EECS 127/227AT Optimization Models in Engineering
Spring 2019 **Homework 1**

Release date: 9/05/19.
Due date: 9/12/19, 23:00 (11 pm). Please LaTeX or handwrite your homework solution and submit an electronic version.

Submission Format
Your homework submission should consist of a single PDF file that contains all of your answers (any handwritten answers should be scanned) as well as your IPython notebook saved as a PDF.

If you do not attach a PDF "printout" of your IPython notebook, you will not receive credit for problems that involve coding. Make sure that your results and your plots are visible. Assign the IPython printout to the correct problem(s) on Gradescope.

1. Norm and angles

(a) Let $x, y \in \mathbb{R}^n$ be two unit-norm vectors, that is, such that $\|x\|_2 = \|y\|_2 = 1$. Show algebraically that the vectors $x - y$ and $x + y$ are orthogonal. Then, show this graphically by drawing the two vectors on the 2D plane, as well as any other necessary shapes. You may use right angles, circles and straight lines to make your point.

(b) Show that the following inequalities hold for any vector $x \in \mathbb{R}^n$:

$$\frac{1}{\sqrt{n}}\|x\|_2 \leq \|x\|_\infty \leq \|x\|_2 \leq \|x\|_1 \leq \sqrt{n}\|x\|_2 \leq n\|x\|_\infty.$$

Hint: For $\|x\|_1 \leq \sqrt{n}\|x\|_2$, how might you express $\|x\|_1$ as the dot product of two vectors? Can you then use the Cauchy-Schwarz inequality to bound this dot product?

(c) Show that for any non-zero vector x,

$$\mathrm{card}(x) \geq \frac{\|x\|_1^2}{\|x\|_2^2},$$

where $\mathrm{card}(x)$ is the *cardinality* of the vector x, defined as the number of non-zero elements in x. Find all vectors x for which the lower bound is attained.

Hint: Try using Cauchy-Schwarz like in the previous part.

2. Gradients and Hessians

The *gradient* of a scalar-valued function $g : \mathbb{R}^n \to \mathbb{R}$, is the column vector of length n, denoted as ∇g, containing the derivatives of components of g with respect to the variables:

$$(\nabla g(x))_i = \frac{\partial g}{\partial x_i}(x),\ i = 1, \dots n.$$

The *Hessian* of a scalar-valued function $g : \mathbb{R}^n \to \mathbb{R}$, is the $n \times n$ matrix, denoted as $\nabla^2 g$, containing the second derivatives of components of g with respect to the variables:

$$(\nabla^2 g(x))_{ij} = \frac{\partial^2 g}{\partial x_i \partial x_j}(x), \quad i = 1, \dots, n, \quad j = 1, \dots, n.$$

For the remainder of the class, we will repeatedly have to take gradients and Hessians of functions we are trying to optimize. This exercise serves as a warm up for future problems.

Compute the gradients and Hessians for the following functions:

(a) $g(x) = y^\top Ax$

(b) $g(x) = x^\top Ax$

(c) $g(x) = \|Ax - b\|_2^2$

(d) $g(x) = \sin(x_1^2)\log(x_3 - x_2)$ where x_i are scalars and $x_3 - x_2 > 0$.

3. Jacobians

The *Jacobian* of a vector-valued function $g : \mathbb{R}^n \to \mathbb{R}^m$ is the $m \times n$ matrix, denoted as Dg, containing the derivatives of components of g with respect to the variables:

$$(Dg)_{ij} = \frac{\partial g_i}{\partial x_j}, \quad i = 1, \dots, m, \quad j = 1, \dots, n.$$

In class, we will also be using the Jacobian frequently.

Compute the Jacobians for the following maps:

(a) $g(x) = Ax$

(b) $g(x) = f(x)x$ where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is once-differentiable

(c) $g(x) = f(Ax + b)x$ where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is once differentiable and $A \in \mathbb{R}^{n \times n}$.

(d)

$$g(x) = \begin{bmatrix} x_1^2/x_2 \\ \log(x_3)\sin(x_1/x_3) \end{bmatrix}$$

4. Level Sets

Plot/hand-draw the level sets of the following functions.

Also draw the gradient directions in the level-set diagram. Additionally, compute the first and second order Taylor series approximation around the point $(1,1)$ for each function and comment on how accurately they approximate the true function.

(a) $g(x_1, x_2) = \dfrac{x_1^2}{4} + \dfrac{x_2^2}{9}$

(b) $g(x_1, x_2) = x_1 x_2$

5. Jupyter Notebook Setup

Note: Please feel free to ask for help from the TAs during office hours and homework parties to ensure your conda environment works as it should

Conda Download

If you already, have anaconda, skip to the Conda Environment header. The goal of this problem is to confirm that you are proficient with the software environment, which you will need to complete the class.

(a) To get started, download anaconda from this link: `https://www.anaconda.com/distribution/`

(b) Download the command line installer

(c) Run the downloaded script

(d) Follow the instructions on the terminal

(e) Quit the terminal and restart it — this refreshes the environment and lets your terminal see Conda

Conda Environment

To create the environment we will be using in this class, run the following command:
`conda env create -f ee127_***.yml`, where `***` corresponds to your system (mac, linux, or windows).

For windows users, you may need to download Visual Studio. If for any reason, the environment creation fails, please refer to this link for instructions on how to manually install the correct libraries.

Then, run `conda activate ee127`. **You will need to run this command every time you start doing any code work for this class**

Please refer to the corresponding jupyter notebook for the rest of the question: `intro_to_jupyter.ipynb`