

1 Adjoint Method

Let M a manifold, and $x(t) \in C^1(\mathbb{R}, M)$ a trajectory, obeying

$$\frac{dx}{dt}(t) = f(t, x(t); \theta),$$

and

$$x(t_0) = x_0,$$

where $f \in C(\mathbb{R} \times M, T_M)$ parameterized by θ . For $\forall t_1 > t_0$, let

$$x_1 := x_0 + \int_{t_0}^{t_1} f(t, x(t); \theta) dt.$$

Then

Theorem 1. *Let $L \in C^1(M, \mathbb{R})$, and $\forall x(t) \in C^1(\mathbb{R}, M)$ obeying dynamics $f(t, x; \theta) \in C^1(\mathbb{R} \times M, T_M)$ with initial value $x(t_0) = x_0$. Denote*

$$L_x(t; t_0, x_0, \theta) := L(x(t; t_0, x_0, \theta)) = L\left(x_0 + \int_{t_0}^t f(t, x(t); \theta) dt\right),$$

and XXX

$$a_\alpha(\tau) := \frac{\partial L_x}{\partial x_0^\alpha}(t; \tau, x_0; \theta).$$

Then we have

$$\frac{\partial L_x}{\partial x_0^\alpha}(t) = - \int_{t_0}^t a_\beta(\tau) \frac{\partial f^\beta}{\partial x^\alpha(\tau)}(\tau, x(\tau); \theta) d\tau$$

and

$$\frac{\partial L_x}{\partial \theta}(t) = - \int_{t_0}^t a_\beta(\tau) \frac{\partial f^\beta}{\partial \theta}(\tau, x(\tau); \theta) d\tau$$

Proof. Suppose the $x(t)$ is layerized, the L depends on the variables (inputs and model parameters) on the i th layer can be regarded as the loss of a new model by truncating the original at the i th layer, which we call $L_i(z_i)$.

$$\begin{aligned} \frac{\partial L_i}{\partial x_i^\alpha}(x_i) &= \frac{\partial L_{i+1}}{\partial x_{i+1}^\beta}(x_{i+1}) \frac{\partial x_{i+1}^\beta}{\partial x_i^\alpha}(x_i) \\ &= \frac{\partial L_{i+1}}{\partial x_1^\beta}(x_{i+1}) \frac{\partial}{\partial x_i^\alpha}(x_i^\beta + f^\beta(t_i, x_i; \theta) \Delta t) \\ &= \frac{\partial L_{i+1}}{\partial x_{i+1}^\alpha}(x_{i+1}) + \frac{\partial L_{i+1}}{\partial x_{i+1}^\beta}(x_{i+1}) \partial_\alpha f^\beta(t_i, x_i; \theta) \Delta t. \end{aligned}$$

This hints that

$$\frac{d}{dt} \frac{\partial L(x(t_1))}{\partial x^\alpha(t)} = - \frac{\partial L(x(t_1))}{\partial x^\beta(t)} \frac{\partial f^\beta}{\partial x^\alpha}(t, x(t); \theta).$$

Varying θ will vary the $L_i(x_i)$ from two aspects, the effect from $\partial L_{i+1}/\partial\theta$ and the Δx_{i+1} caused by $\Delta\theta$.

$$\begin{aligned}\frac{\partial L_i}{\partial\theta}(x_i) &= \frac{\partial L_{i+1}}{\partial\theta}(x_{i+1}) + \frac{\partial L_{i+1}}{\partial x_{i+1}} \frac{\partial x_{i+1}}{\partial\theta} \\ &= \frac{\partial L_{i+1}}{\partial\theta}(x_{i+1}) + \frac{\partial L_{i+1}}{\partial x_{i+1}} \frac{\partial}{\partial\theta}(x_i^\beta + f^\beta(t_i, x_i; \theta)\Delta t) \\ &= \frac{\partial L_{i+1}}{\partial\theta}(x_{i+1}) + \frac{\partial L_{i+1}}{\partial x_{i+1}} \frac{\partial f^\beta}{\partial\theta}(t_i, x_i; \theta)\Delta t.\end{aligned}$$

This hints that

$$\frac{d}{dt} \frac{\partial L(x(t_1))}{\partial\theta} = - \frac{\partial L(x(t_1))}{\partial x^\alpha(t)} \frac{\partial f^\beta}{\partial\theta}(t, x(t), \theta).$$

□

1.1 Calculation of $\partial L / \partial x$

1.2 Calculation of $\partial L / \partial\theta$

Suppose the model is layerized, the loss depends on the variables (inputs and model parameters) on the i th layer can be regarded as the loss of a **new** model by truncating the original at the i th layer, which we call $L_i(z_i)$. Varying θ will vary the $L_0(z_0)$ from two aspects, the effect from $dL_1/d\theta$ and the Δz_1 caused by $\Delta\theta$.

$$\frac{dL_0}{d\theta}(z_0) = \frac{dL_1}{d\theta}(z_1) + \frac{dL_1}{dz_1} \frac{\partial z_1}{\partial\theta}.$$

The same relation holds for any i , by simply considering a truncated model,

$$\frac{dL_i}{d\theta}(z_0) = \frac{dL_{i+1}}{d\theta}(z_{i+1}) + \frac{dL_{i+1}}{dz_{i+1}} \frac{\partial z_{i+1}}{\partial\theta}.$$

Thus we have, recursively,

$$\begin{aligned}\frac{dL_0}{d\theta}(z_0) &= \frac{dL_1}{d\theta}(z_1) + \frac{dL_1}{dz_1} \frac{\partial z_1}{\partial\theta} \\ &= \left[\frac{dL_2}{d\theta}(z_1) + \frac{dL_2}{dz_2} \frac{\partial z_2}{\partial\theta} \right] + \frac{dL_1}{dz_1} \frac{\partial z_1}{\partial\theta} \\ &= \frac{dL_2}{d\theta}(z_1) + \frac{dL_2}{dz_2} \frac{\partial z_2}{\partial\theta} + \frac{dL_1}{dz_1} \frac{\partial z_1}{\partial\theta} \\ &= \dots \\ &= \frac{dL_N}{d\theta}(z_1) + \sum_{i=1}^N \frac{dL_i}{dz_i} \frac{\partial z_i}{\partial\theta}.\end{aligned}$$

By $z_{i+1}(t) = z_i(t) + \epsilon f(z_i(t), t; \theta)$, $\partial z_i / \partial\theta = \epsilon \partial f(z_i, t; \theta) / \partial\theta$

2 Continuum of Hopfield

2.1 Hopfield Network

Consider Hopfield network. Let $x(t) \in \{-1, +1\}^N$ denotes the state of the network at discrete time $t = 0, 1, \dots$; and W a matrix on \mathbb{R}^N , essentially ensuring $W_{\alpha\beta} = W_{\beta\alpha}$ and $W_{\alpha\alpha} = 0$. Define energy $E_W(x(t)) := -W_{\alpha\beta} x^\alpha(t) x^\beta(t)$.

Theorem 2. Along dynamics $x_\alpha(t+1) = \text{sign}[W_{\alpha\beta} x^\beta(t)]$, $E_W(x(t+1)) - E_W(x(t)) \leq 0$.

Proof. Consider the async-updation of Hopfield network. Let's change the component at dimension $\hat{\alpha}$, i.e. $x'_{\hat{\alpha}} = \text{sign}[W_{\hat{\alpha}\beta} x^\beta]$, then

$$\begin{aligned}E_W(x') - E_W(x) &= -W_{\alpha\beta} x'^\alpha x'^\beta + W_{\alpha\beta} x^\alpha x^\beta \\ &= -2(x'^{\hat{\alpha}} - x^{\hat{\alpha}}) W_{\hat{\alpha}\beta} x^\beta,\end{aligned}$$

which employs conditions $W_{\alpha\beta} = W_{\beta\alpha}$ and $W_{\alpha\alpha} = 0$. Next, we prove that, combining with $x'_{\hat{\alpha}} = \text{sign}[W_{\hat{\alpha}\beta}x^{\beta}]$, this implies $E_W(x') - E_W(x) \leq 0$.

If $(x'^{\hat{\alpha}} - x^{\hat{\alpha}}) > 0$, then $x'^{\hat{\alpha}} = 1$ and $x^{\hat{\alpha}} = -1$. Since $x'_{\hat{\alpha}} = \text{sign}[W_{\hat{\alpha}\beta}x^{\beta}]$, $W_{\hat{\alpha}\beta}x^{\beta} > 0$. Then $E_W(x') - E_W(x) < 0$. Contrarily, if $(x'^{\hat{\alpha}} - x^{\hat{\alpha}}) < 0$, then $x'^{\hat{\alpha}} = -1$ and $x^{\hat{\alpha}} = 1$, implying $W_{\hat{\alpha}\beta}x^{\beta} < 0$. Also $E_W(x') - E_W(x) < 0$. Otherwise, $E_W(x') - E_W(x) = 0$. So, we conclude $E_W(x') - E_W(x) \leq 0$. \square

Since the states of the network are finite, the E_W is lower bounded. Thus the network converges (relaxes) at finite t .

2.2 Continuum

Let's consider applying the convergence of Hopfield network to neural ODE for generic network architecture. This makes the discrete time t a continuum.

Theorem 3. *Let M be a Riemann manifold with metric g . Given any function $\mathcal{E} \in C^1(M, \mathbb{R})$. Let $x(t) \in C^1(\mathbb{R}, M)$ denote trajectory. Then, $d\mathcal{E}/dt \leq 0$ along $x(t)$ if*

$$\frac{dx^{\alpha}}{dt}(t) = -\nabla^{\alpha}\mathcal{E}(x(t)).$$

Proof. We have

$$\frac{d\mathcal{E}}{dt}(t) = \nabla_{\alpha}\mathcal{E}(x(t))\frac{dx^{\alpha}}{dt}(t) = -\nabla_{\alpha}\mathcal{E}(x(t))\nabla^{\alpha}\mathcal{E}(x(t)) \leq 0. \quad \square$$

Further, if the function \mathcal{E} is lower bounded, then the trajectory converges (relaxes) at finite t . We call this dynamic with lower bounded \mathcal{E} as ‘‘Hopfield dynamic with energy \mathcal{E} ’’.

This is the continuum analogy to the convergence of Hopfield network. Indeed, let M be \mathbb{R}^N , and $\mathcal{E}(x) = -W_{\alpha\beta}x^{\alpha}x^{\beta}$ with $W_{\alpha\beta} = W_{\beta\alpha}$, then dynamics becomes

$$\frac{dx^{\alpha}}{dt}(t) = -\nabla_{\alpha}\mathcal{E}(x(t)) = -2W_{\alpha\beta}x^{\beta}(t),$$

which makes

$$\frac{d\mathcal{E}}{dt}(t) = -2\frac{dx^{\alpha}}{dt}(t)W_{\alpha\beta}x^{\beta}(t).$$

Comparing with the proof of convergence of Hopfield network, i.e. $\Delta E_W(x) = -2\Delta x^{\alpha}W_{\alpha\beta}x^{\beta}$, the analogy is obvious. The only differences are that the condition $W_{\alpha\alpha} = 0$ and the sign-function are absent here.

It is known that a Riemann manifold can be locally re-coordinated to be Euclidean. Thus, locally $\exists \hat{x}$ coordinate, s.t.

$$\frac{dx^{\alpha}}{dt}(t) = -\delta^{\alpha\beta}\frac{\partial\mathcal{E}}{\partial x^{\beta}}(x(t)).$$

2.3 General Form

In the proof of theorem 3,

$$\frac{d\mathcal{E}}{dt}(t) = \nabla_{\alpha}\mathcal{E}(x(t))\frac{dx^{\alpha}}{dt}(t).$$

We try to find the generic form of dx^{α}/dt that ensures $d\mathcal{E}/dt \leq 0$. To restrict the formation, symmetries are called for. Denote

$$\frac{dx^{\alpha}}{dt} = F^{\alpha}[\mathcal{E}](x),$$

where operator $F: C^\infty(M, M) \mapsto C(M, M)$.

Axiom 4. *Locality.*

This implies:

$$F[\mathcal{E}] = F(\mathcal{E}, \nabla \mathcal{E}, \nabla^2 \mathcal{E}, \dots);$$

Axiom 5. $\mathcal{E} \rightarrow \mathcal{E} + C$ for any constant C .

Combining with the previous, this then implies

$$F[\mathcal{E}] = F(\nabla \mathcal{E}, \nabla^2 \mathcal{E}, \dots).$$

Axiom 6. *Co-variance.*

This implies the balance of index. Thus

$$F^\alpha[\mathcal{E}] = c_1 \nabla^\alpha \mathcal{E} + c_3 \nabla^\alpha \mathcal{E} (\nabla^\beta \mathcal{E} \nabla_\beta \mathcal{E}) + c_3' \nabla^\alpha \mathcal{E} (\nabla^\beta \nabla_\beta \mathcal{E}) + \mathcal{O}(\nabla^5).$$

Axiom 7. For $x \rightarrow \lambda x$, $\exists k < m < M < K$ s.t. $m < F[\mathcal{E}](x) < M$ for any λ , where k and K are numerically finite. E.g. $k \sim 1$ and $K \sim 10$. This is essential for numerical stability, i.e. no under- and over-flow.

First, we have to notice a property of the feed forward neural network with rectified activations (e.g. ReLU, leaky ReLU, and linear).

Lemma 8. *Rectified activations are linearly homogeneous.*

Lemma 9. *If f and g are homogeneous with order λ_f and λ_g respectively, then $f \circ g$ is homogeneous with order $\lambda_f + \lambda_g$.*

Theorem 10. *Let $f_{\text{nn}}(x; \theta)$ a feed forward neural network with rectified activations, where θ represents the parameters (weights and biases). At the initial stage of training, $f_{\text{nn}}(\cdot; \theta)$ is linearly homogeneous. That is*

$$f_{\text{nn}}(\lambda x; \theta_{\text{ini}}) = \lambda f(x; \theta_{\text{ini}}).$$

Proof. Notice that $f_{\text{nn}}(\cdot; \theta)$ is linearly homogeneous when its biases vanish, and that biases are initialized as zeros. So $f_{\text{nn}}(\cdot; \theta)$ is linearly homogeneous at initial stage of training. \square

If \mathcal{E} is constructed by such neural network, $F[\mathcal{E}]$ can be further simplified. Indeed, if $\mathcal{E}(x; \theta) := \sqrt{f_\alpha(x; \theta) f^\alpha(x; \theta)}$, then $\mathcal{E}(\lambda x; \theta_{\text{ini}}) = \lambda \mathcal{E}(x; \theta_{\text{ini}})$, implying $F^\alpha[\mathcal{E}] = c_1 \nabla^\alpha \mathcal{E} + c_3 \nabla^\alpha \mathcal{E} (\nabla^\beta \mathcal{E} \nabla_\beta \mathcal{E}) + \mathcal{O}(\nabla^5)$, which scales as λ^0 .¹

Alternatively, if $\mathcal{E}(x; \theta) := f^2(x; \theta)$, then $\mathcal{E}(\lambda x; \theta_{\text{ini}}) = \lambda^2 \mathcal{E}(x; \theta_{\text{ini}})$. In this case, axiom 7 can never be satisfied.

1. Numerical experiment on MNIST dataset shows that this configuration indeed out-performs than others, like $\mathcal{E}(x; \theta) := f_\alpha(x; \theta) f^\alpha(x; \theta)$, $\mathcal{E}(x; \theta) := f^2(x; \theta)$, and non-Hopfield, e.t.c. In this experiment, $c_1 = 5$ and $c_{i>1} \equiv 0$; Nadam optimizer is employed, with standard parameters, except for $\epsilon = 10^{-3}$; the dimension of x is 64. For the details, c.f. the file `node/experiments/Hopfield.ipynb`.