Data driven deep kernel learning: Combining mathematics with physics to explore deep learning

Soutrik Roy Chowdhury

Abstract

In this report we mention a framework to define a data driven deep kernel which plays pivotal role in many deep learning tasks. We aim to achieve our goal by using theories derived from representation theory, topology and physics.

Motivation

These days kernel methods are widely used. It facilitates number of learning tasks such as classification, generative models, in unsupervised learning (random forest, clustering etc.) and even in reinforcement learning (kernel based RL). However In most cases we pre-define the kernel. We don't explicitly construct a data driven kernel from scratch even in deep network which might give us many valuable information about the data. Only few literature are available trying to explain the behaviour of hidden layers in deep network through topology which creates problem while understanding a generative model and hence creating an explainable AI system.

Objective

We would like to define a **equivariant(steerable) deep kernel** using representation theory from scratch which fits in numerous available deep learning models. We also mention the drawback of the available framework in terms of CNN and point out the possible application of learning representations on Reproducing Kernel Hilbert Space (RKHS). We would apply the concepts from Topological Data Analysis (TDA) viz. new concepts like topological signature, density regularizer and probability distribution on manifolds which together with the concept of equivariant kernel defines a framework for **data driven deep kernel construction**.

Deep Equivariant Kernel Theory

Definition(Deep equivariant kernel): A continuous linear map $f : F_n \mapsto F_{n+1}$ can be written using a kernel k with kernel signature function from $\mathbb{R}^n \times \mathbb{R}^n$ to $\mathbb{R}^{K_{n+1} \times K_n}$ as follows:

$$[k \cdot f](x) = \int_{\mathbb{R}^n} k(x, y) f(y) dy \tag{1}$$

where F_n is the n-th feature space in a deep network. We now state a lemma without proof which is going to play a pivotal role in our kernel construction.
Lemma: The map $f \mapsto k \cdot f$ is equivariant if and only if for all $g \in SE(n)$

$$k(gx, gy) = \rho_2(r) k(x, y) \rho_1(r)^{-1} \tag{2}$$

Where ρ_1 and ρ_2 are irreducible representations of $SO(n)$. Note that $\mathbb{R}^n \cong SE(n)/SO(n)$ for $n \geq 3$. Which leads to the following theorem:
Theorem: A linear map $f : F_n \mapsto F_{n+1}$ is equivariant if and only if it is a cross correlation with a equivariant kernel.
Proof: The above lemma states that we can write k in terms of one-argument kernel since for $g = -x$, we have $k(x, y) = k(y - x)$. Substituting this in (4) we get

$$[k \cdot f](x) = \int_{\mathbb{R}^n} k(x, y) f(y) dy = \int_{\mathbb{R}^n} k(y - x) f(y) dy = [k * f](x)$$

A kernel satisfying $k(rx) = \rho_2(r)k(x)\rho_1(r)^{-1}$ (derived from (2)) can be called as *rotation-steerable*. (We would like to give such name after considering its action on $SO(3)$ and $SE(3)$.

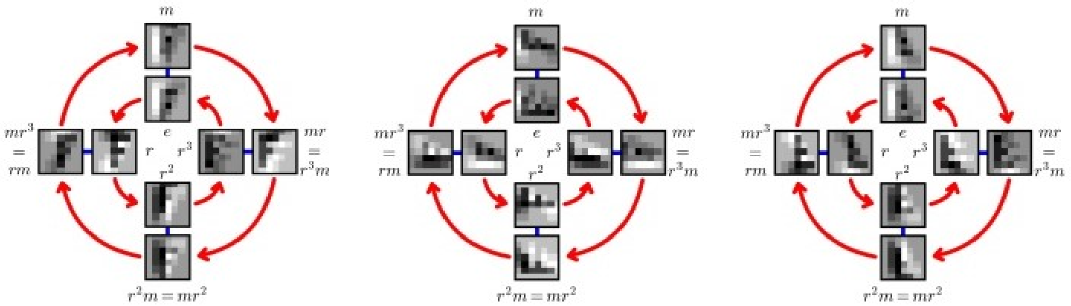


figure1: The action of group $p4m$ (Dihedral group D_4) on fibers. A fiber is considered as a finite dimensional vector space defined at each point of a channel (can be thought of as a filter bank) (Cohen, 2016)

Computing equivariant kernel basis

Our aim is to compute basis for equivariant kernel which satisfies (2) so that we can parametrize the kernel as a linear combination of basis kernels. We would demonstrate such computation with $SO(3)$ and $SE(3)$. From representation theory of finite groups we already have the irreducible representations $D^{l_i}(r)$ (our ρ's) of $SO(3)$. Representation $\rho_n(r)$ that acts on fiber in layer n is block diagonal where elements are given by $D^{l_i}(r)$. Implying kernel $k : \mathbb{R}^3 \mapsto \mathbb{R}^{K_n \times K_{n+1}}$ splits into blocks k^{jl} given by

$$k^{jl}(rx) = D^j(r)k^{jl}(x)D^l(r)^{-1} \tag{3}$$

again which can be written into vector-matrix form

$$vec(k^{jl}(rx)) = [D^j \otimes D^l](r)vec(k^{jl}(x)) \tag{4}$$

Facts:
1. Tensor product of representations itself is a representation and hence can be decomposed into irreducible representations.
2. For Irreducible representations D^j and D^l in $SO(3)$ of order j, l, $D^j \otimes D^l$ can be decomposed into $2min(j, l) + 1$ irreducible representations of order $|j - l| \leq J \leq j + l$.
3. We can find a basis change matrix Q of shape $(2l + 1)(2j + 1) \times (2l + 1)(2j + 1)$ such that $D^j \otimes D^l$ becomes block diagonal

$$[D^j \otimes D^l](r) = Q^T[\bigoplus_{J=|j-l|}^{j+l} D^J(r)]Q$$

which can be written as $\eta^{jl}(rx) = [\bigoplus_J D^J(r)]\eta^{jl}(x)$ after change of basis to $\eta^{jl}(x) = Q\ vec(k^{jl}(x))$. $\eta^{jl}(x)$ further splits into direct sum of $\eta^{jl,J}(x)$ where $\eta^{jl,J}(rx) = D^J(r)\eta^{jl,J}(x)$.
The solution is given by spherical harmonics $Y^J(x) = (Y^J_{-J}(x), \ldots, Y^J_J) \in \mathbb{R}^{2J+1}$. It tells us that the constraint only restricts the angular part of η_{jl} but leaves its radial part free. Therefore the solutions are given by spherical harmonics with some arbitrary continuous radial function $\varphi : \mathbb{R}^+ \mapsto \mathbb{R}$ as $\eta^{jl,J}(x) = \varphi(||x||)Y^J(x/||x||)$.
To choose the complete basis we define kernel basis functions as $\eta^{jl,Jm}(x) = \varphi^m(||x||)Y^J(x/||x||)$.
To obtain a basis $k^{jl,Jm}$ we map each $\eta^{jl,Jm}$ into original basis via Q^T.
We linearly combine the basis kernel $k^{jl} = \sum_{Jm} w^{jl,Jm}k^{jl,Jm}$ using learnable weights and stack them into a complete kernel k which is ready for its task (further fit in model).
Kernel space computation (algorithm):
Output: basis kernels $K^{jl,Jm}(x_i)$ sampled on a $s \times s \times s$ cubic grid of points $x_i \in \mathbb{Z}^3$.

- For each j and l we sample spherical harmonics Y^J in a radially independent manner in an array of shape $(2J + 1) \times s \times s \times s$.
- Transform spherical harmonics into original basis by multiplying Q^J.

- Unvectorize the resulting array into $unvec(Q^J Y^J(x_i))$ of shape $(2j+1)\times(2l+1)\times s\times s\times s$.
- Compute Q from the block-diagonal matrix relation.

There are total $C = \sum_{m=0}^{s}\sum_{J} 1 \leq (s+1)(2min(j,l)+1)$ basis kernel mapping between fields of order j and l and hence basis array of shape $C\times(2j+1)\times(2l+1)\times s\times s\times s$. Still this is computable!
Limitation of the theory:

1. We consider the theory on homogeneous space as in homogeneous space group action is more evident. We can go from $SO(3)$ to $SO(n)$ for any n or can take examples of other representations for both G and H but constrained by G/H as homogeneous space. More general (for any manifold, RKHS) theory is soughted.
2. Learning of representations itself is quite interesting i.e. which group representation to choose depending on the data.
3. We assume feature space as fields over continuous base space but while doing experiments in computer it usually involves discretization of the space (however this is negligible as its mainly aimed for mathematical elegance and simplicity).

To get a data driven framework we would like to propose a TDA based approach.
1. We understanding the data using topological signature (which gives us set of persistent diagram with topological nature to fit with the kernel so that the representation selection can be done online!). It is a very general 'birth-death' framework. We pick some interesting framework and mix our theory in it.
2. Also we would like to propose generative model based approach using geodesic learning under non-simply-connected manifold topology. (A kernel based density regularizer is aiming to get constructed to mitigate the topological difference between model and data).
Learning the latent space through geometric data interpolation: We would like to study geometric interpolation method based on geodesic curve on the learned data-manifold through density regularizer. There are flaws in available literature as there exists topological difference between the model and the dataset. Model defines family of smooth manifolds which are simply connected where dataset can be non-simply-connected. Density regularizer is estimating probability densities over holes (disconnected areas).
Normally the gradient descent method is used to compute geodesic interpolation. The density regularizer (Kim, 2019) is given by

$$L(z;\mu) = E(z) + \mu\int_a^b (-logp(z) + \frac{1}{2}log|det[g_z]|)dt$$

where $E(z)$ is the energy functional, μ is the regularization weight.
Kernel version similar to above can be proposed

$$L(z) = E(z) + \int_a^b (-logp(k(z,z_0)) + \frac{1}{2}log|det[g_z * k_z]|)dt$$

Here we get rid of the μ term (hoping to reach a non-parametric setting),z_0 landmark point corresponding to z and Riemannian metric g_z is element wise multiplication with k_z.

What we can achieve further:

1. We can explore the kernel learning theory in more general setting i.e. by understanding the representations of Lie algebras. As well as defining a suitable objective function is always a challenging goal. We would like to achieve it keeping in mind that we also at same time need to reduce computational complexity. We also would like to approach from other spaces (such as Krein spaces, Banach spaces etc.)
2. Recent progress in sampling techniques with help of optimal transport and algebraic geometry shows us some intriguing paths to explore deep generative model. There are some recent works on exploring sampling with algebraic varieties. We believe such architecture together with optimal

transport can help us in working on kerenl based generative model.
3. Such kernel based learning together with supra-Laplacians (defined Laplacians for multiplex network) can help us to study representation of states in a deep reinforcement learning architecture. There exists a work (Yifan,2019) on applying graph Laplacians in reinforcement learning and we would like to explore it for multi agent multi-task network by extending it with supra-Laplacians and with our kernel learning framework.

References

1. Thomas Hofmann, Bernhard Schölkopf, Alexander J.Smola, *Kernel methods in machine learning*, The Annals of Statistics, 2008, Vol. 36, No. 3, 1171-1220.
2. A.L.Carey, *Group representations in reproducing kernel hilbert spaces*, Reports on Mathematical Physics, vol. 14(1978), No. 2, 247-259.
3. Jiseob Kim, Byoung-Tak Zhang, *Data interpolations in deep generative models under Non-Simply-Connected Manifold Topology*, arXiv:1901.08553 2019.
4. Yifan Wu, George Tucker, Offir Nachum, *The Laplacian in RL: Learning repesentations with efficient approximations*, ICLR 2019.
5. T.S. Cohen, M. Welling, *Group Equivariant Convolutional Networks.* Proceedings of the International Conference on Machine Learning (ICML), 2016.
6. M. Weiler, W. Boomsma, M. Geiger, M. Welling, T.S. Cohen, *3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data*, Advances in Neural Information Processing Systems (NeurIPS), 2018.

Some demonstrations

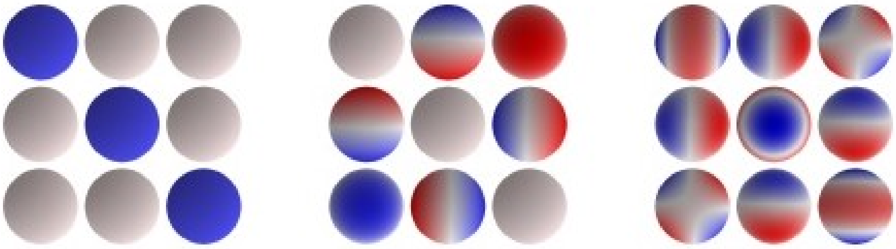
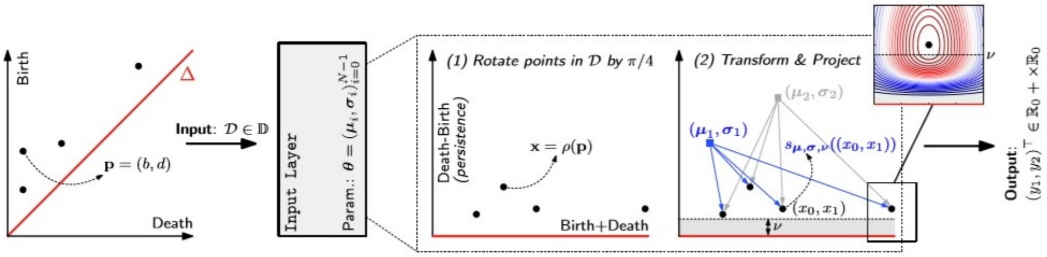


figure2: Angular part of the basis for the space of steerable kernels k^{jl} (take $j = l = 1$ i.e. $3D$ vector fields as input and output). We plot 3×3 matrices from left to right for $|j - l| \leq J \leq j + l$. Each 3×3 matrix provides information about one learnable parameter per radial basis function φ^m. The first one corresponds to identity, second one curl ($\nabla\wedge$) and the last gradient of divergence ($\nabla\nabla\cdot$).



Birth
Δ
$\mathbf{p} = (b, d)$
Death
Input: $\mathcal{D} \in \mathbb{D}$
Input Layer
Param.: $\boldsymbol{\theta} = (\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i)_{i=0}^{N-1}$
(1) Rotate points in $\mathcal{D}$ by $\pi/4$
Death-Birth (persistence)
$\mathbf{x} = \rho(\mathbf{p})$
Birth+Death
(2) Transform & Project
(μ_2, σ_2)
$(\boldsymbol{\mu}_1, \boldsymbol{\sigma}_1)$
$s_{\boldsymbol{\mu},\boldsymbol{\sigma},\nu}((x_0, x_1))$
(x_0, x_1)
ν
ν
Output: $(y_1, y_2)^\top \in \mathbb{R}_0 + \times \mathbb{R}_0$

figure3: Diagramtic illustration of the framework. After input layer is learnt we understand the nature of representation required from the barcode diagram (persistent diagram). In this illustration we see that groups of rotation, translation are the best option.

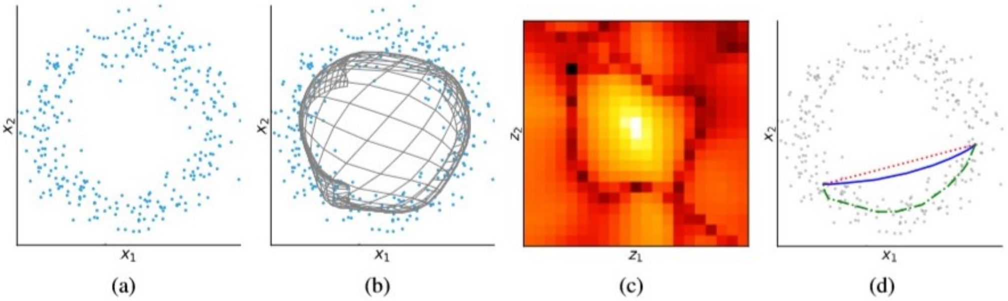


figure4: (a) 2D circular dataset (b) GAN trained on the dataset (c) probability density function learning (d) comparison of geodesic learning method: showing best result is obtained by combining geodesic interpolation method with density regularizer.

Implementations so far:

Classification with rotated MNIST data gives us a classification error 1.95 when we use group p4 and p4m. The learnt kernel is then feed into a CNN where the filters are also defined by equivariance (Cohen 2016). Classification result:
Group used: $p4$ and $p4m$
Network architecture: CNN architecture 7 layers of 3×3 convolutions (4×4 in final layer), no. of filters are given by dimension and multiplicity of irreducible representations of $p4$ which gives 20,20,20,20,16,16,8,8 no. of channels per layer.
Test error: 1.95
Better compared to other available results (Larochelle, 2007) which was around 8.5.
Classification result on CIFAR10 dataset:
Group used: Dihedral group (D_4)
Network architecture: ResNets (He, 2015,2016) convolutional network with 20 layer architecture.
Sample used for training: 2000
Test error (Equivariant network on ResNets): 3.65 (parameters used 9.1M) compared to ResNets 4.62 (parameter used 10.2M).
Group used: $\mathbb{Z}^2$, $p4$ and $p4m$.
Network architecture: All-CNN and ResNet44.
Test error: 9.44, 9.45 using $\mathbb{Z}^2$ on all-CNN and ResNet44 respectively; 7.59, 6.46 using $p4m$ on all-CNN and ResNet44 resp.
Group $p4$ composition of translation and rotation by $\pi/2$. Group $p4m$ translation, rotation by $\pi/2$ and mirror reflection.

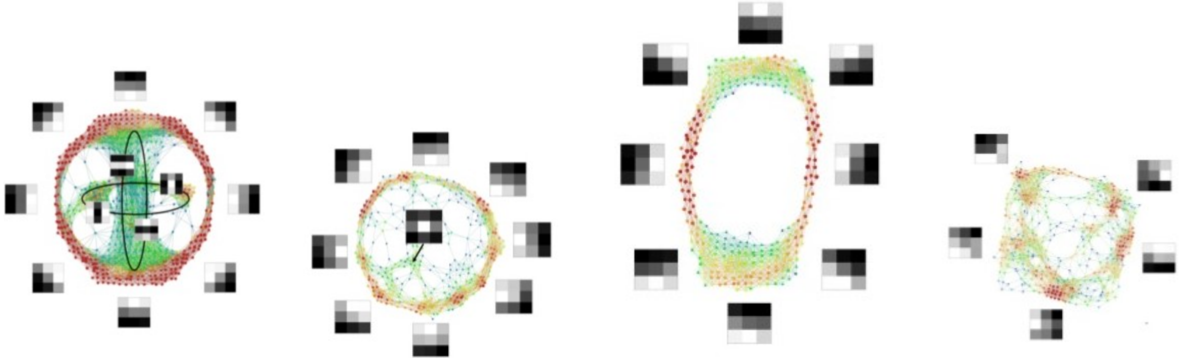


figure5: Persistent homology (layer 1 and 2) on CIFAR 10 dataset and rotated MNIST and possible information for group representation selection (Carlsson 2018).