x_2^2 \ge 1$ and find a Lagrange multiplier vector $\widehat{\lambda}$ such that $(\widehat{x},\widehat{\lambda})$ satisfies the KKT conditions of the new problem, where $\widehat{x} = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$.
(v) [5 pts] Characterise the set of feasible exit directions from $\widehat{x}$ and use second order optimality conditions to show that $\widehat{x}$ is not a local minimiser of the problem with the extra constraint.
(vi) [4 pts] Use necessary optimality conditions to show that $\widehat{x}$ is not a local maximiser of the problem with the extra constraint either.

Problem 2. Consider applying the conjugate gradient algorithm to the unconstrained minimisation problem

$$\min_{x\in\mathbb{R}^n} f(x) = \frac{1}{2}x^{\mathrm{T}}Ax + b^{\mathrm{T}}x + c,$$

where $A \succ 0$ is a positive definite symmetric $n\times n$ matrix, $b\in\mathbb{R}^n$ and $c\in\mathbb{R}$. Recall that the algorithm proceeds via exact line searches, starting at some $x_0\in\mathbb{R}^n$ and with search directions

$$\begin{aligned} d_0 &= -\nabla f(x_0), \\ d_k &= -\nabla f(x_k) + \frac{\|\nabla f(x_k)\|^2}{\|\nabla f(x_{k-1})\|^2} d_{k-1}. \end{aligned}$$

(i) [2 pts] Show by induction on k that

$$\operatorname{span}\{d_0,\dots,d_k\} \subseteq \operatorname{span}\{\nabla f(x_0),\dots,\nabla f(x_k)\}, \qquad (k=0,\dots,n). \tag{0.2}$$

(ii) [6 pts] Show by induction that

$$\nabla f(x_k) \in \mathcal{K}_k := \text{span}\{\nabla f(x_0), A\nabla f(x_0), \ldots, A^k \nabla f(x_0)\}, \qquad (k = 0, \ldots, n).$$

(iii) [7 pts] Now let A have r distinct eigenvalues $\lambda_1, \ldots, \lambda_r$ (that is, if $r < n$ then some of the eigenvalues appear with multiplicity > 1). Show that there exist eigenvectors $v_1, \ldots, v_r$ corresponding to $\lambda_1, \ldots, \lambda_r$ such that $\nabla f(x_0) \in \text{span}\{v_1, \ldots, v_r\}$.

(iv) [4 pts] Using part (iii), show that $\mathcal{K}_k \subseteq \text{span}\{v_1, \ldots, v_r\}$.

(v) [6 pts] Why does part (iv) imply that the algorithm converges in at most $r + 1$ iterations? You may use the fact that in the lectures we proved that $\nabla f(x_j) \perp \nabla f(x_k)$ for all $j \neq k$.