GIVING ML A BOOST TOWARDS RESPECTING (APPROXIMATE) SYMMETRIES

Bay Area Particle Theory Seminar, Apr 2025

INBAR SAVORAY

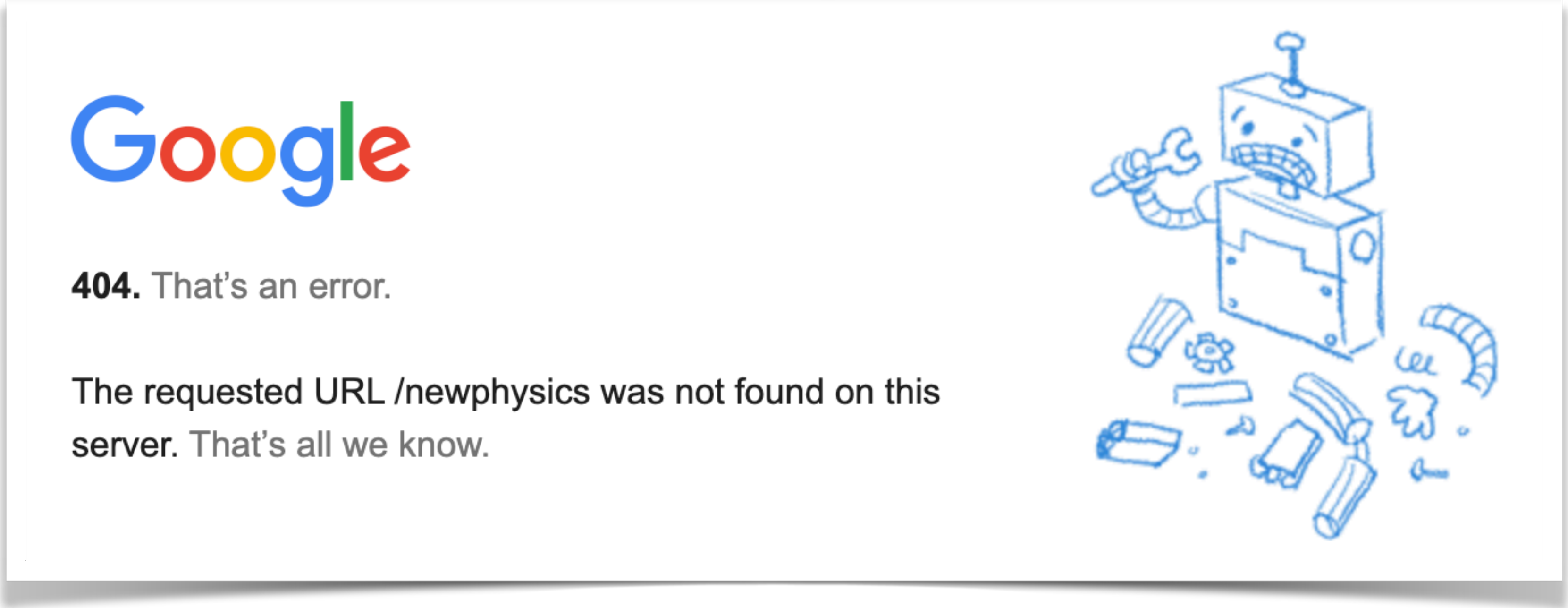
UC Berkeley & Lawrence Berkeley National Lab

WITH: PRADYUN HEBBAR, THANDIKIRE MADULA, VINICIUS MIKUNI,
BENJAMIN NACHMAN & NADAV OUTMEZGUINE

INTRODUCTION

PARTICLE PHYSICS 404

- Despite great theoretical and experimental effort, no evidence of New Physics has been found to date.
- Many dedicated searches ruled out a significant portion of the parameter space of theoretically motivated models.
- However, there is still much more to explore:
  - New theoretical models.
  - A lot of data.



Google

404. That's an error.

The requested URL /newphysics was not found on this server. That's all we know.

MACHINE LEARNING 101

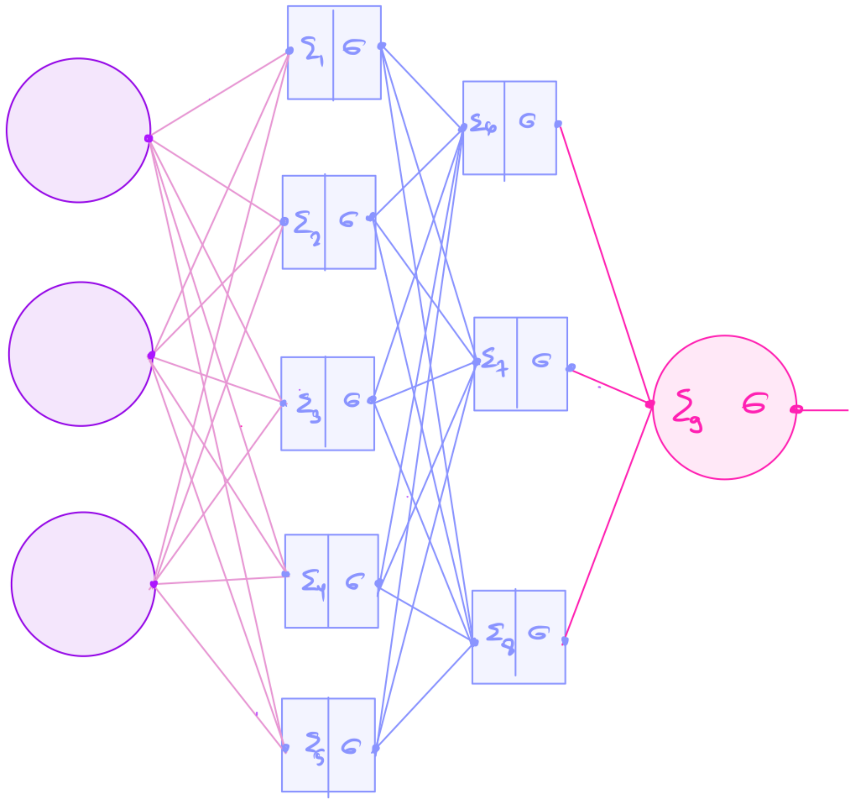
- A family of functions - **expressive, universal approximators**
  - Architecture - the specific family of functions

Flexible

MACHINE LEARNING 101

- A family of functions - **expressive**, **universal approximators**
  - Neural Network (NN) - sequence of linear and non-linear functions

input
("features")
x



hidden layers
θ_{NN}

output

$y\,(x)$

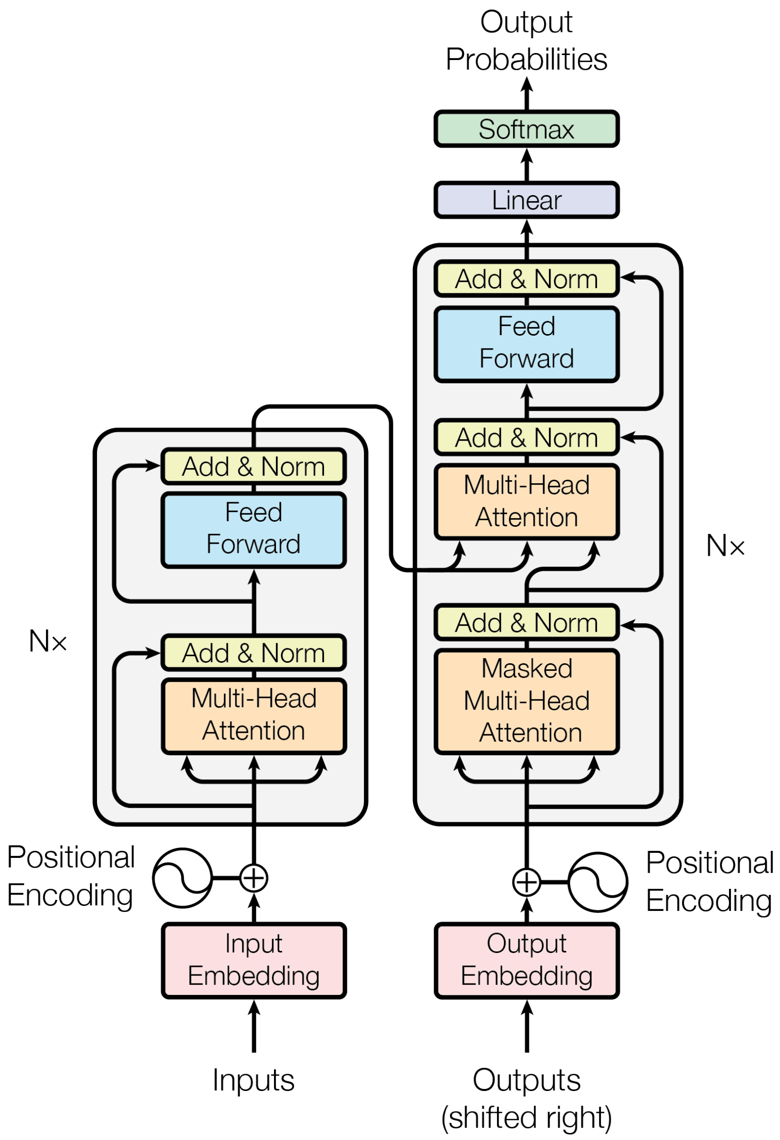
$$\Sigma_i = \overrightarrow{w}_i \cdot \vec{x}_{\mathrm{input}} + b_i$$

σ: non-linear activation

Flexible

MACHINE LEARNING 101

- A family of functions - expressive, universal approximators
  - Transformer ~ sequence of NN + attention (non-linear)

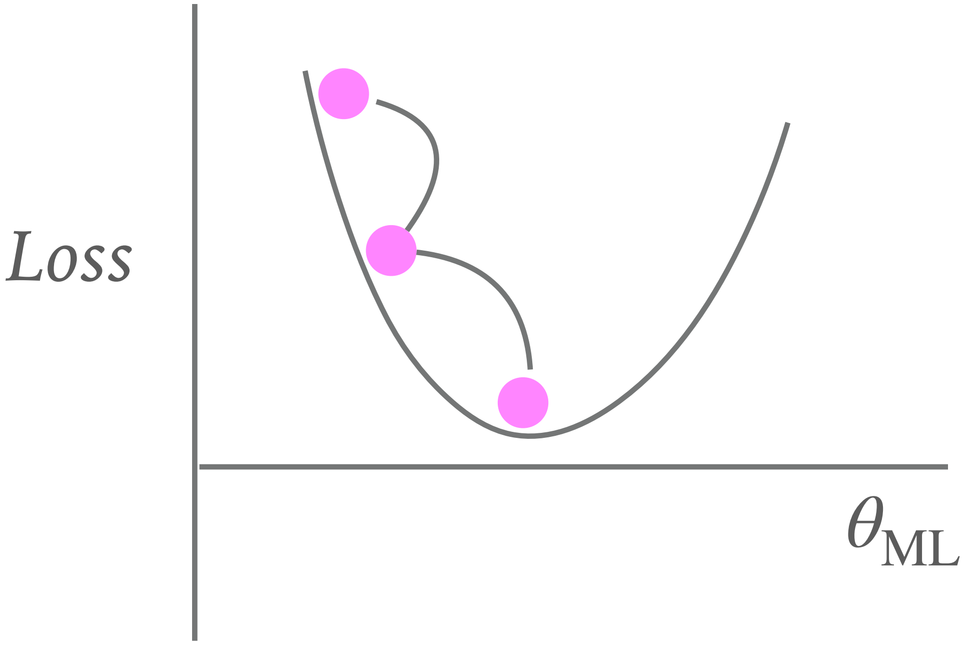


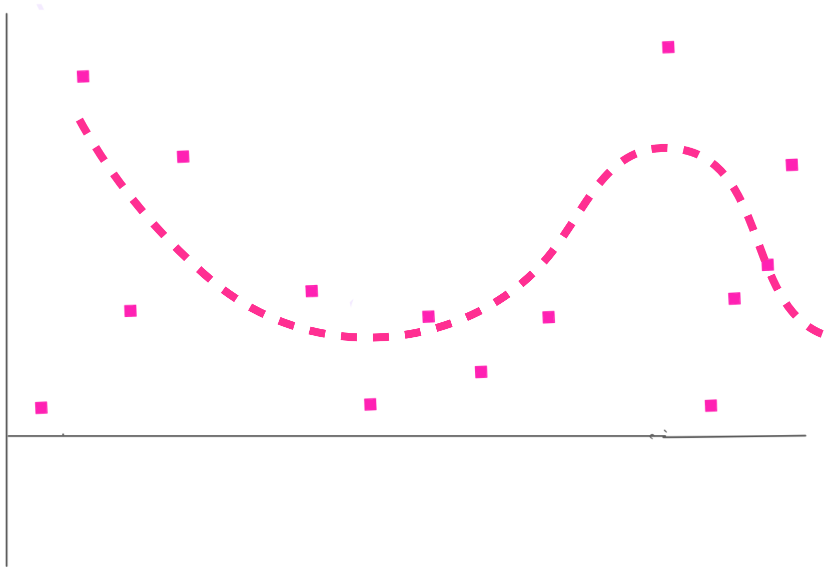
Flexible

A. Vaswani et al, [1706.03762]

MACHINE LEARNING 101

- A family of functions - expressive, universal approximators
- Learning - fit to data.
  - Training - parameters of function found by minimizing "loss" calculated on given dataset.

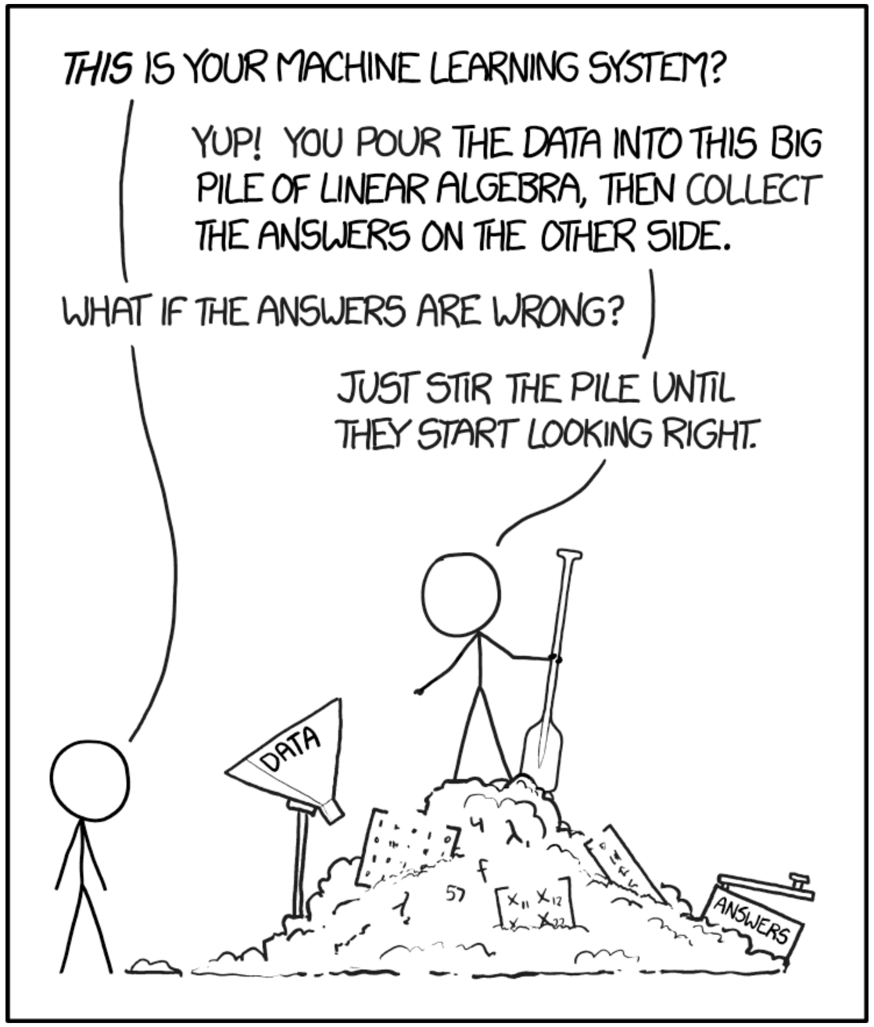




Great with a lot of data

MACHINE LEARNING 101

- A family of functions - expressive, universal approximators
  - Architecture - the specific family of functions (NN, CNN, GNN, transformer, etc.)
- **Learning** - **fit to data.**
  - Training - parameters of function found by minimizing "loss" calculated on given dataset.


THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS

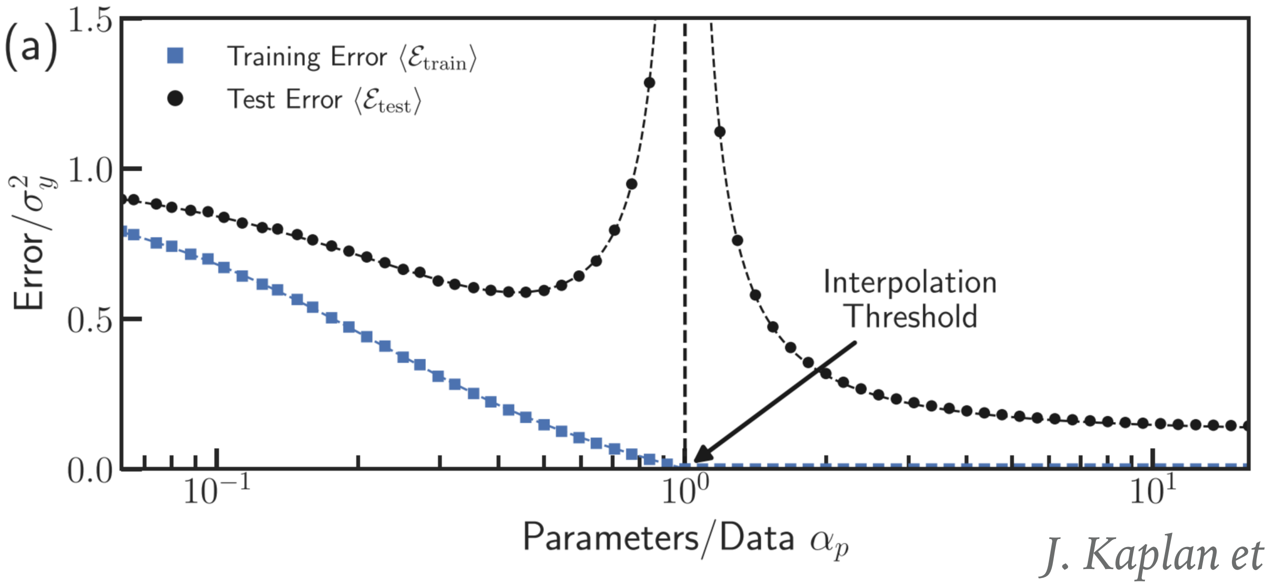
Great with a lot of data

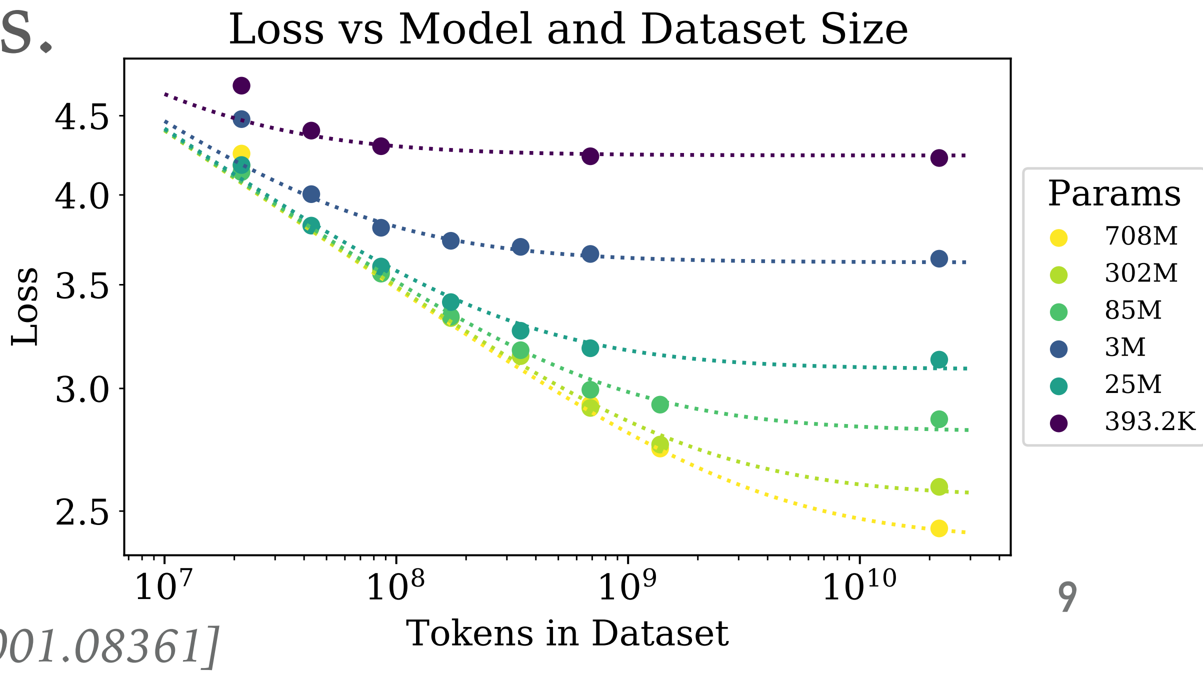
Flexible

xkcd.com

MORE DATA/PARAMETERS VS. MORE STRUCTURE

- Modern ML is "more is more" -
  - More data, more parameters, more compute.
  - Better capabilities, but also better generalization.
- Modern ML is less specialized -
  - Transformers perform well on a wide range of tasks.
  - Shift from carefully designing models for specific tasks to fine tuning foundational models.





J. Kaplan et al, [2001.08361]

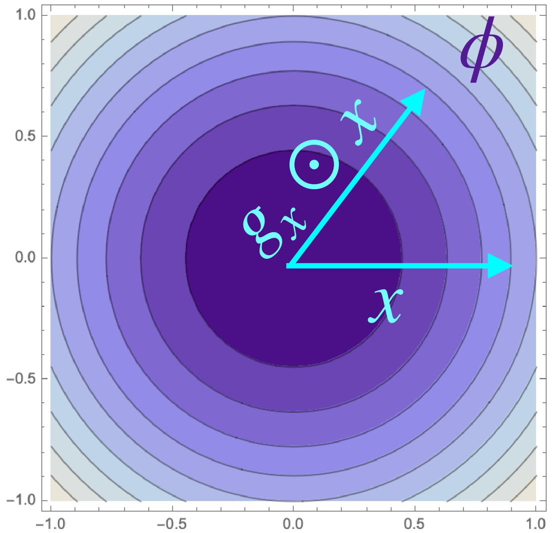
MORE DATA/PARAMETERS VS. MORE STRUCTURE

- On the other hand, more information is also more - especially for **scientific applications**
- **Data**
  - Noisy data can “trick” over-parameterized models
  - Might require more precision than language or images
- **Theory**
  - Often the underlying truth is “simple” - Ockham’s razor
  - We have **guiding theoretical principles** that can be easily phrased as clear mathematical/logical statements

Physical~structure

SYMMETRIES

- ➤ Symmetries as theoretical input - physical information about the system we are trying to describe



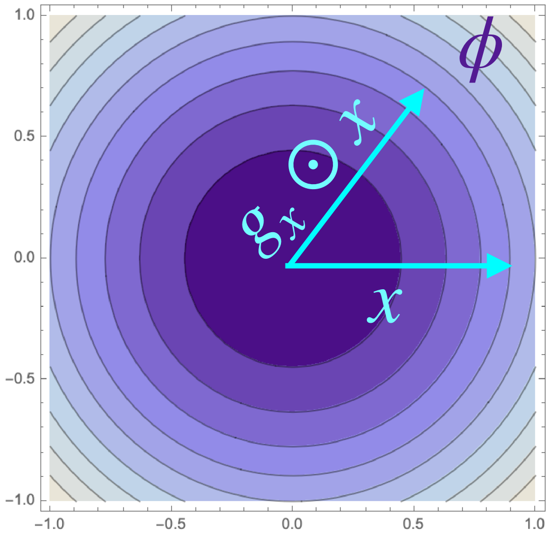
$$\phi\left(\underbrace{g_x}_{\textit{property of input data}} \odot x\right) = \underbrace{g_\phi}_{\textit{desired property of output}} \odot \phi(x) \qquad g \in G_{\text{symm}}$$

- ➤ Ubiquitous in particle physics - flavor, P/CP, rotations, translations…

- ➤ Often approximate - either theoretically or practically

IMPOSING SYMMETRIES

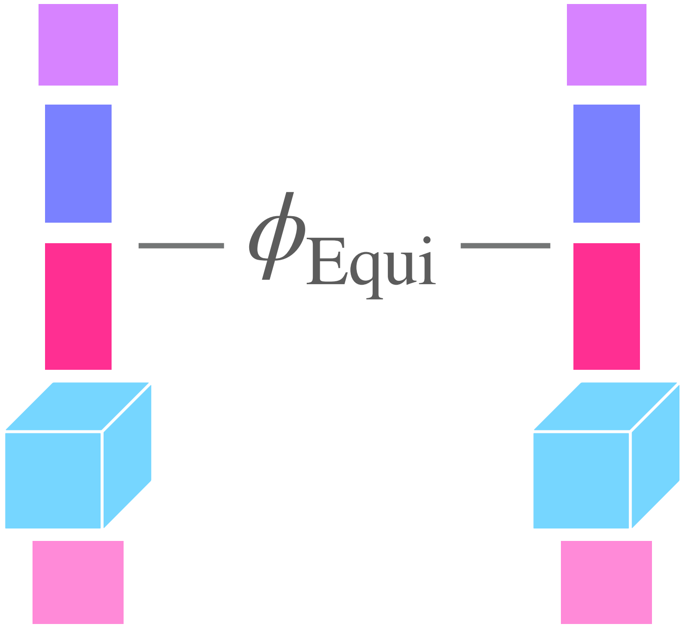
- Symmetric architecture - model can only output functions that transform in the correct way by construction.



$$\phi_{\mathrm{ML}}\left(g_x \odot x\right) = g_\phi \odot \phi_{\mathrm{ML}}(x)$$

property of input data

desired property of output



- Invariant - $g_\phi = 1$
- “Equivariant” - $g_\phi = g_x = g$ (covariant)

IMPOSING LORENTZ INVARIANCE

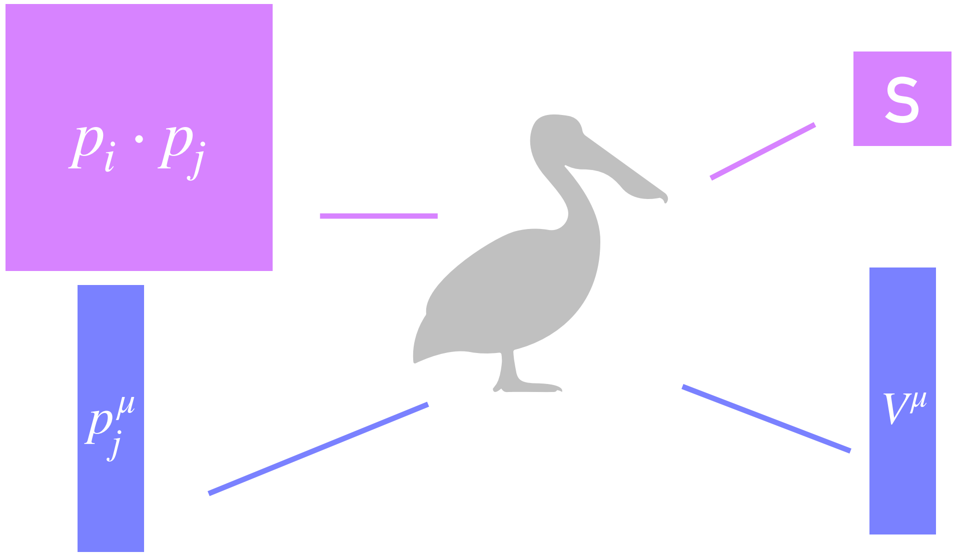
- Symmetric architecture - model can only output functions that transform in the correct way by construction.

$$\phi_{\mathrm{ML}}\left(g_x \odot x\right) = g_\phi \odot \phi_{\mathrm{ML}}(x)$$

- **Lorentz invariance** - theoretically exact, space-time symmetry, continuous and non-compact. $g = \Lambda\left(\vec{\beta}, \vec{\theta}\right)$

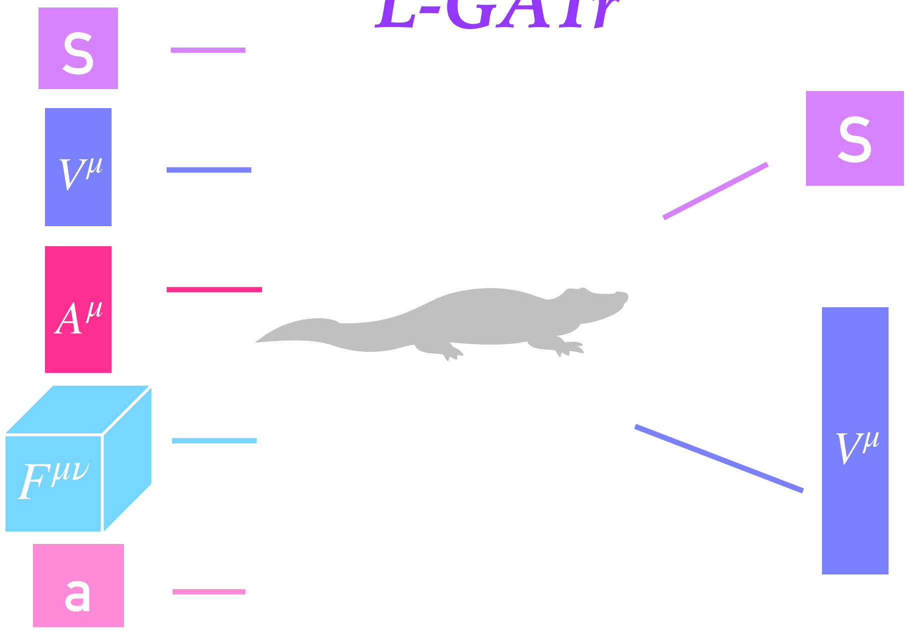
- Systematically build representations

PELICAN



A. Bogatskiy, T. Hoffman, D. W. Miller, J. T. Offermann, X. Liu, [2307.16506]

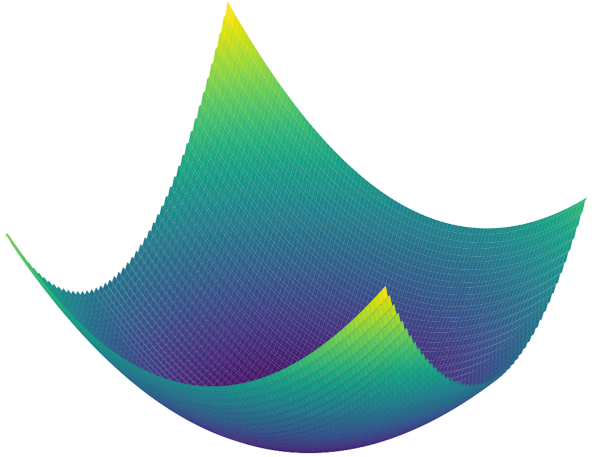
L-GATr



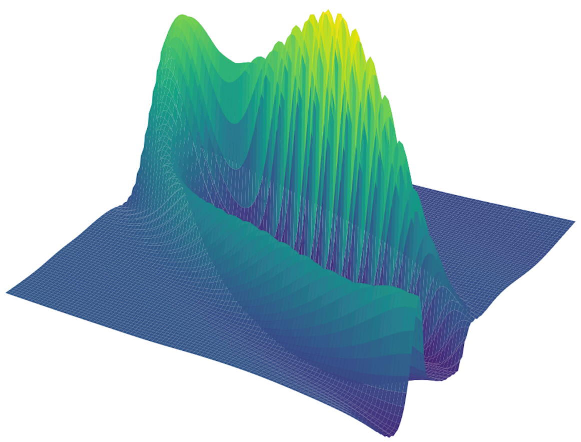
J. Spinner, V. Bresó, P. De Hann, T. Plehn, J. Thaler, J. Brehmer, [2405.14806]

CHALLENGES OF EQUIVARIANT MODELS

- ➤ Equivariant models have shown to improve performance on particle physics tasks.
- ➤ Expressivity could be challenging due to limited “building blocks”.
- ➤ Can be more compute intensive - overhead evaluation time and more FLOPs per parameter.
- ➤ Trainability - less smooth loss surface.

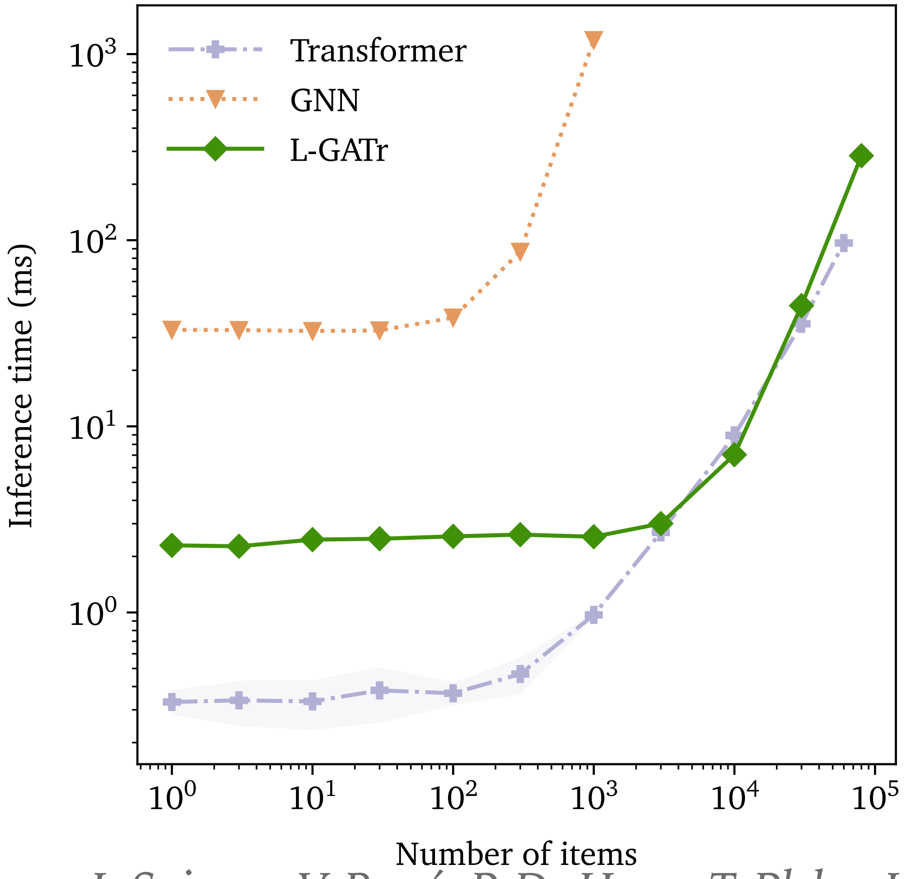


Transformer



GATr

A. Elhag, T. Rusch, F. Di Giovanni and M. Bronstein, [2410.17878]



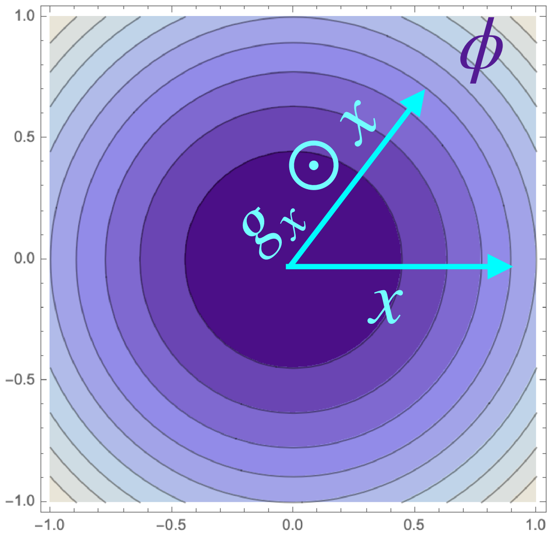
J. Spinner, V. Bresó, P. De Hann, T. Plehn, J. Thaler, J. Brehmer, [2405.14806]

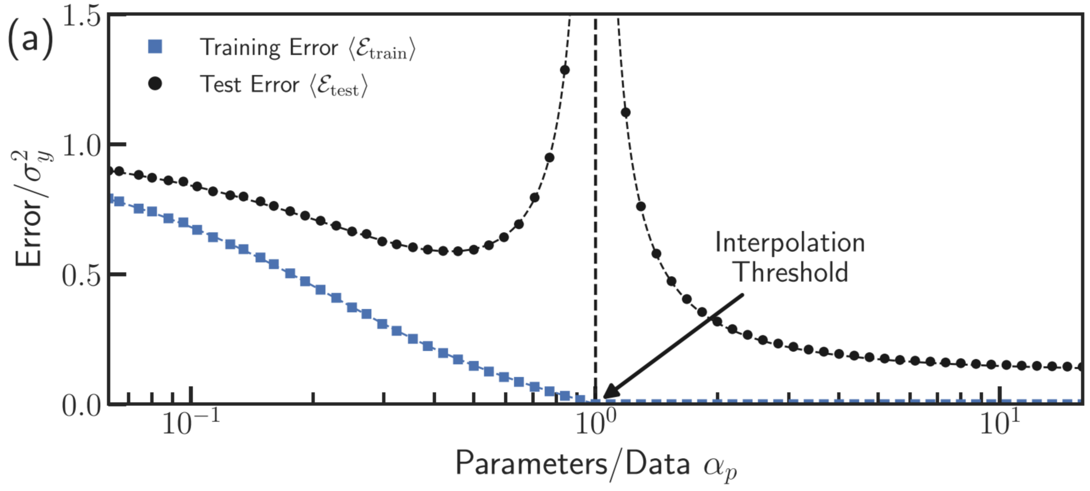
APPROXIMATE SYMMETRIES

- Often physical symmetries are only approximate.
- Although Lorentz invariance is exact, it is effectively broken if one only transforms the final state momenta
  - Beam - introduces a preferred direction
  - Detector - different energy efficiencies and spatial coverage/sensitivity.
  - Clustering - algorithm takes into account euclidean distances.

SOFT SYMMETRIES?

- We want flexible and easy to train models, that are aware of symmetries but can choose how to use that information.
- Instead of imposing symmetries, specify a preference towards respecting them.





Physical

Scalable

Flexible

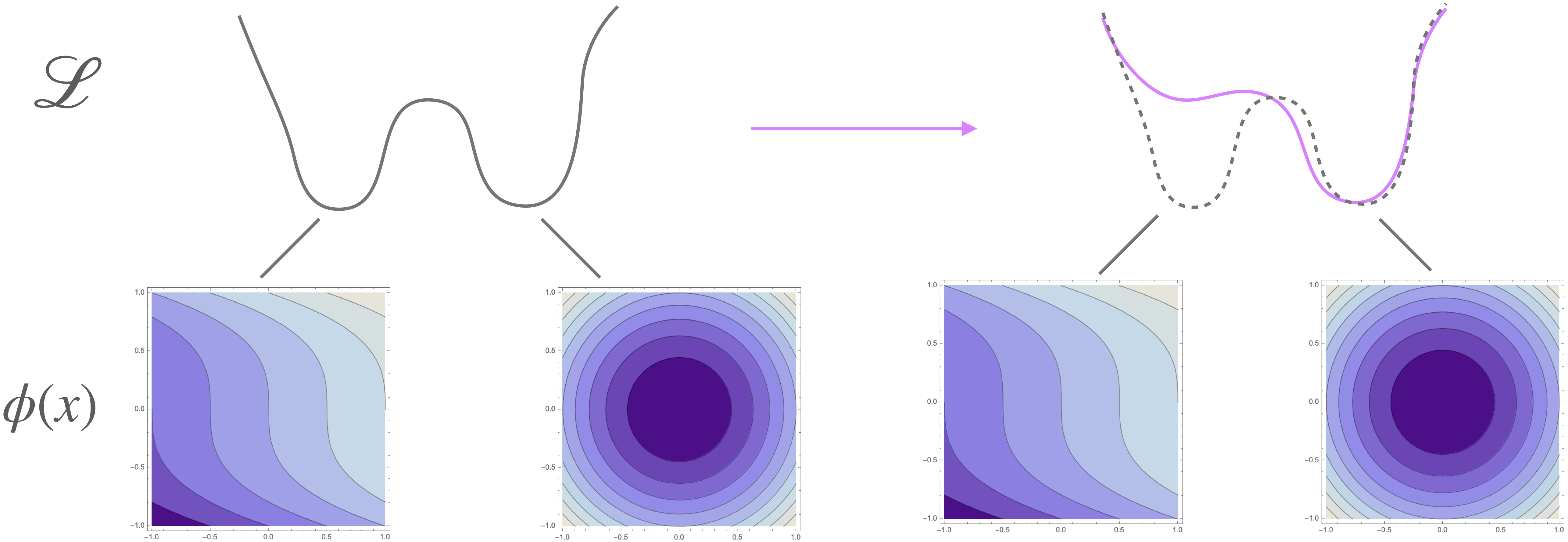
SYMMLOSS

SYMMLOSS

- A symmetry-encouraging term added to the loss

$$\mathscr{L} = \mathscr{L}_{\text{task}} + \lambda_{\text{symm}}\mathscr{L}_{\text{symm}}$$

$$\mathscr{L}_{\text{symm}} = \|\phi_{\text{ML}}\left(g_x \odot x\right) - g_\phi \odot \phi_{\text{ML}}(x)\|^2$$



SYMMLOSS

- A symmetry-encouraging term added to the loss

$$\mathscr{L} = \mathscr{L}_{\text{task}} + \lambda_{\text{symm}} \mathscr{L}_{\text{symm}}$$

$$\mathscr{L}_{\text{symm}} = \| \phi_{\text{ML}} \left(g_x \odot x \right) - g_\phi \odot \phi_{\text{ML}} (x) \|^2$$

- **Relax hard constraints** -
  - Allow for approximate symmetries (and even no symmetries at all).
  - Bias is tunable and controllable.
- **Flexible** - can be added to any model.

Scalable

Flexible

Physical

SYMMLOSS

- A symmetry-encouraging term added to the loss

$$\mathscr{L} = \mathscr{L}_{\text{task}} + \lambda_{\text{symm}} \mathscr{L}_{\text{symm}}$$

$$\mathscr{L}_{\text{symm}} = \| \phi_{\text{ML}} \left(g_x \odot x \right) - g_\phi \odot \phi_{\text{ML}} (x) \|^2$$

- $\mathscr{L}_{\text{symm}} \to 0$ if ϕ is in the desired representation for <u>any group element g</u> and <u>any input x</u>.
- In practice:
  - average over data
  - Group: *Gsymm - group sample* *δsymm - infinitesimal*

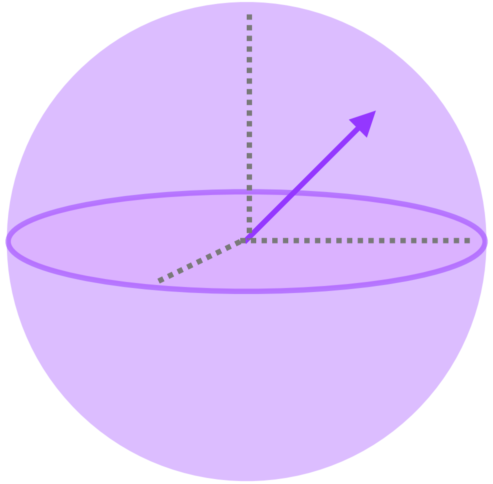
GSYMM

- Measure how different the output is on transformed inputs

$$\textit{Gsymm:}\quad \mathscr{L}_G = \frac{1}{N}\sum_{i=1}^{N} \left\| \phi_{ML}\left(g_i^x \odot x_i\right) - g_i^{\phi} \odot \phi_{ML}\left(x_i\right) \right\|^2$$

Sample $g_i \in G$

- sample from the group.
- cheap to calculate.



GSYMM - LORENTZ

- Measure how different the output is on transformed inputs

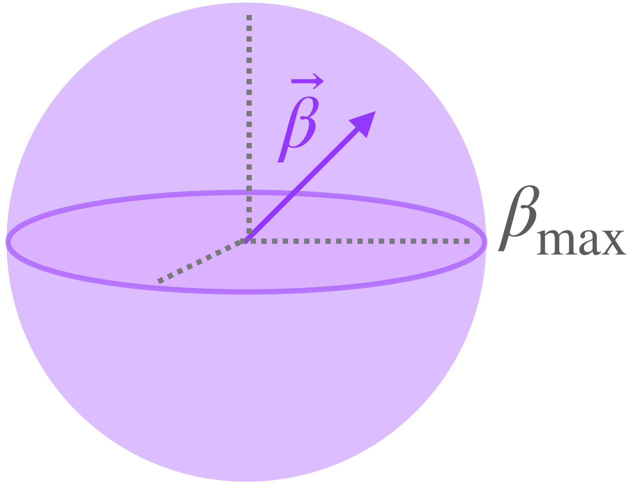
Gsymm: $$\mathscr{L}_G = \frac{1}{N}\sum_{i=1}^{N} \left\| \phi_{ML}\left(\Lambda\left(\vec{\beta}_i\right)\odot x_i\right) - g_i^{\phi}\odot\phi_{ML}\left(x_i\right)\right\|^2$$

Sample $g_i \in G$

scalar: $g_i^{\phi} = 1$

4-vector: $g_i^{\phi} = \Lambda\left(\vec{\beta}_i\right)$

- sample from the group.
- cheap to calculate.
- **Lorentz**: boost $\vec{\beta}$ uniformly sampled from a sphere of radius $\beta_{\max}$



δSYMM

- Infinitesimal transformations by generator L^a:

$$\delta^a \phi\left(x\right) = \partial_x \phi\left(x\right) \delta^a \vec{x}$$

δsymm:

$$\mathscr{L}_\delta = \left\| \sum_{j=1}^{\text{features}} \left(L_\phi^a\left(\phi\right) - L_x^a x_j \cdot \partial_{x_j} \phi \right) \right\|_{\text{gens, data}}^2$$

$$\delta^a \phi \qquad\qquad \delta^a \vec{x} = L_x^a \vec{x}$$

L_x in the rep. of x

δSYMM

- Infinitesimal transformations by generator L^a:

$$\delta^a \phi\left(x\right) = \partial_x \phi\left(x\right) \delta^a \vec{x}$$

δsymm:

$$\mathscr{L}_\delta = \left\| \sum_{j=1}^{\text{features}} \left(L^a_\phi\left(\phi\right) - L^a_x x_j \cdot \partial_{x_j} \phi \right) \right\|^2_{\text{gens, data}}$$

L_x in the rep. of x

- Is already approximate.
- No need to figure out sampling over group.
- On the other hand - computationally more expensive.

δSYMM - LORENTZ

- Infinitesimal transformations by generator L^a:

$$\delta^a \phi(x) = \partial_x \phi(x) \, \delta^a \vec{x}$$

δsymm:

$$\mathcal{L}_\delta = \left\| \sum_{j=1}^{\text{features}} \left(L^a_\phi(\phi) - L^a_x x_j \cdot \partial_{x_j} \phi \right) \right\|^2_{\text{gens, data}}$$

scalar: 0

4-vector: $L\phi$

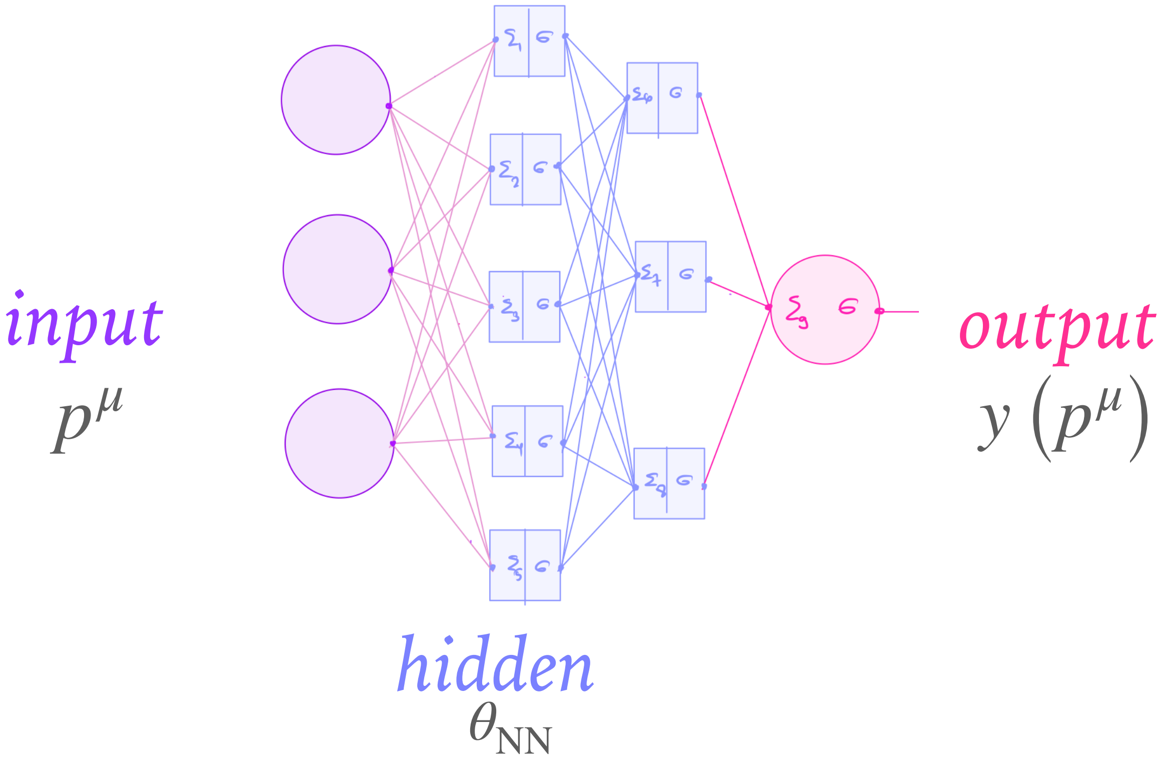
L_x in the rep. of x

- Is already approximate.
- No need to figure out sampling over group.
- On the other hand - computationally more expensive.
- **Lorentz**: 6 generators: $K_x, K_y, K_z, L_x, L_y, L_z$

EXPERIMENTS & RESULTS

TOY EXPERIMENTS

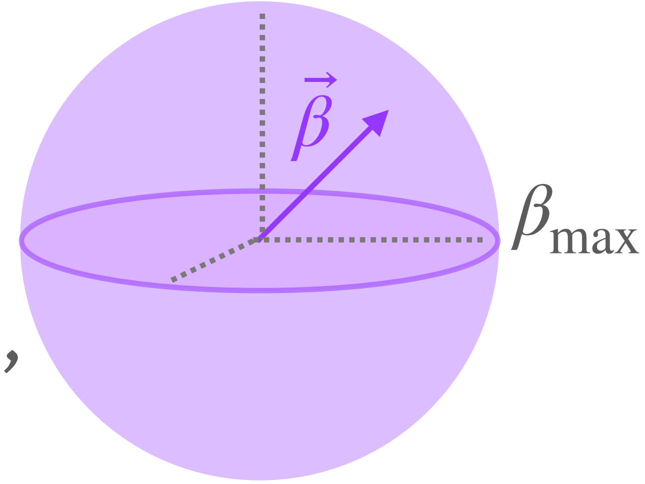
- Input: list of 4-momenta p_i^μ
- NN with 3 hidden layers of width 300, GeLU activation.
- **Exact Symmetry:** $f_{\text{truth}}\left(p_i^\mu\right) = \text{poly}\left(\mathrm{p_i} \cdot \mathrm{p_j}\right)$
- MSE loss - $\mathscr{L}_{\text{MSE}} = \sum_i \left| f_{\text{truth}}\left(p_i^\mu\right) - y_{\text{pred}}\left(p_i^\mu\right)\right|^2 \quad + \lambda_{\text{symm}}\mathscr{L}_{\text{symm}}$

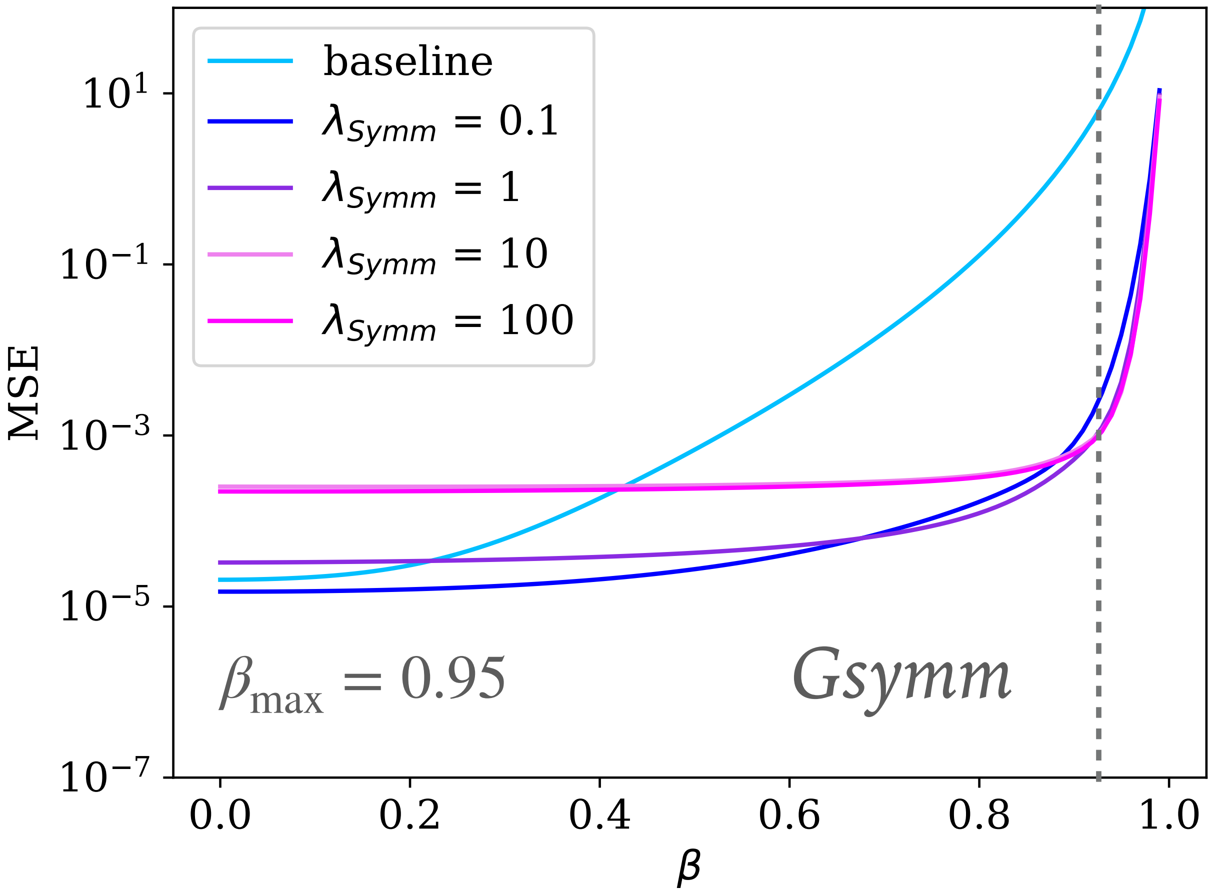


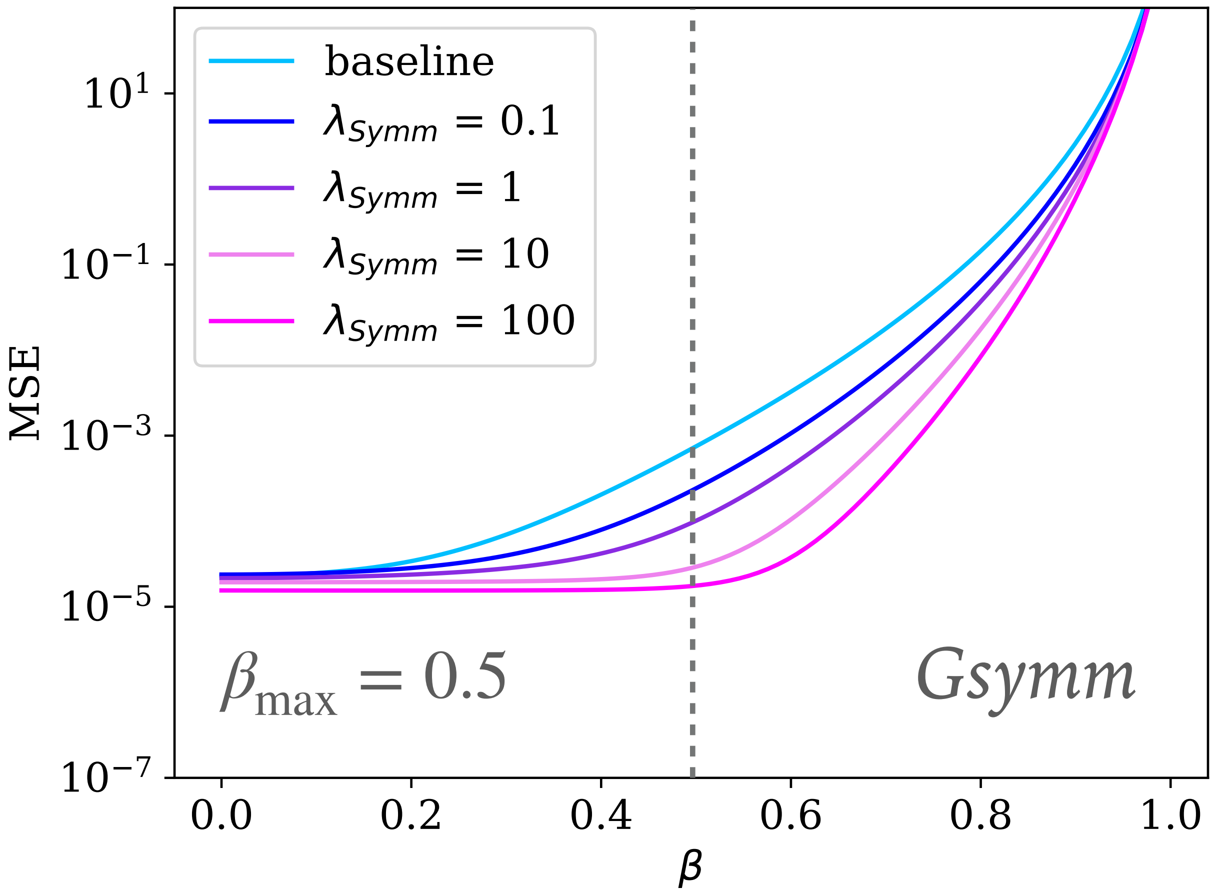
TOYS - EXACT SYMMETRY

$$\mathscr{L} = \mathscr{L}_{\text{MSE}} + \lambda_{\text{symm}}\mathscr{L}_G \qquad \mathscr{L}_G = \left\| \phi_{ML}\left(B_i\left(x_i\right)\right) - \phi_{ML}\left(x_i\right) \right\|^2$$

- ➤ Gsymm can achieve better performance than baseline on boosted inputs.
- ➤ Larger training β_{max} - flatter as function of boost, but can under-perform for small boosts.





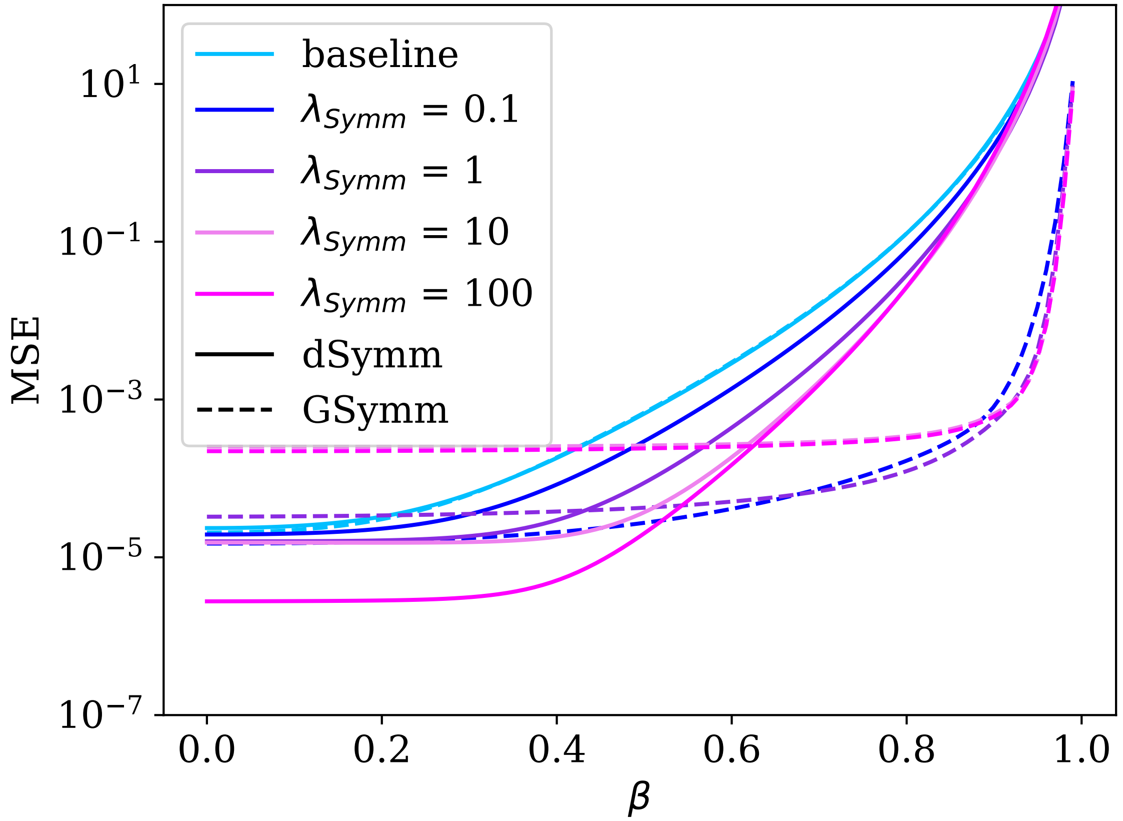


TOYS - EXACT SYMMETRY

$$\mathscr{L}_\delta = \left\| \frac{\partial \phi_{ML}}{\partial p_\mu} \cdot \left(L_{\mu\nu} p_i^\nu \right) \right\|^2 \qquad \mathscr{L}_G = \left\| \phi_{ML}\left(B_i\left(x_i\right)\right) - \phi_{ML}\left(x_i\right) \right\|^2$$

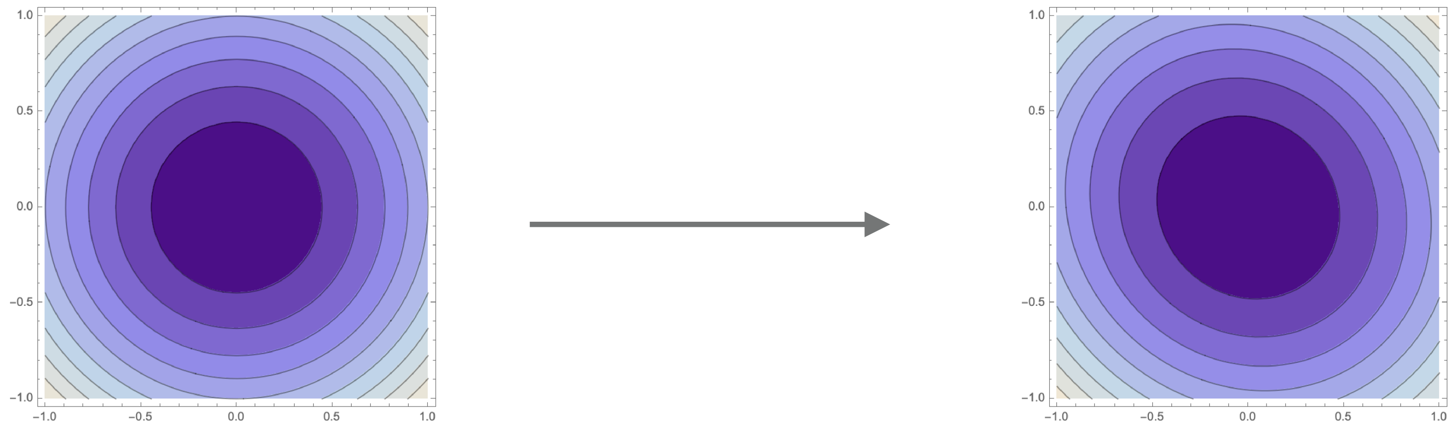
- Even infinitesimal loss achieves better performance than baseline, and can extend to non-infinitesimal boosts!
- δsymm better at smaller β.
- Big λ doesn't hurt for small transformations.

$$\mathscr{L} = \mathscr{L}_{\text{MSE}} + \lambda_{\text{symm}} \mathscr{L}_{\text{symm}}$$



TOY EXPERIMENTS

- Input: list of 4-momenta p_i^μ
- NN with 3 hidden layers of width 300, GeLU activation.
- **Exact Symmetry:** $f_{\text{truth}}\left(p_i^\mu\right) = \text{poly}\left(\mathrm{p_i}\cdot\mathrm{p_j}\right)$
- **Approximate Symmetry:** $f_{\text{truth}}\left(p_i^\mu\right) = \text{poly}\left(\mathrm{p_i}\cdot\mathrm{p_j}, \mathrm{p_i}\cdot\mathrm{s}\right)$
  - "Spurion" $s = \begin{pmatrix} 0 & 0 & 0 & 10^{-3} \end{pmatrix}$



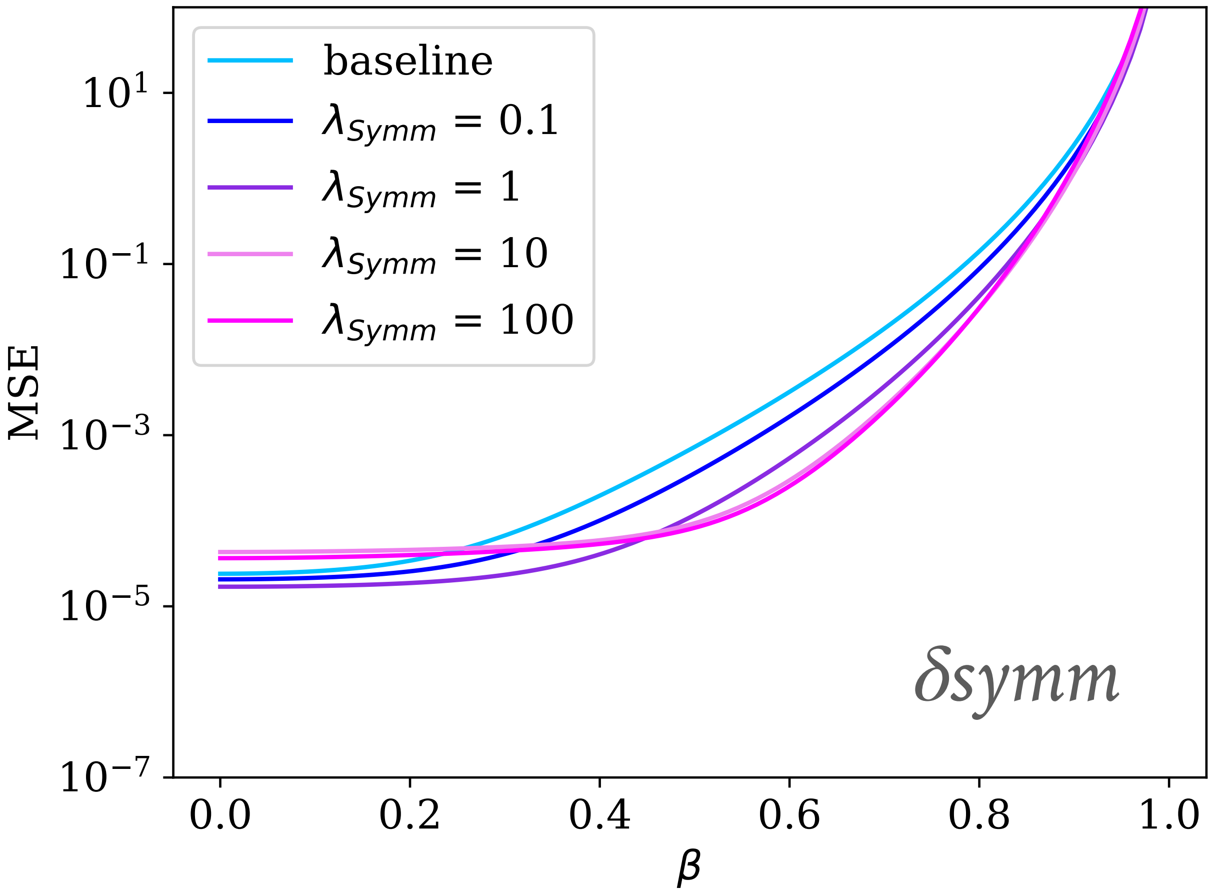
TOYS – APPROXIMATE SYMMETRY

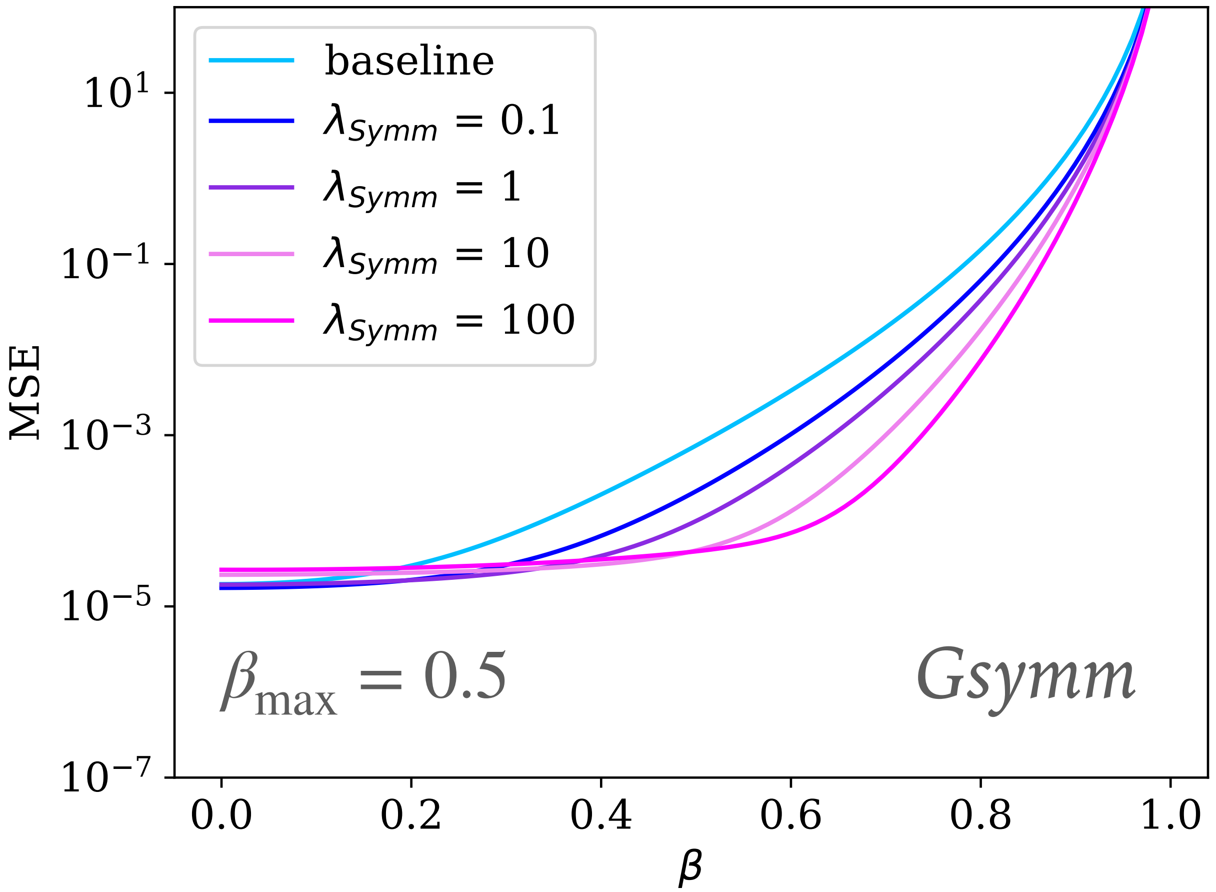
$$\mathscr{L} = \mathscr{L}_{\mathrm{MSE}} + \lambda_{\mathrm{symm}} \mathscr{L}_{\mathrm{symm}}$$

- Gain even when the symmetry is not exact.

$$\mathscr{L}_{\delta} = \left\| \frac{\partial \phi_{ML}}{\partial p_{\mu}} \cdot \left(L_{\mu\nu} p_i^{\nu} \right) \right\|^2$$

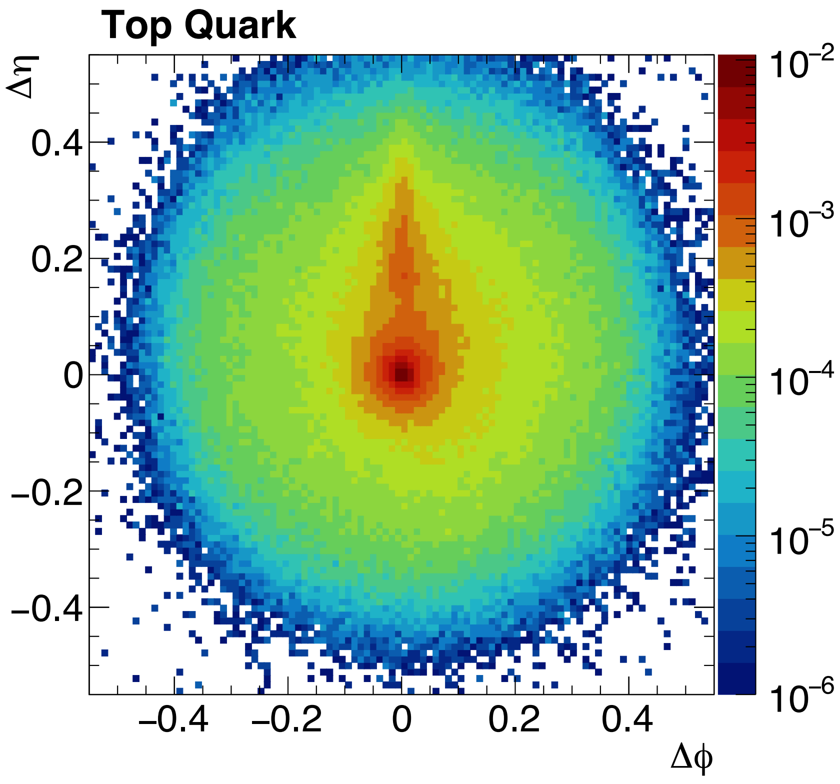
$$\mathscr{L}_{G} = \left\| \phi_{ML}\left(B_i\left(x_i\right)\right) - \phi_{ML}\left(x_i\right) \right\|^2$$

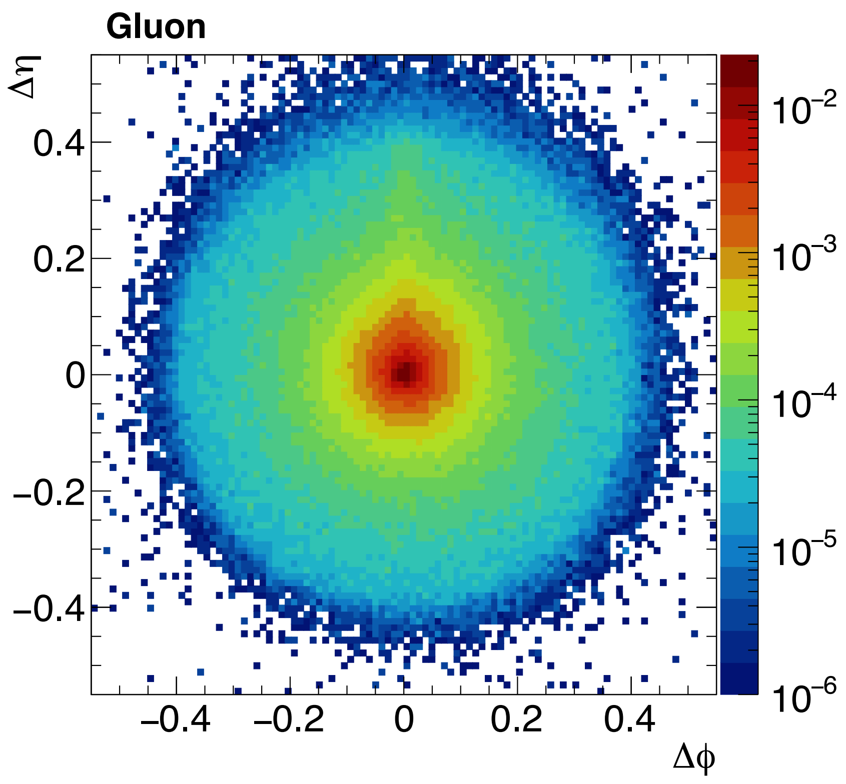




TOP TAGGING

- Physics example - QCD vs. top-jets
  - Precision measurements
  - BSM studies
- Goal - learn $p(x\,|\,\text{top})$ vs. $p(x\,|\,\text{QCD}) \rightarrow$ classify jet.



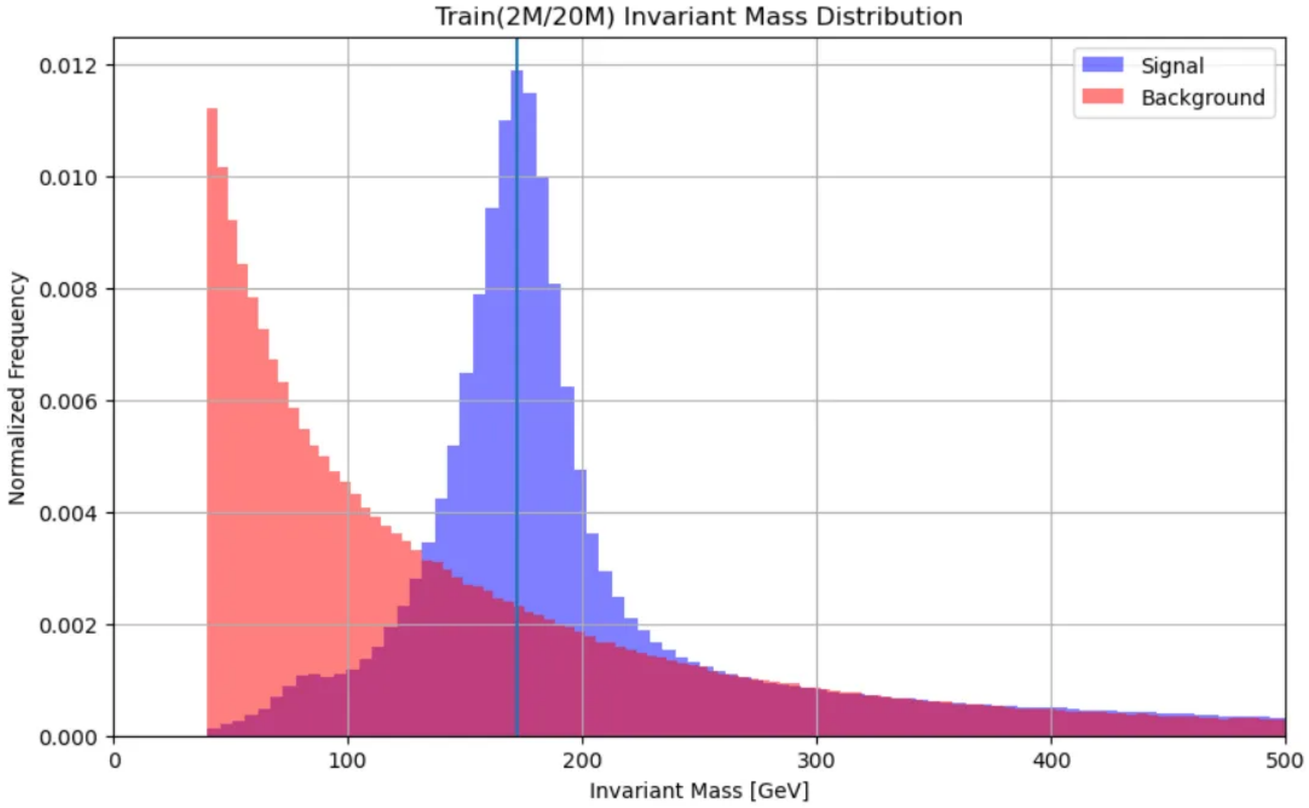


V. Mikuni, F. Canelli,[2102.05073]

TOP TAGGING - DATASET

ATLAS collaboration (2022),
https://opendata.cern.ch/record/15013

- ATLAS top tagging dataset
- Most realistic dataset
  - Full LHC Run-2 conditions (including pile-up)
  - Full detector simulation
  - Event reconstruction



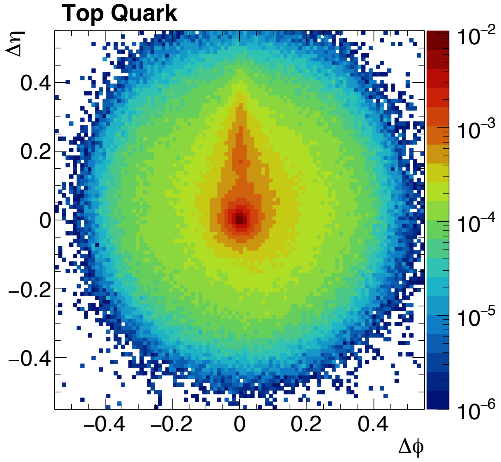
TOP TAGGING - MODEL

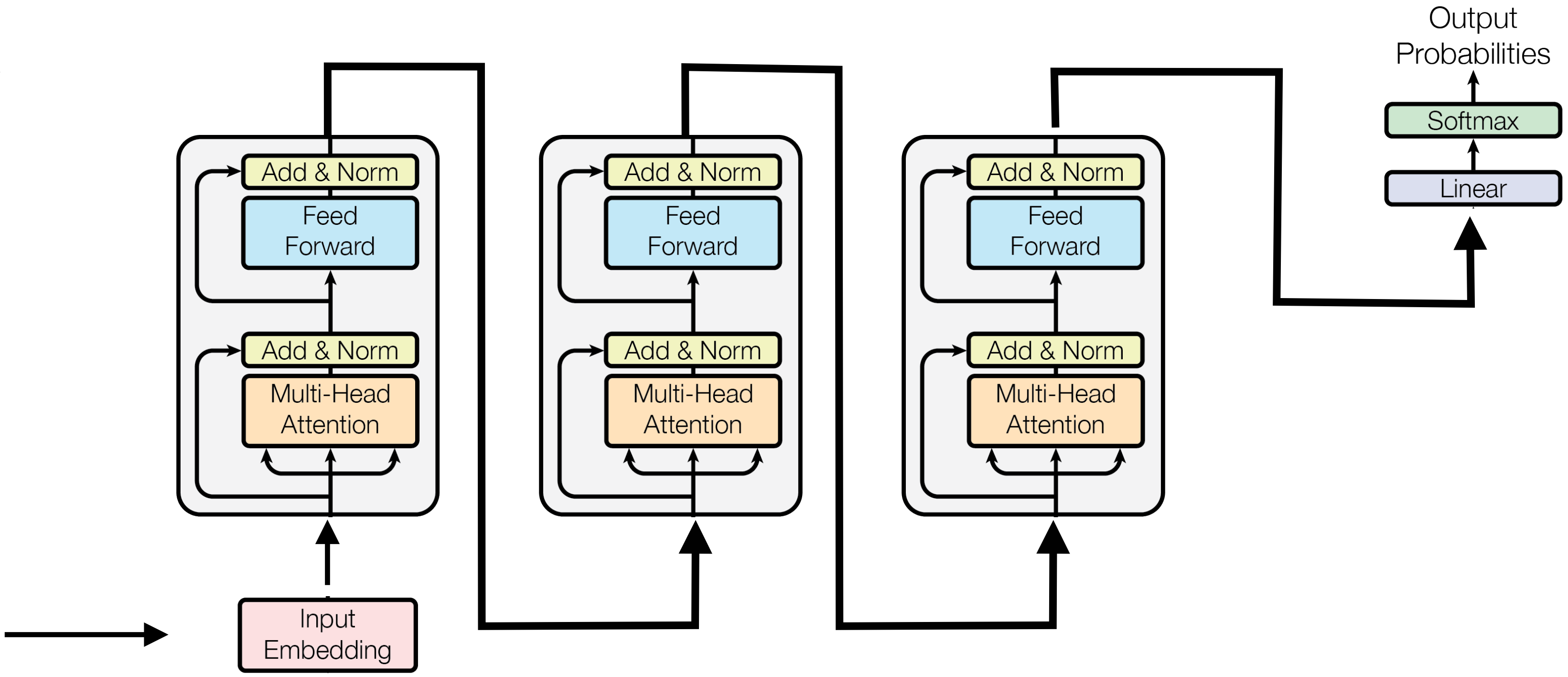
- Input - jet constituents 4-momenta

$$\left\{ p_T^i, E^i, \frac{p_T^i}{p_T^{\text{jet}}}, \frac{E^i}{E^{\text{jet}}}, \Delta\phi^i, \Delta\eta^i, \Delta R^i = \sqrt{\Delta\eta^2 + \Delta\phi^2} \right\}$$

$$p\left(x \,|\, \text{top}\right)$$

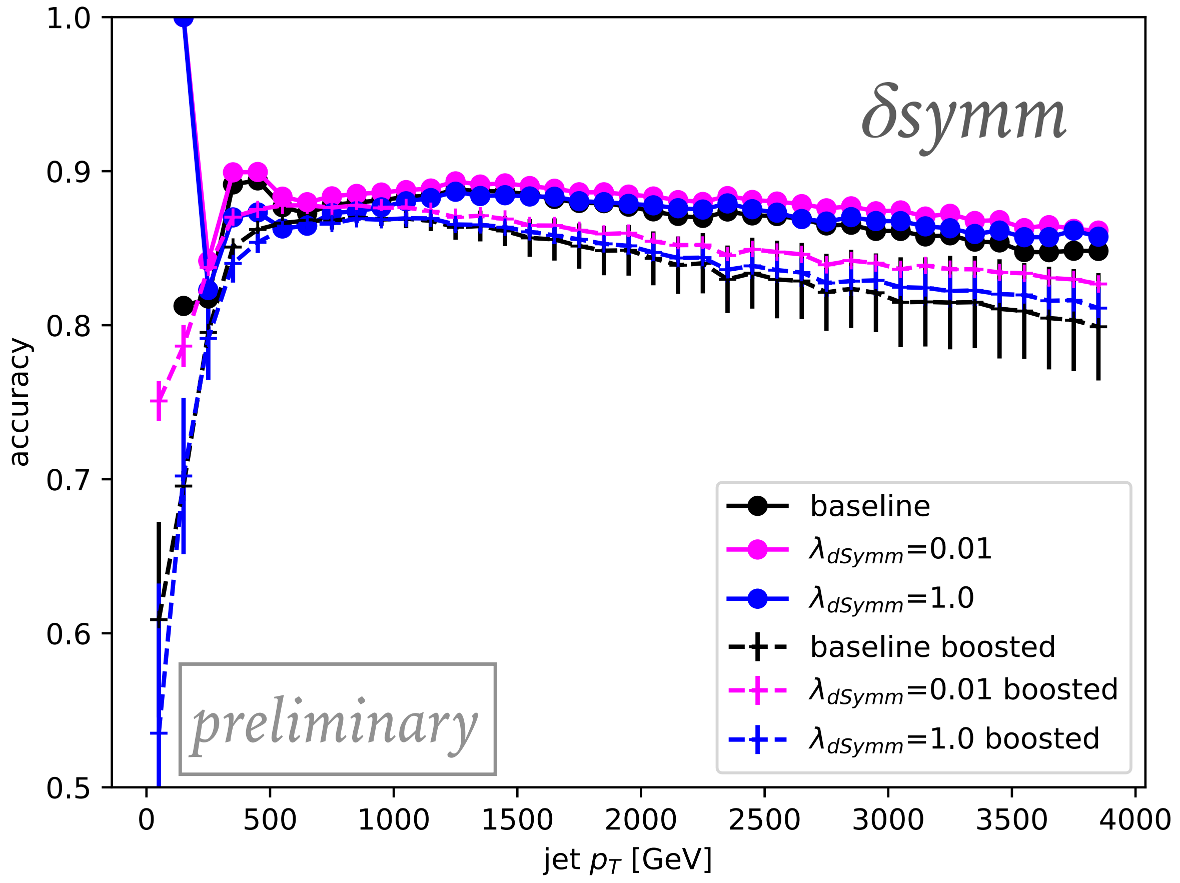
- Transformer -

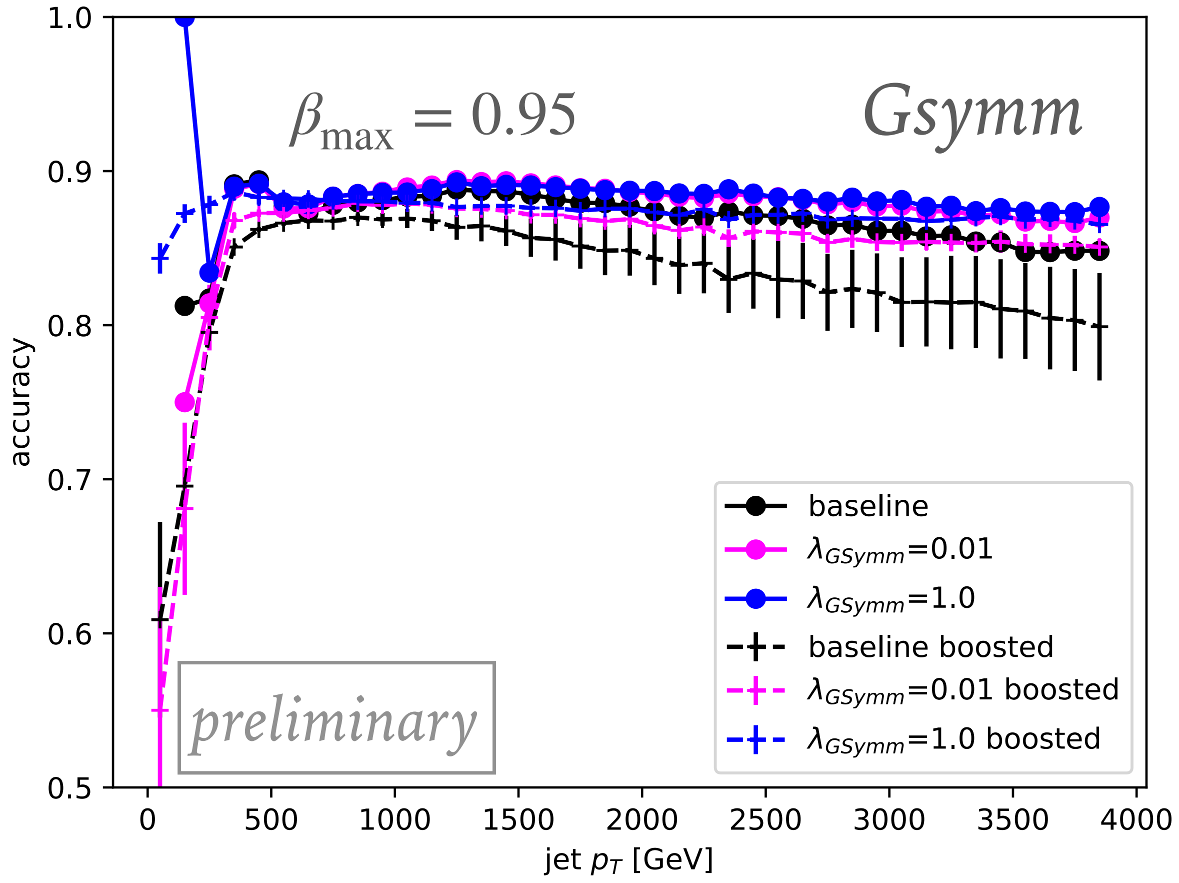




TOP TAGGING RESULTS

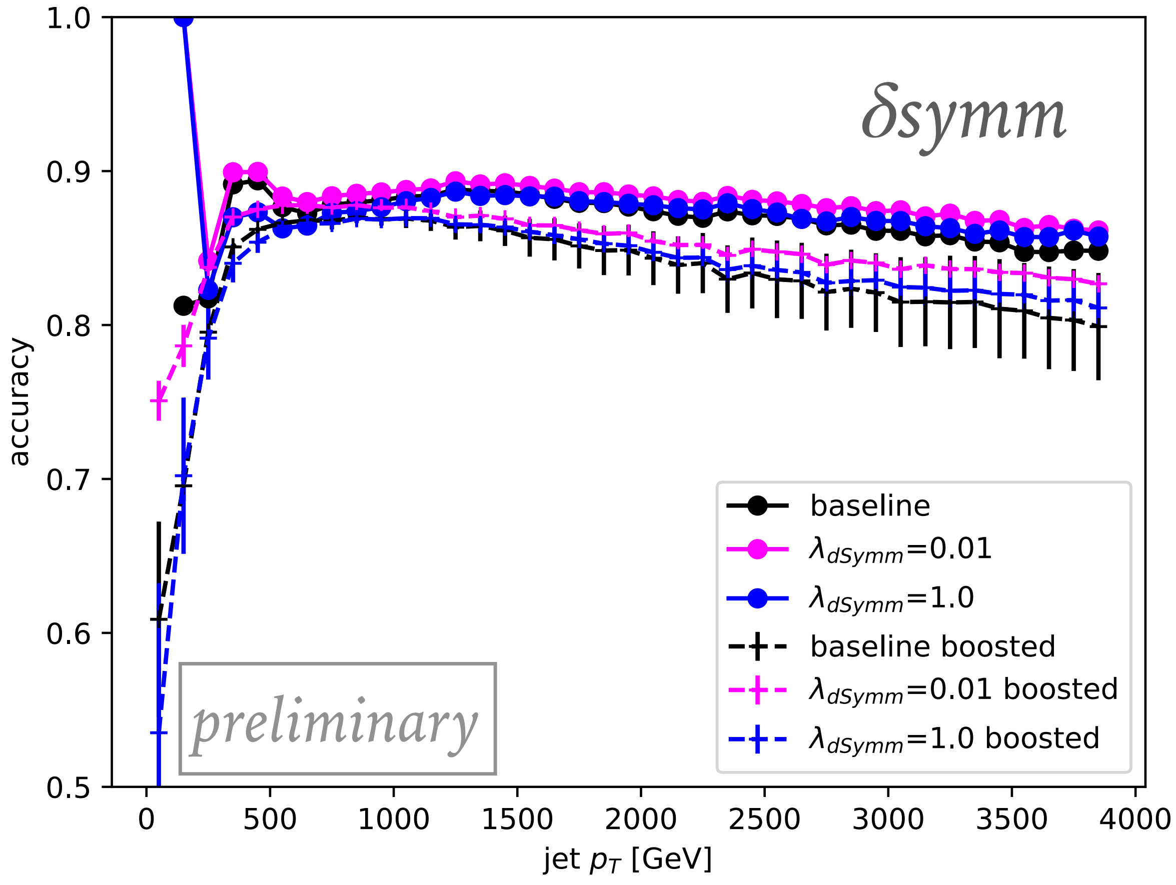
- Invariance check - boosting inputs and assigning them with the same truth values as original inputs.
- As expected - improved and flatter performance over boosted inputs.

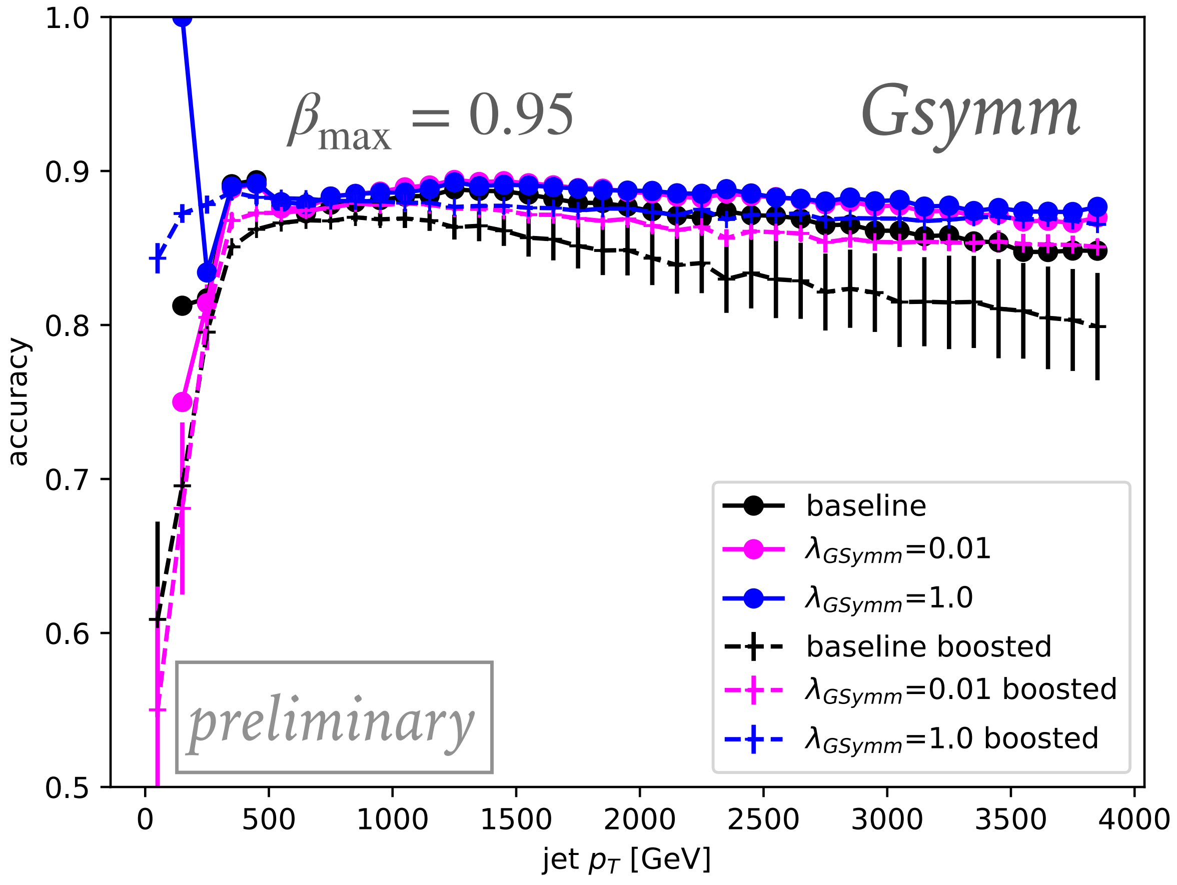




TOP TAGGING RESULTS

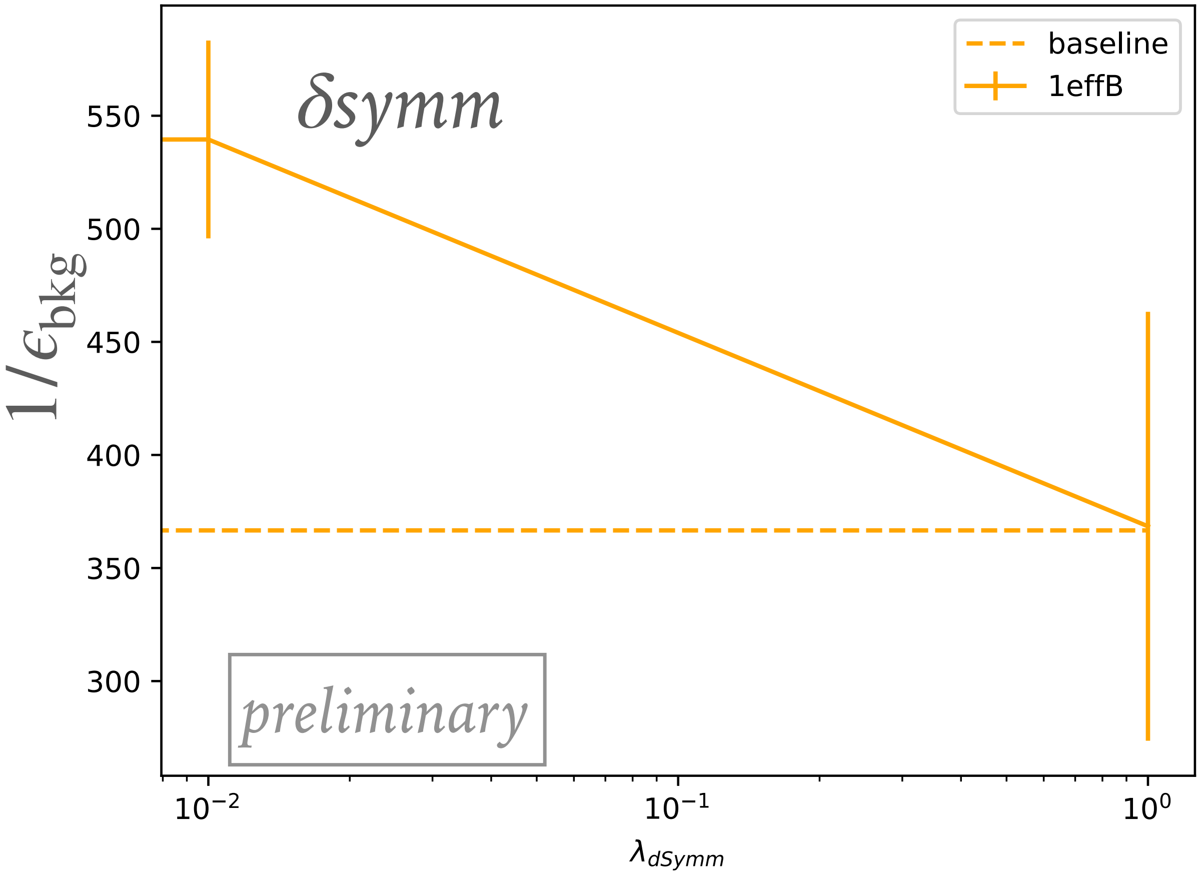
- For the real data - performance at least on-par with baseline.

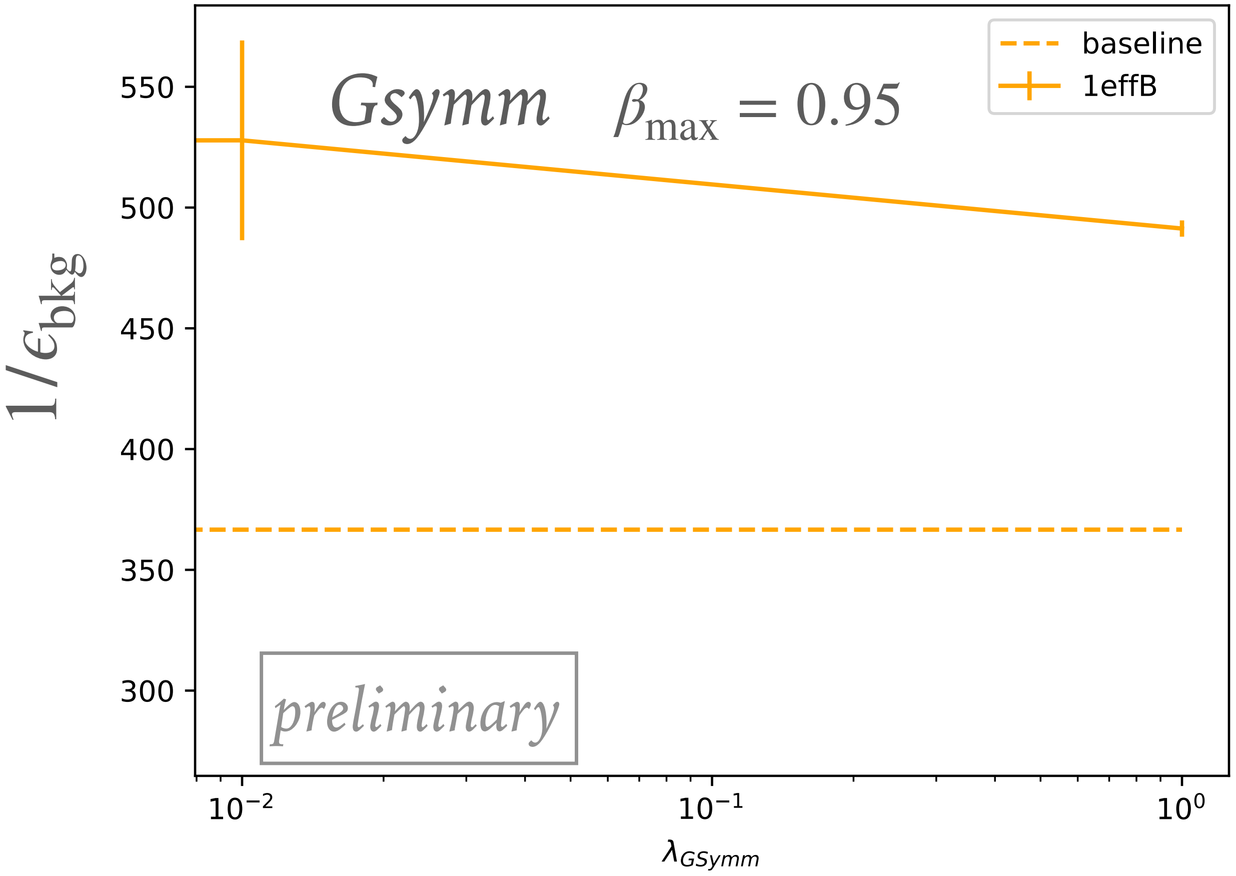




TOP TAGGING RESULTS

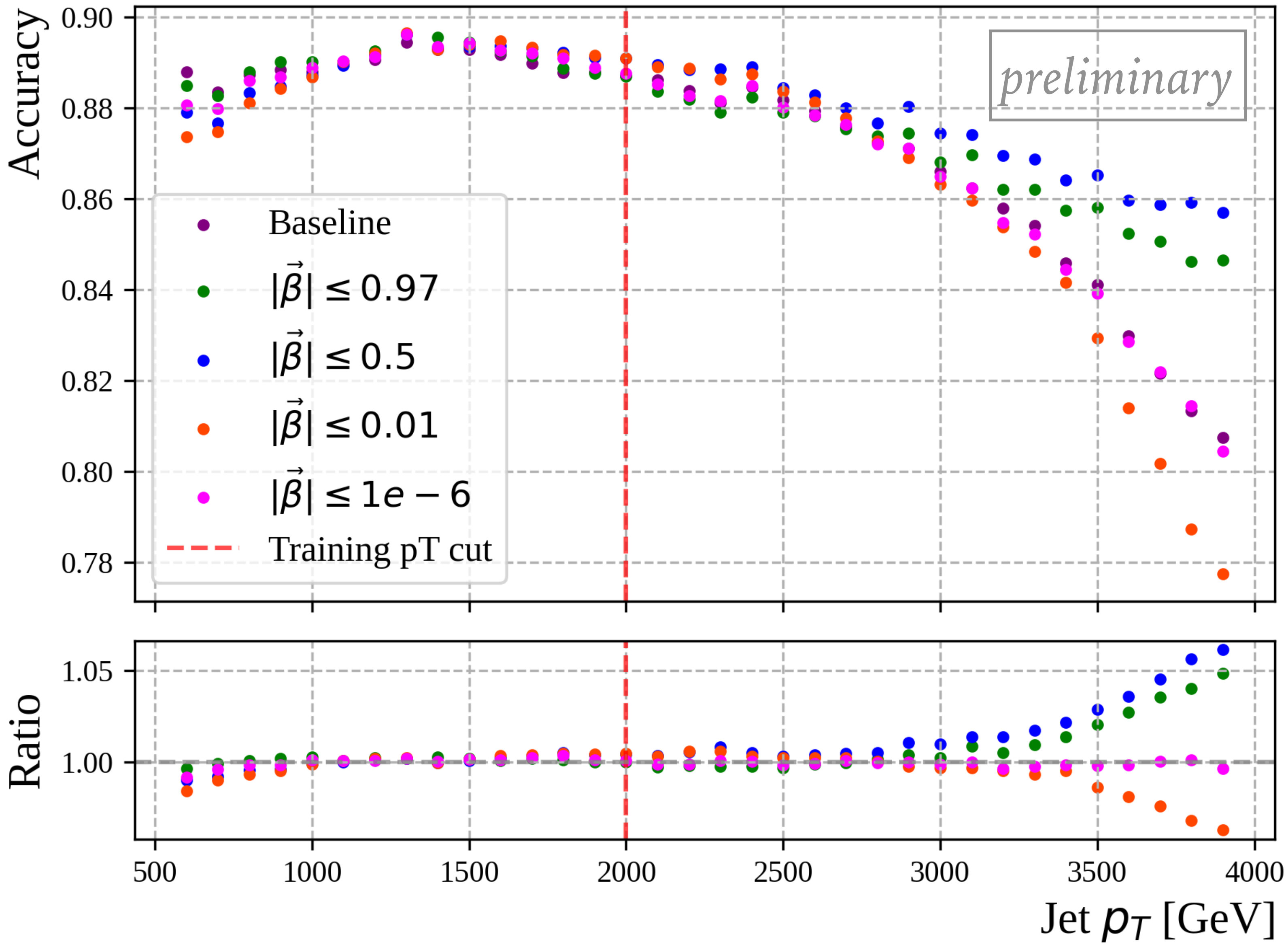
- For the real data - performance at least on-par with baseline.
- Improved background rejection at signal efficiency of 0.3.





TOP TAGGING RESULTS

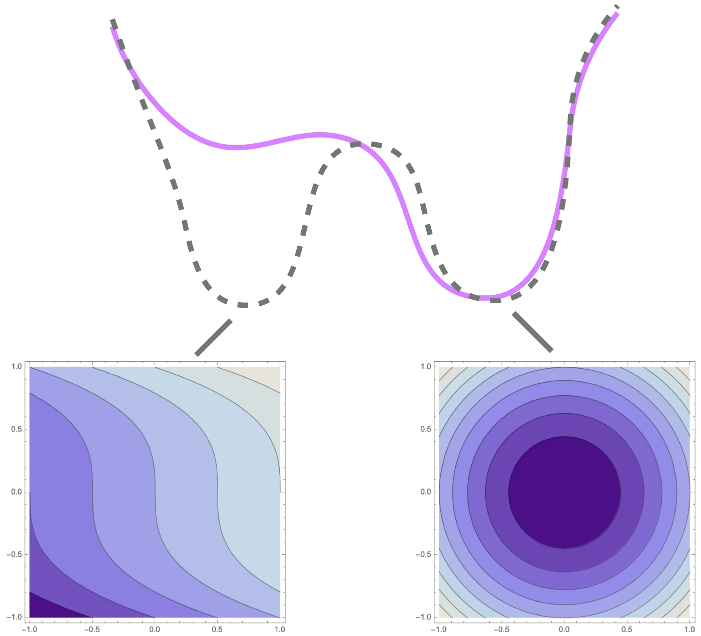
- Extrapolation test: train only on $p_T \leq 2$ TeV
- Gsymm with $\beta_{\max} = 0.5$ extrapolates best to unseen p_T !



CONCLUSIONS

- ML + physical knowledge help extract more information from data.
- Symmetries can be imposed on architecture-level, but can be challenging to build and train.
- **SymmLoss - bias the model towards respecting symmetries.**

$$\mathscr{L} = \mathscr{L}_{\text{task}} + \lambda_{\text{symm}} \mathscr{L}_{\text{symm}}$$



  - Flexible - can be added to any model, easy to implement.
  - Multi-purpose - accommodates approximate symmetries (and no symmetries).
  - Bias is tunable and controllable.
  - **Better results for symmetric problems, even if the symmetry is broken.**

FUTURE WORK

- Working on full comparison to PELICAN & L-GATr
- Performance -
  - combine with SOTA models
  - Other ideas for broken symmetry losses
- Scaling behavior

THANK YOU!