TECHNISCHE UNIVERSITÄT MÜNCHEN
Fakultät für Elektrotechnik und Informationstechnik
Fachgebiet für Geometrische Optimierung und Maschinelles Lernen
Prof. Dr. Martin Kleinsteuber

Non-convex Optimization for Analyzing Big Data

Assignment #2, May 31, 2016

Due date: **10.06.2016, 15:00**

Please hand in your solutions via the Moodle Forum. You can add your conclusions for the Matlab Tasks as comments in the Matlab files. For the other exercises deliver a PDF file either created using Latex or as a scan of your handwritten solution.

Solutions can be handed in by groups of up to **two** people. Please list the members of your group at the beginning of your submission.

Chain and Product rule

Task 1: *[5 Points]*

(a) Given the functions $f\colon \mathbb{R}^n \to \mathbb{R}$, $\mathbf{x} \mapsto \mathbf{x}^\top \mathbf{x}$ and $g\colon \mathbb{R}^n \to \mathbb{R}^n$, $\mathbf{z} \mapsto \mathbf{A}\mathbf{z}$, where $\mathbf{A}$ is a symmetric, positive definite $n \times n$ matrix. Use the chain rule to determine the directional derivative and the gradient of $f \circ g$ at a point $\mathbf{x}_0$ using the standard inner product.

(b) Given the two functions $f\colon \mathrm{skew}_n \to \mathbb{R}^n$, $\mathbf{X} \mapsto \mathbf{X}^\top \mathbf{a}$ and $g\colon \mathrm{skew}_n \to \mathbb{R}^n$, $\mathbf{X} \mapsto \mathbf{X}\mathbf{b}$, where $\mathrm{skew}_n := \{\mathbf{X} \in \mathbb{R}^{n\times n} : \mathbf{X}^\top = -\mathbf{X}\}$ is the set skew-symmetric $n \times n$ matrices and $\mathbf{a}, \mathbf{b}$ are vectors in $\mathbb{R}^n$ unequal to zero. Determine the gradient of the function $h(\mathbf{X})\colon \mathrm{skew}_n \to \mathbb{R}$ defined as

$$h(\mathbf{X}) = f(\mathbf{X})^\top \cdot g(\mathbf{X})$$

in accordance to the standard matrix inner product.

Gradient descent

Task 2: *[5 Points]*

In this task you are supposed to implement a gradient descent algorithm that solves a multilinear regression problem, that is, given a matrix of predictor variables $\mathbf{X} \in \mathbb{R}^{n\times p}$ and a set of target variables $\mathbf{y} \in \mathbb{R}^n$ find the model parameters $\boldsymbol{\beta} \in \mathbb{R}^p$ such that $\mathbf{y} \approx \mathbf{X}\boldsymbol{\beta}$.
Download the file `A2_mats.zip` from Moodle. The `.dat` files contain the training data which is an excerpt of housing prices in Portland, Oregon. The file `A2T2x.dat` contains the living area in square feet and the number of bedrooms while `A2T2y.dat` contains the corresponding prices in USD. The Matlab script provided as `A2_Task2.m` contains code that imports the data and performs a preprocessing step. In this preprocessing step we normalize the columns of **Xorg** and append a column of all ones (which is called the intercept). The

resulting matrix is saved as $\mathbf{X}$.
Your task is to implement a gradient descent algorithm that solves the minimization problem

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} f(\boldsymbol{\beta})$$
$$f(\boldsymbol{\beta}) = \frac{1}{2n} \|\mathbf{X}\boldsymbol{\beta} - \mathbf{y}\|_2^2.$$

Use backtracking to find an appropriate step size. That is, in the k-th iteration start with a step size $\alpha = 1$ and incrementally decrease α until the Armijo condition

$$f(\boldsymbol{\beta}^{(k)} + \alpha \mathbf{h}^{(k)}) \leq f(\boldsymbol{\beta}^{(k)}) + \alpha \cdot c \cdot \langle \nabla f(\boldsymbol{\beta}^{(k)}), \mathbf{h}^{(k)} \rangle$$

is met. Here, $\boldsymbol{\beta}^{(k)}$ is the current iteration point, $\mathbf{h}^{(k)}$ is the search direction, c is a scalar between 0 and 1 (a good choice is $c = 0.9$), and $\langle \cdot, \cdot \rangle$ is the standard inner product. As a starting point use $\boldsymbol{\beta}^{(0)} = (0, 0, 0)^\top$. Stop the optimization procedure if the difference between consecutive iteration points $\boldsymbol{\beta}^{(k)}$ and $\boldsymbol{\beta}^{(k+1)}$ is smaller than $\varepsilon = 10^{-6}$ for 5 consecutive iterations.

Once you have found a solution to this minimization problem, answer the following questions:

(a) What are the final values of $\boldsymbol{\beta}$?

(b) According to your trained model, what would be the price for a flat with 1850 square feet and 3 bedrooms? (Don't forget the normalization.)

Please comment your code appropriately!