Exercises for Lecture Course on Modeling and System Identification (MSI)
Albert-Ludwigs-Universität Freiburg – Winter Term 2019

Exercise 9: Gauss Newton Algorithm and Inroduction to Machine Learning

(to be returned on Jan 29th, 2020, 8:30 in HS 00 036 (Schick - Saal), or before in building 102, 1st floor, 'Anbau')

Prof. Dr. Moritz Diehl, Tobias Schöls, Katrin Baumgärtner, Naya Baslan, Jakob Harzer, Bryan Ramos

In this exercise you will implement the Gauss Newton Alogorithm and apply some basic knowledge in machine learning.

Exercise Tasks

1. **Gauss Newton Algorithm** (6 points)

   In this task you will solve an unconstrained minimization problem by implementing the Gauss Newton algorithm. You are given a residual function of the form:

$$f(\theta) = \frac{1}{2}\|R(\theta)\|_2^2 \quad \theta = [\theta_1, \theta_2]^\top \quad \text{where}$$

$$R(\theta) = [\theta_1 - 1, \quad 10(\theta_2 - \theta_1^2), \quad \theta_2]^T \quad \text{and} \quad f(\theta) = \frac{1}{2}(\theta_1 - 1)^2 + \frac{1}{2}(10(\theta_2 - \theta_1^2))^2 + \frac{1}{2}(\theta_2^2)$$

   Look carefully at the function plots shown on Grader and try to guess where the minimizer is.

   (a) ON PAPER: Derive the gradien and the exact Hessian matrices of the function $f(\theta)$.

   (b) ON PAPER: Derive the Gauss-Newton Hessian and compare it with the exact one.

   (c) ON PAPER: Under which circumstances do Gauss Newton and exact Hessian coincide?

   (d) MATLAB: Fill in the code on Grader to implement the Gauss-Newton algorithm.

   (e) ON PAPER:What happens when the function to be minimized has local minima?

   (f) ON PAPER: What is stochastic gradient? Why is it commonly used instead of e.g. Gauss Newton if the training data set it very large?

2. **Introduction to Machine Learning** (6 points)

   The neural network shown in Figure 1 below is used to solve the XOR problem. The training data is given in the table below. As seen in the plot in Figure 2, the XOR is a non-linearly separable function. This is why the network has a hidden layer with 2 neurons. This network simply represents a set of weighted inputs to which we apply the sigmoid activation function, which is defined as the following:

$$\sigma(z) = \frac{1}{1 + \exp(-z)}$$

   (a) ON PAPER:Fill out the equations corresponding to the XOR network:
   Note that $x \in R^2$ and $w \in R^9$ where w_7, w_8, w_9 are the biases. For simplified notation, we define then as the last three elements of the weight vector w. (1 point)

   $h_1(x, w) = \sigma(\qquad\qquad)$
   $h_2(x, w) = \sigma(\qquad\qquad)$
   $\hat{y}(x, w) \ = \sigma(\qquad\qquad)$

   ***Hint**: Assume we are using a sigmoid activation function.*

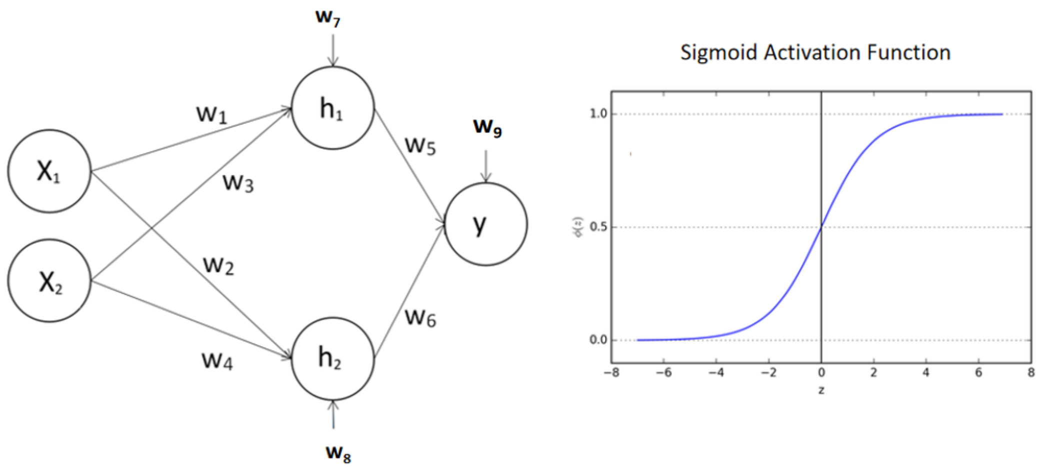

Figure 1: Neural network for learning the xor function.

x_1	x_2	output	plot
0	0	0	-
0	1	1	+
1	0	1	+
1	1	0	-

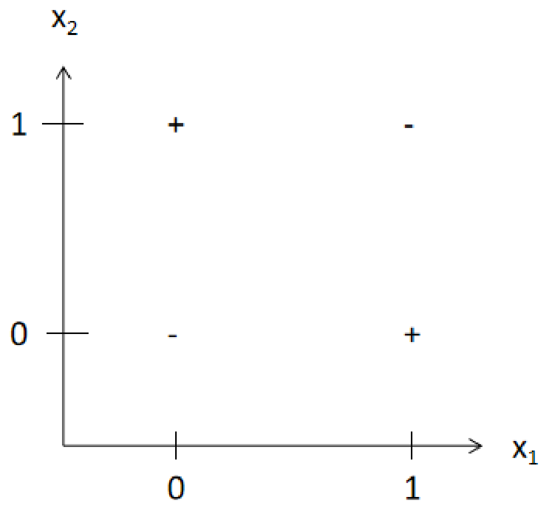

Figure 2: Dataset for training the network.

(b) ON PAPER: For a given training set $(x^{(k)}, y^{(k)})$ for $k = 1, 2, ..., N$ where $x \in R^2$ and $y \in R$, formulate the cost function that should be minimized to estimate the model parameters w. State the optimization problem. (2 points)

(c) MATLAB: Solve the optimization problem in Grader using the `lsqnonlin` and give the values obatined for all the weights and the biases. (3 points)
***Hint**: You should first implement h_1, h_2, $y_h at$, and the sigmoid activation function and initialize your weight vector properly.*

This sheet gives in total 12 points