Universität Salzburg — Florian Graf

Machine Learning

Übungsblatt **11** — 15 Punkte

Aufgabe 1. *Ableitungen* 6 P.

Es seien

$$f : \mathbb{R}^n \times \mathbb{R}^{m\times n} \times \mathbb{R}^m, \qquad (\mathbf{x}, \mathbf{W}, \mathbf{b}) \mapsto \mathbf{W}\mathbf{x} + \mathbf{b} \ ,$$
$$g : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^{m\times n} \times \mathbb{R}^m, \quad (\mathbf{x}, \mathbf{a}, \mathbf{W}, \mathbf{b}) \mapsto \mathbf{a}^\top \mathrm{ReLU}(\mathbf{W}\mathbf{x} + \mathbf{b}) \ .$$

Bestimmen Sie die folgenden partiellen Ableitungen (wo sie wohldefiniert sind).

(a) $\frac{\partial [f(\mathbf{x},\mathbf{W},\mathbf{b})]_1}{\partial x_j}$

(b) $\frac{\partial [f(\mathbf{x},\mathbf{W},\mathbf{b})]_1}{\partial W_{ij}}$

(c) $\frac{\partial [f(\mathbf{x},\mathbf{W},\mathbf{b})]_1}{\partial b_i}$

(d) $\frac{\partial [\mathrm{ReLU}(f(\mathbf{x},\mathbf{W},\mathbf{b}))]_1}{\partial x_j}$

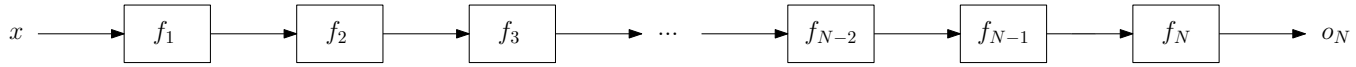
(e) $\frac{\partial [\mathrm{sigmoid}(f(\mathbf{x},\mathbf{W},\mathbf{b}))]_1}{\partial x_j}$

(f) $\frac{\partial g(\mathbf{x},\mathbf{a},\mathbf{W},\mathbf{b})}{\partial x_j}$

Hierbei steht $[\cdot]_1$ für die erste Komponente eines Vektors.
Im Falle von (e), drücken Sie die Ableitung mithilfe der Sigmoid-Funktion aus.

Aufgabe 2. *Verschwindende Gradienten* 9 P.

Gegeben sei ein neuronales Netz mit N linearen Layern, das auf skalaren Eingabedaten $x_i \in \mathbb{R}$ operiert.



Formal bedeutet dies, dass für jeden Layer $i = 1, \ldots, N$,

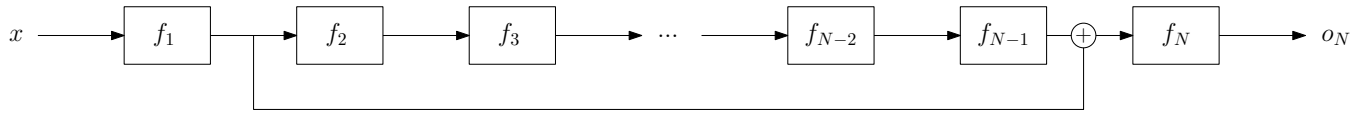
$$s_i = f_i(o_{i-1}) = w_i o_{i-1} + b_i \tag{1}$$
$$o_i = \sigma(s_i) \ , \tag{2}$$

wobei σ eine (beliebige) Aktivierungsfunktion und $o_0 = x$ ist. Einfachheitshalber besteht jeder Layer aus nur einem Neuron, sodass $w_i, b_i, o_i, s_i \in \mathbb{R}$ skalarwertig sind.

(a) Bestimmen Sie die Ableitung $\frac{\partial o_N}{\partial w_1}$ des Outputs nach dem Gewicht des ersten Layers in Abhängigkeit von s_i, w_i (für $i = 1, \ldots, N$), x und der Ableitung der Aktivierungsfunktion $\sigma'(\cdot)$.

(b) Erklären Sie mithilfe von (a) und Aufgabe 1 das Vanishing-, bzw. Exploding-Gradient Problem.

Wir ändern nun die Architektur durch das Einführen einer sogenannten Skip Connection, die die Layer 2 bis $N-1$ überspringt.



(c) Adaptieren Sie die Formeln (1) und (2) für $i = N$ auf die geänderte Architektur.

(d) Bestimmen Sie $\frac{\partial o_N}{\partial w_1}$ für die geänderte Architektur.

(e) Wie wirkt sich die Skip Connection bzgl. des Vanishing-, bzw. Exploding-Gradient Problems aus?

(f) Wie lässt sich die Netzwerkarchitektur anpassen, sodass auch das Layer f_j (für $j > 1$ aber klein) günstigere Gradienten bekommt?