Exercise sheet 4

The Exercises are meant to be done in groups of three, **please enroll your group in OLAT**. Even if you are doing the exercies on your own, please enroll yourself in an enmpty group in OLAT. Without enrollment, you wont be able to submit.
Submission of your solutions is done via OLAT. Please add the names of all group members to your solution.

Excercise 4.1: Vanishing and Exploding Gradients in Deep Networks (5 Points)

Consider a deep feedforward neural network with L layers. Let the forward propagation be defined by:

$$z^{(l)} = W^{(l)} x^{(l-1)} + b^{(l)}$$
$$x^{(l)} = \sigma(z^{(l)})$$

where $x^{(l)}$ is the activation vector of the l-th layer, $W^{(l)}$ is the weight matrix connecting layer $l-1$ to layer l, $b^{(l)}$ is the bias vector for layer l, and $\sigma(\cdot)$ is an element-wise differentiable activation function. Let E be the scalar loss function at the final output layer $x^{(L)}$.
Analyze the behavior of the gradient of the loss with respect to the weights of an early layer, $W^{(k)}$ (where $k \ll L$), as the depth of the network increases. What problems can arise as L becomes very large.
Hint: Think about what can happen when gradients become large or small.

Excercise 4.2: Minimization (8 Points)

Let $x_1, x_2, \ldots, x_N \in \mathbb{R}^n$ and $M := \sum_{i=1}^{N} x_i x_i^T$. Show that
(a.1) M is symmetric and positive-semidefinite

(a.2) M is positive-definite $\Leftrightarrow$ dim span$\{x_1, x_2, \ldots, x_N\} = n$

Consider now the function

$$f : \mathbb{R}^n \to \mathbb{R},\ x \mapsto \frac{1}{2} x^T M x + m^T x + c,$$

where $M \in \mathbb{R}^{n\times n}$ symmetric positive-definite, $m \in \mathbb{R}^n$ and $c \in \mathbb{R}$. Show that

(b) f has a unique minimum x_{min}.

(c) Every solution the differential equation $\dot{x} = -Mx - m$ converges to x_{min}.

Excercise 4.3: Project 3 (5 Points)

The objective of this assignment is to get familiarized with implementing custom optimization modules. The project files are presented as python notebook files in Material/Projects/Proj3 folder. It contains the script file: adam_regression_classification.ipynb. This file implements a one variable and two variable regression problem as in the previous project.

Let $g_t = \nabla f_t(\theta_t)$ be the gradient of the stochastic loss function f_t, and $\alpha, \beta_1, \beta_2, \epsilon$ be some given constants (hyperparameters). The Adam optimization method has the following update rules at step t:

$$\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t, \quad m_0 = 0 \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2, \quad v_0 = 0 \\
\hat{m}_t &= \frac{m_t}{1-\beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t}, \\
\theta_{t+1} &= \theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon},
\end{aligned}$$

1. Implement the Adam algorithm (mentioned above) to update the weights and biases.
2. Compare the results of training and testing with that of Project 2 in Exercise 3.

Submission guidelines:

1. Please submit the solved file (with outputs for each code-block) along with the theory exercise solutions. Please name your submission file as adam_regression_classification_solved_group_id.ipynb, where id is your two digit group number. For example, if you belong to Group01 please name the file as: adam_regression_classification_solved_group_01.ipynb
2. Please ensure that the submitted ipynb files are runnable and does not produce any errors. The project shall be considered for marking <u>if and only if</u> there are <u>zero</u> errors in the script file.