Minimizing Movement Scheme for Deep Learning

We want to use a minimal movement optimization scheme to avoid using Fourier Features. Therefore, we set

$$\mathcal{A}(\theta) := \int_\Omega \frac{1}{4\varepsilon}(v_\theta(x)-1)^2 + \varepsilon\|\nabla v_\theta(x)\|^2 \mathrm{d}x + \frac{C\varepsilon^{-\frac{1}{3}}}{|\mathcal{P}|}\sum_{p\in\mathcal{P}}|v_\theta(p)| \qquad \varepsilon = 10^{-4}, C = 14$$

where v_θ is a neural network v with parameter θ.

Let θ_0 be a random set of parameters. Then we define

$$\begin{aligned}\theta_{k+1} :&= \mathrm{argmin}_\Theta \Phi(\Theta;\theta_k) \\ &:= \mathrm{argmin}_\Theta \|v_\Theta - v_{\theta_k}\|^2_{H^1(\Omega;\tau)} + \tau\mathcal{A}(\Theta) \\ &= \mathrm{argmin}_\Theta \int (v_\Theta(x) - v_{\theta_k}(x))^2 + \tau\left(\nabla v_\Theta(x) - \nabla v_{\theta_k}(x)\right)^2 \mathrm{d}x + \tau\mathcal{A}(\Theta)\end{aligned} \tag{1}$$

with $\tau = \varepsilon$.

We iterate this, until we reach set of parameters θ_K, with $K \in \mathbb{N}$.

Algorithmus 1 Deep Minimizing Movement Scheme

Require: $n \geq 0$
Ensure: $y = x^n$
Guess θ_0
for $k = 0, ..., K$ **do**
    $v_k \leftarrow$ NN with parameter θ_k in Evaluation mode only
    $\theta_{k+1} \leftarrow \theta_k$
    $v_{k+1}\leftarrow$ NN with parameter θ_{k+1}
    for Number of Trainingssteps **do**
        Loss function$\leftarrow\Phi(\theta_{k+1}, \theta_k)$
        Backpropagate for training v_{k+1}
        Make Optimization step
    end for
end for