Deep Lorentz Invariants for Particle Physics

A. Bogatskiy, T. Hoffman, D. W. Miller, J. T. Offermann
abogatskiy@flatironinstitute.org, {hoffmant, david.w.miller, jano}@uchicago.edu

Problem

Machine learning in physics faces a crisis of model complexity and interpretability. The primary way of limiting these issues while reaching great performance is by exploiting symmetries. CNNs, by virtue of their translational equivariance, greatly reduced model complexity and eased training. Similarly, permutation equivariance is the cornerstone of graph network methods. These ideas can be applied to other physical symmetries, such as the Lorentz symmetry and particle permutations, to tackle tasks such as jet tagging [1] and momentum reconstruction.

Lorentz Equivariance

If the inputs to our problem are a collection of particles' 4-momenta: $p_1^\mu, \ldots, p_N^\mu$, then a fundamental theorem from classical Invariant Theory lets us characterize Lorentz-invariant or equivariant observables of these particles. With permutation invariance necessarily demanded, all such observables are functions of only the pairwise dot products:

$$I(p_1, \ldots, p_N) = I\left(\{p_i \cdot p_j\}_{i,j}\right). \qquad (1)$$

Similarly, equivariant observables that output a Lorentz vector such as a 4-momentum can all be written as

$$E^\mu(p_1, \ldots, p_N) = \sum_k p_k^\mu \cdot I_k\left(p_1, \ldots, p_N\right), \qquad (2)$$

where each I_k is a Lorentz invariant as in (1).

Permutation Equivariance

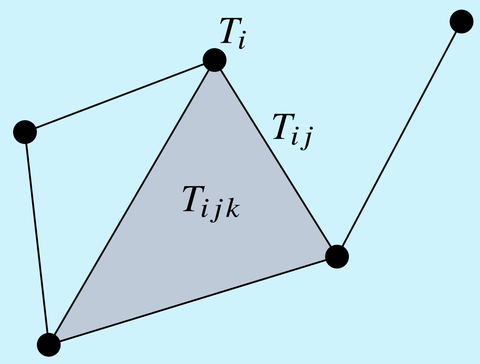
Permutation equivariance is a constraint on mappings between arrays $T_{i_1 i_2 \cdots i_r}$ of any rank r, every index $i_k \in \{1, \ldots, N\}$ referring to a particle label, whereby permutations of the particles "commute" with the map:

$$F\left(\pi \circ T_{i_1 i_2 \cdots i_r}\right) = \pi \circ F\left(T_{i_1 i_2 \cdots i_s}\right), \quad \pi \in S_N. \quad (3)$$

Graph Neural Networks explicitly implement this constraint for rank 1 arrays (node information). As a maximal generalization, a Message Passing layer can be defined as

$$\textbf{Message Passing:} \quad T^{(\ell+1)} = A \circ M\left(T^{(\ell)}\right).$$

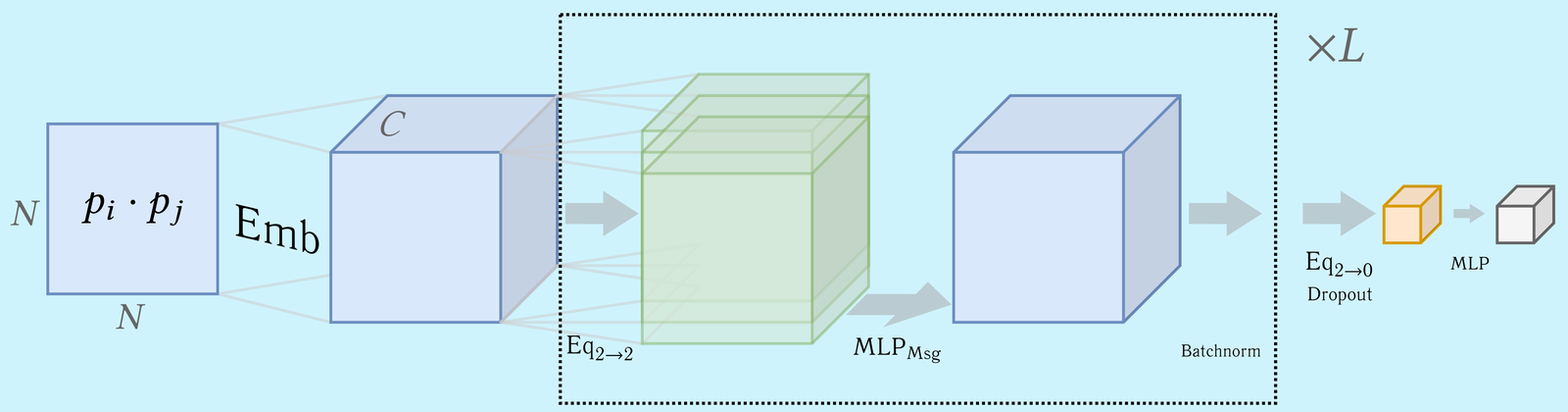
Here, M is a point-wise nonlinear map ("message forming") shared between all nodes, and A is a general permutation-equivariant linear mapping ("aggregation") acting on the particle indices of T.



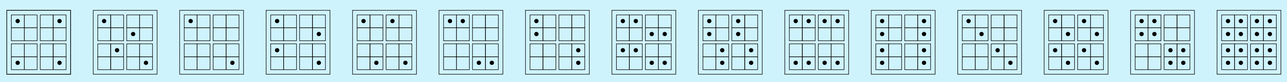
Depending on the task, the output layer will require an equivariant layer reducing the rank to 0 (e.g. classification) or 1 (e.g. to produce I_k in (2)).

PELICAN Architecture

The bases of permutation-equivariant linear maps between tensors of arbitrary ranks can be found in the related work [2]. For us the relevant cases are $2 \to 2$, $2 \to 1$ and $2 \to 0$, in which case the dimensionalities are $15, 4$, and 2, respectively. Some examples of the $2 \to 2$ maps include the identity, $T'_{ij} = \sum_k T_{kj}$, $T'_{ij} = \mathrm{diag}(T)$, and $T' = I \cdot \mathrm{tr}\, T$, all illustrated below. Here, summation can be replaced with any aggregation function (a symmetric function $\mathbb{R}^N \to \mathbb{R}$). Using the entire space of permutation-equivariant aggregations allows for optimal expressivity while stabilizing the training. Below is the architecture for a permutation-invariant task, but a permutation-equivariant architecture as in (2) can be obtained simply by replacing the $2 \to 0$ layer with a $2 \to 1$ one.

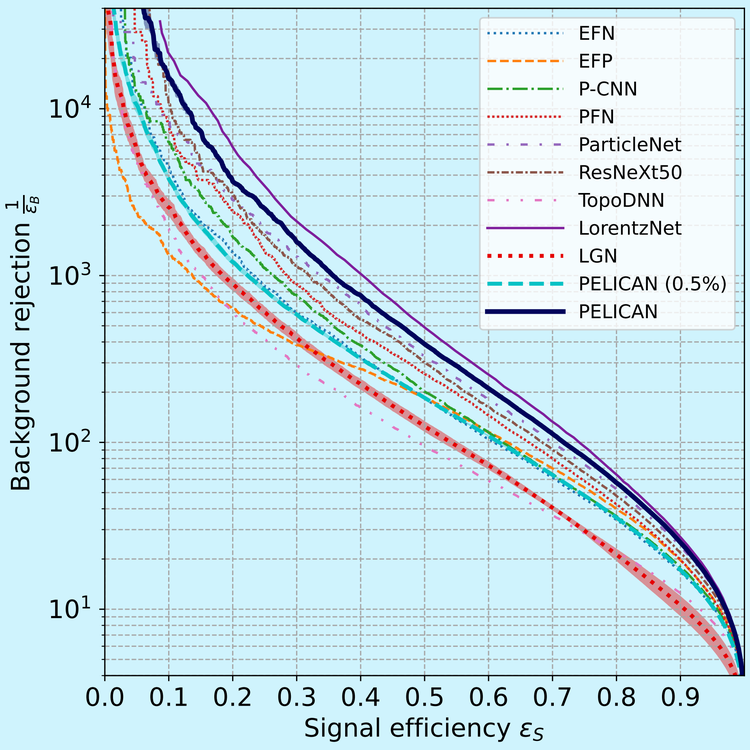


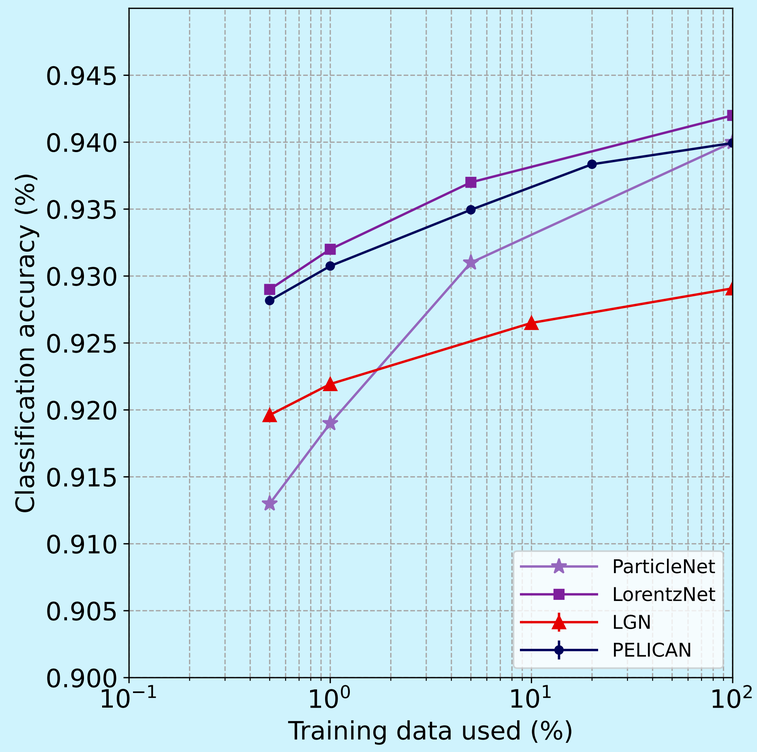
The 15 aggregators of $\mathrm{Eq}_{2\to 2}$ for $N = 2$, viewed as rank 4 binary tensors:



Jet Tagging

We benchmark the architecture on the popular task of top tagging, which is a binary classification task with an open dataset of 1.2M simulated events [1]. PELICAN bests all non-permutation-equivariant models and performs extremely well even when trained on only 0.5% of the training data.





Comparing top tagger performance [1, 3, 4], we see that the PELICAN architecture achieves competitive performance in terms of classification accuracy, area under the ROC curve (AUC) and background rejection, and exhibits good training sample efficiency.

Architecture	Accuracy	AUC	$1/\epsilon_B$ (@ $\epsilon_S = 0.3$)	#Parameters
EFN	0.927	0.979	633 ± 31	82k
EFP	0.932	0.980	384	1k
P-CNN	0.930	0.980	732 ± 24	348k
PFN	0.932	0.982	891 ± 18	82k
ParticleNet	0.938	0.985	1298 ± 46	498k
ResNeXt	0.936	0.984	1122 ± 47	1.46M
TopoDNN	0.916	0.972	295 ± 5	59k
LorentzNet	0.942	0.987	. 2195 ± 173	224k
LGN	0.929	0.964	424 ± 82	4.5k
PELICAN	0.931	0.981	695 ± 31	100k

References

[1] G. Kasieczka and T. Plehn, SciPost Phys. **7**, 014 (2019), arXiv:1902.09914 [hep-ph] .
[2] H. Pan and R. Kondor, in *AISTATS* (PMLR, 2022) pp. 5987–6001.
[3] A. Bogatskiy, B. Anderson, J. T. Offermann, M. Roussi, D. W. Miller, and R. Kondor (ICML, 2020) arXiv:2006.04780 .
[4] S. Gong, Q. Meng, Zhang, *et al.*, JHEP **07**, 030 (2022), arXiv:2201.08187 [hep-ph] .