Optimisation Algorithms in Statistics I – Autumn 2020
Assignment 1

Frank Miller, frank.miller@stat.su.se,
Department of Statistics, Stockholm University

October 2, 2020

Perform the solutions individually and send your report **until October 12** to me. Try to keep this deadline. However, if you have problems with it, there will be a final deadline on January 25 for all assignments.
For Problem 1.3 in this assignment: highly recommended to perform it until the first deadline since we want to build on it later. If you do not solve it until the first deadline, you will get another problem as replacement.

Problem 1.1

We have independent data $x_1, \ldots, x_n$ from a Cauchy-distribution with unknown location parameter θ and known scale parameter 1. The log likelihood function is

$$-n \log(\pi) - \sum_{i=1}^{n} \log(1 + (x_i - \theta)^2),$$

and it's derivative with respect to θ is

$$\sum_{i=1}^{n} \frac{2(x_i - \theta)}{1 + (x_i - \theta)^2}.$$

Data of size $n = 12$ is given:
$\mathbf{x} = (0.72, 5.50, -2.21, -0.35, -0.67, 0.16, 23.64, 1.00, 1.06, -495.81, -1.98, 37.72)$.

a. Plot the log likelihood function for the given data within an appropriate range. Plot the derivative in the same range and check visually how often the derivative is equal to 0.

b. Program the Newton-Raphson method. You need the second derivative and you might use the R-function D to help. Choose suitable starting values (based on your plots) to identify all local maxima of the likelihood function.

c. How would you use your Newton-Raphson function if you want to determine the global maximum and if you do not want to choose the starting value based on a plot (i.e. the algorithm should run automatised)?

Problem 1.2

Let

$$g(x, y) = -x^2 + 10y - 2y^3 + \frac{1}{2}x^2 y.$$

a. Plot the function with a contour plot.

b. **(optional)** If you want, you can add also a 3-dimensional plot to visualise the function.

c. Compute the gradient analytically. Set the gradient to 0 and solve the equations analytically to identify candidates for maxima, minima, and saddle points.

d. Compute the Hessian matrix analytically. Determine if it is positive, negative, or indefinite in the candidate points (if you want to calculate Eigenvalues, you can use software for it).

Problem 1.3

Three doses (0.1, 0.3, and 0.9 g) of a drug and placebo (0 g) are tested in a study. A dose-dependent event is recorded afterwards. The data of $n = 10$ subjects is shown in Table 1; x_i is the dose, $y_i = 1$ if the event occured, $y_i = 0$ otherwise.

x_i	0	0	0	0.1	0.1	0.3	0.3	0.9	0.9	0.9
y_i	0	0	1	0	1	1	1	0	1	1

Table 1: Data for Problem 1.3b

You should fit a simple logistic regression

$$p(x) = P(Y = 1|x) = \frac{1}{1 + \exp(-\beta_0 - \beta_1 x)}.$$

to the data, i.e. estimate β_0 and β_1.

a. Program your own function to determine the ML-estimator for (β_0, β_1) using the steepest ascent method. Note that the log-likelihood function was mentioned in the lecture. You need the derivative; you can either calculate it yourself or search it in the literature. If you use information from literature, please cite appropriately.

b. Plot the log-likelihood function (contour plot) for the observations in Table 1. Compute the ML-estimator with your function and compare with the solution when using the function `glm` in `R`.

c. To test how your function works for large datasets, generate a dataset with a large n by duplicating the dataset in Table 1 e.g. 100000 times. Does your function still work satisfactory? Observe the running time of your function. You can use the function `proc.time()` in `R` to measure running time.

d. **(optional)** Modify your ML-solver for multiple logistic regression with p explanatory variables. How large can the dataset be (number of observations n and number of parameters $p + 1$) such that the running time is still acceptable?