Equivariant Neural Boundary Operators

1 Introduction

There has recently been a surge in deep-learning based methods for solving partial differential equations (PDE). In this context, a neural network is used to parameterize the PDE solution. Some research like physics-informed neural nets (PINN) focuses on designing unsupervised or semisupervised approaches, in which the PDE (or its weak form) is incorporated into the loss function.

Operator learning is a more recent direction that focuses on architecture design. The main philosophy is that the neural network parameterizes not the PDE solution, but the *solution operator*. The setting is typically quite general, but for the purpose of this work we will narrow it down to a certain type of problem:

Note 1. *Suppose a manifold $\mathcal{M}$ of codimension 1, embedded in $\mathbb{R}^d$. We will study a generic class of geometry-dependent operators $A[\,\cdot\,;\mathcal{M}]\colon \mathcal{H}_s(\mathcal{M})^d \to \mathcal{L}(\mathcal{M})^d$, where the equation*

$$A[x;\mathcal{M}] = y$$

has a unique solution $x \in \mathcal{H}_s(\mathcal{M})^d$ for any $y \in \mathcal{L}(\mathcal{M})^d$.

In particular, we will study the *Fredholm second-kind integral equations*:

$$A[x;\mathcal{M}](t) = x(t) - \int_{\mathcal{M}} k(t,s;\mathcal{M})x(s)\mathrm{d}s,$$

Where $k(\cdot,\cdot;\mathcal{M})\colon \mathcal{M}\times\mathcal{M} \to \mathrm{GL}(\mathbb{R}^d)$ is a *pairwise potential.* Typically, k depends on local first order properties of $\mathcal{M}$ like the normal direction. We refer to this class of operators as $\mathrm{FH2}(\mathcal{M},d)$. FH2-operators are often equivariant wrt. the euclidean group $\mathrm{E}(d)$, in the sense that

Definition 1. *A family of operatos $A[\cdot;\mathcal{M}]\colon \mathcal{H}_s(\mathcal{M})^d \to \mathcal{L}(\mathcal{M})^d$ is equivariant to $\mathrm{E}(d)$ if, for any $g=(r,h)\in\mathrm{E}(d)$ where r is a rotation and h a translation,*

$$A[gx; g\mathcal{M}] = gA[x;\mathcal{M}],$$

where $(gx)(t) = r(x(g^{-1}t))$ and $g\mathcal{M} = \{gt\colon t\in\mathcal{M}\}$.

A^{-1} is equivariant in the same way. Equivalently, A can be seen as a map between curves $\gamma = \{(t,x(t))\}_{t\in\mathcal{M}}$ of the fiber bundle $\mathcal{M}\times\mathbb{R}^d$. The group action $g\gamma$ is then g transforming the entire fiber bundle as $g(t,x) = (gt, rx)$, restricted to γ, and is a more natural interpretation of $A[g\gamma] = gA[\gamma]$.

2 Equivariant Neural Operators

The goal in this project is to develop a neural architecture that respects euclidean equivariance and can evaluate on *arbitrary* geometries $\mathcal{M}$. The data input (and output) for our architecture will be a generalized curve:

$$\gamma = \{(t, x(t)) \colon t \in \mathcal{M}\}.$$

we seek an operator Φ that takes any γ, so that $g\Phi[\gamma] = \Phi[g\gamma]$ for any $g \in \mathrm{E}(d)$.

Note 2. *Let $K_{\mathcal{M}}$ define integration with the kernel $k(\cdot, \cdot; \mathcal{M})$, and suppose $\|K_{\mathcal{M}}\| < 1$ for all $\mathcal{M}$. Then there is a trivial composition of computable integrals over $\mathcal{M}$, that can approximate the inverse to arbitrary precision:*

$$\Phi[y, \mathcal{M}] := \sum_{\ell=1}^{L} (-K_{\mathcal{M}})^{\ell} y$$

Moreover, the above operator can be written (by changing order of integration) in terms of an equivalent kernel $k_L(s, t; \mathcal{M})$:

$$k_L(t, s; \mathcal{M}) = \sum_{\ell=0}^{L} (-1)^{\ell} \int_{\mathcal{M}^{\ell}} k(t, u_1) k(u_1, u_2) \ldots k(u_{\ell-1}, s) \mathrm{d}u_1 \mathrm{d}u_2 \ldots \mathrm{d}u_{\ell-1}$$

The equivalent inverse kernel k_L cannot in general be written as a function of local properties of $\mathcal{M}$. Furthermore, it requires the evaluation of $L \sim \log(1/\epsilon)$ successive integrals to reach an error level of ϵ, which is often expensive since k decays slowly with respect to the distance $\|s - t\|$ between its arguments.

Suppose $\mathcal{D} = \{(\mathcal{M}_n, x_n, y_n)\}_{n=1}^{N}$ is a data set and $y_n = A[x_n; \mathcal{M}_n]$ for each n. Our objective is to learn the map $x_n = A^{-1}[y_n; \mathcal{M}_n]$. We seek a model on the form

$$\Phi[y; \mathcal{M}] = \sum_{n=1}^{N} \alpha_n(y; \mathcal{M}) x_n,$$

Gaussian Process Regression (GPR): Recall that for a Gaussian Process Y over x, with mean 0 and kernel q we have that $Y(x) \mid \boldsymbol{Y}$ where $\boldsymbol{Y} = Y(\boldsymbol{x})$ with $\boldsymbol{x} = (x_1, \ldots, x_N)$ is normal with mean

$$\mathbb{E}\left[Y(x) \mid \boldsymbol{Y}\right] = q(x; \boldsymbol{x})^T Q(\boldsymbol{x}; \boldsymbol{x})^{-1} \boldsymbol{Y},$$

where $Q(\boldsymbol{x}; \boldsymbol{x})$ is an $N \times N$ matrix with entries $q(x_m, x_n)$ and $q(x; \boldsymbol{x})$ is a vector with entries $q(x, x_n)$.

GPR for Hilbert Spaces: In the case where Y is a Gaussian process taking values in a Hilbert Space $\mathcal{Y}$, we have for $x, z \in X$ and $h, f \in Y$ that

$$\mathrm{Cov}(\langle f, Y(x) \rangle, \langle h, Y(z) \rangle) = q(x, z)[f, h]$$

if the spectrum of $q(x, y)$ decays fast, there is an ON-basis $\{\varphi_\ell\}_{\ell=1}^L \subset Y$ so that

$$q(x, z)[g, h] \approx q_L(x, z)[f, h] := \sum_{1 \le i, j \le L} q(x, z)[\varphi_i, \varphi_j] \langle f, \varphi_i \rangle \langle h, \varphi_j \rangle.$$

Let $\boldsymbol{Y}_L := (\langle \varphi_\ell, Y(x_n) \rangle)_{\ell, n}$. Then, $\langle \varphi_k, Y(x) \rangle \mid \boldsymbol{Y}_L$ is a normal with expectation

$$\mathbb{E}\left[\langle \varphi_k, Y \rangle \mid \boldsymbol{Y}_L\right] = \sum_{n=1}^{N} \sum_{\ell, \ell'=1}^{L} q(x, x_n)[\varphi_k, \varphi_\ell] Q^{-1}_{\ell, \ell', n, m} \langle \varphi_{\ell'}, Y(x_m) \rangle.$$

The matrix multiplication $Q^{-1}_{\ell, \ell', n, m} \langle \varphi_{\ell'}, Y(x_m) \rangle$ is computed at training.

Equivariant GPR: Let $N = 1$ for simplicity (one-shot learning). Let g^* be the adjoint of g with respect to $\langle \cdot, \cdot \rangle$. We now compute the conditional expectation of gY projected to the basis φ_ℓ. To simplify, we will initially denote $\overline{Y} = \sum_{\ell, \ell'} Q^{-1}_{\ell, \ell'} \langle \varphi_{\ell'}, Y(x_1) \rangle \varphi_\ell$. Then,:

$$\mathbb{E}\left[\langle \varphi_\ell, gY(x) \rangle \mid Y_1\right] = \mathbb{E}\left[\langle g^* \varphi_\ell, Y(x) \rangle \mid Y_1\right] = q(x, x_1)[g^* \varphi_\ell, \overline{Y}]$$

For equivariance, we must then have

$$\mathbb{E}\left[\langle \varphi_\ell, Y(gx) \rangle \mid Y_1\right] = q(gx, x_1)[\varphi_\ell, \overline{Y}] = q(x, x_1)\left[g^* \varphi_\ell, \overline{Y}\right].$$

In the case of $\mathrm{E}(d)$ with the $\mathcal{L}^2$-inner product, $g^* = g^{-1}$. Hence, a sufficient condition for input-output equivariance is that

$$q(gx, z)[u, v] = q(x, z)[g^{-1}u, v].$$

By symmetry of q, we infer $q(gx, z)[u, v] = q(x, z)[u, g^{-1}v]$. Suppose now that $q(x, z)[u, v] = \langle u, Q(x, z)v \rangle$ where $Q(x, z)$ is a linear operator. Then, the above (plus symmetry) translates to

$$\langle u, Q(gx, hz)v \rangle = \langle u, gQ(x, hz)v \rangle = \langle u, gQ(x, z)h^{-1}v \rangle,$$

for any $g, h \in \mathrm{E}(d)$, $x, z \in \mathcal{X}$ and $u, v \in \mathcal{Y}$. Hence, $Q(gx, hz) = gQ(x, z)h^{-1}$.

Equivariant Kernels: We give some examples of equivariant kernels.

Example 1. *Let d be a distance measure that satisfies $d(gx, z) = d(x, g^{-1}z)$ for all $g \in \mathrm{E}(d)$ and $x, y \in \mathcal{X}$. Define the shift $g(x, y)$ that minimizes this distance:*

$$\widehat{g}(x, y) = \mathrm{argmin}\{d(gx, y) \colon g \in \mathrm{E}(d)\}$$

and let $\widehat{d}(x, y) = d(\widehat{g}(x, y)x, y)$. Note that $\widehat{d}$ is invariant to $\mathrm{E}(d)$, since x is in $\mathrm{Orb}_{\mathrm{E}(d)}(gx)$. Moreover, $\widehat{g}(gx, y) = \widehat{g}(x, y)g^{-1}$, since

$$d(g(x, y)g^{-1}gx, y) = d(g(x, y)x, y) = \widehat{d}(x, y).$$

A similar argument shows that $\widehat{g}(x, hy) = h\widehat{g}(x, y)$. Hence, $\widehat{g}(gx, hy) = h\widehat{g}(x, y)g^{-1}$. Now, define $Q(x, y) := \phi(\widehat{d}(x, z))\widehat{g}(x, y)^{-1}$ for some decreasing function ϕ.

$$Q(gx, hy) = \widehat{g}(gx, hy)^{-1}\phi(\widehat{d}(gx, hy)) = g\left[\widehat{g}(x, y)\varphi(\widehat{d}(x, y))\right]h^{-1} = gQ(x, y)h^{-1}.$$

The correlation between vectors u and v associated with points x and y is obtained by finding gx in the orbit of x that best aligns to y, and then measuring the resulting alignment between gu and v. The final correlation is a rescaling of the alignment by a function of the distance between x and y times the alignment of u and v.

Example 2. *Alternatively, consider*

$$Q(x, y) = \widehat{g}(x, x_0)Q_I(x, y)\widehat{g}(x_0, y),$$

where x_0 is a reference template (for example $x_0 = x$), and Q_I is a linear operator that induces an invariant kernel, in the sense that.

$$\langle u, Q_I(gx, hy)v\rangle = \langle u, Q_I(x, y)v\rangle.$$

Equivariant Guassian Operator Process: Let ‘$definethe$

3 Data Generation:

For data generation we convert samples from random curves $\{(t_i, z_i)\}_{i=1}^{N}$ to canonical Fourier parameterization, by which we mean solving the following optimization problem:

$$\min_{c_k \in \mathbb{C}} \sum_{n=1}^{N} \left\| \sum_{k=-K}^{K} c_k e^{ikt_n} - z_n \right\|^2, \qquad \text{subject to} \qquad \left\| \sum_{k=-K}^{K} ikc_k e^{ikt} \right\| = 1 \quad \text{for all} \quad t$$