NUMERICAL OPTIMISATION
TUTORIAL 24/01/20
ASSIGNMENT 1

submit by 11pm on Thursday 30/01

Marta Betcke

EXERCISE 1 [DEMO]

(a) Given the function

$$f(x, y) = 2x + 4y + x^2 - 2y^2$$

(i) Visualise the function and its contours. **[0pt]**

(ii) Calculate the contours analytically. **[0pt]**

(iii) Calculate the gradient analytically. **[0pt]**

Find the stationary points and classify them i.e. are them minima, maxima or something else? **[0pt]**

(b) Given the function

$$g(x, y) = (\cos^2(\sqrt{x^2 + 2y^2}) + 2)(x^2 + 2y^2)$$

(i) Visualise the function and its contours. **[0pt]**

(ii) Apply the steepest descent method to find the minimum of this function starting from $(x, y) = (15, 15)$ and an appropriate step length. Is convergence guaranteed? **[0pt]**

EXERCISE 2

(a) Show that $A = B^{\mathrm{T}}B$ is symmetric positive semidefinite for all $B \in \mathbb{R}^{n\times n}$. *Hint: use the Rayleigh quotient representation of the eigenvalue $Ax = \lambda x$.*
*Submit your solutions via **Turnitin**.* **[10pt]**

(b) Let $f(x) = x^{\mathrm{T}}Ax$ with A symmetric positive semidefinite matrix $A \in \mathbb{R}^{n\times n}$. Show that $f(x)$ is convex on the domain $\mathbb{R}^n$. *Hint: you may want to show the equivalent inequality instead*

$$f(y + \alpha(x - y)) - \alpha f(x) - (1 - \alpha)f(y) \leq 0.$$

*Submit your solutions via **Turnitin**.* **[20pt]**

EXERCISE 3

Compute 2nd degree Taylor expansion (the reminder term is of 2nd order) of the given functions

(a) $f(x) = \cos(1/x)$ for $x \in \mathbb{R}^+$. **[10pt]**

(b) $g(\mathbf{x}) = \exp(||\mathbf{x}||^2)$ for $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$. *Hint: expand along a fixed direction $p \in \mathbb{R}^2$.* **[10pt]**

Submit your solutions via ***`Turnitin`****.*

EXERCISE 4

(a) Code backtracking line search, steepest descent and Newton's algorithms. See `MATLAB Grader` for more guidance.
Submit your implementation via `MATLAB Grader`. **[20pt]**

(b) Apply steepest descent and Newton's algorithms (with backtracking line search) to minimise the Rosenbrock function

$$f(x) = 100(y - x^2)^2 + (1 - x)^2.$$

Set the initial point $x_0 = (1.2, 1.2)^T$ and the initial step length $\alpha_0 = 1$. Plot the step sizes used by each method over the iterations as well as the trajectories traced by the iterates in $\mathbb{R}^2$. Try explaining the obtained trajectories.
Submit solution via `TurnitIn`. **[5pt]**

(c) Redo the calculations in b) with the more difficult starting point $x_0 = (-1.2, 1)^T$ and try explaining the obtained trajectories.
Submit solution via `TurnitIn`. **[5pt]**

(d) Plot the convergence of the iterates in b) and c). What is the qualitative convergence rate we obtain for the steepest descent algorithm? Can this be quantified? What is the qualitative convergence we obtain with the Newton's algorithm?
Submit solution via `TurnitIn`. **[20pt]**

(e) **(optional)** Repeat the calculations in b) and c) using the line search in Algorithm 3.5 from Nocedal, Wright. This line search produces step lengths which satisfy the strong Wolfe conditions. Use the implementation provided on Moodle: `lineSearch.m`, `zoomInt.m`. Compare the new step lengths with those obtained with backtracking.
Submit solution via `TurnitIn`. **[0pt]**

<u>Remark</u> The submission to `Turnitin` should not be longer than 5 pages. Avoid submitting more code than needed (if any) and focus on explaining your results.