Symmetries and anomalies

Tilman Plehn

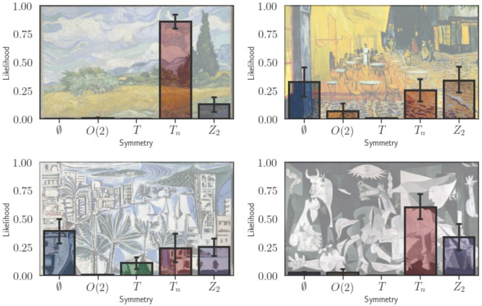
Universität Heidelberg

Stuttgart 7/2022

Symmetries and invariances

Symmetries in particle physics

- analyze & learn symmetries



Symmetries and invariances

Symmetries in particle physics

- analyze & learn symmetries
- learn symmetries
- equivariant network

$$f_\theta(S(x)) = S(f_\theta(x))$$

$$\text{or} \qquad f_\theta(x_1) = f_\theta(x_2) \quad \Rightarrow \quad f_\theta(S(x_1)) = S(f_\theta(x_1)) = S(f_\theta(x_2)) = f_\theta(S(x_2))$$

- invariant network

$$f_\theta(S(x)) = f_\theta(x)$$

→ Matter of taste, style, sophistication

Symmetries and invariances

Symmetries in particle physics

- analyze & learn symmetries
- learn symmetries
- equivariant network

$$f_\theta(S(x)) = S(f_\theta(x))$$

$$\text{or} \qquad f_\theta(x_1) = f_\theta(x_2) \quad \Rightarrow \quad f_\theta(S(x_1)) = S(f_\theta(x_1)) = S(f_\theta(x_2)) = f_\theta(S(x_2))$$

- invariant network

$$f_\theta(S(x)) = f_\theta(x)$$

→ Matter of taste, style, sophistication

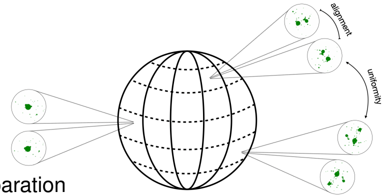
Invariant classifiers

- Lorentz symmetries,
  permutation symmetries,
  theory invariances
- symmetry-related jets in same latent point
- improved training, resilience, anomaly searches

→ Put latent space to use

Contrastive learning

Self-supervised JetCLR



- original dataset $\{x_i\}$
  symmetry-augmented dataset $\{x'_i\}$
- 'positive' pairs $\{(x_i, x'_i)\}$
  'negative' pairs $\{(x_i, x_j)\} \cup \{(x_i, x'_j)\}$
- compact latent sphere with angular separation

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|} \in [-1, 1] \; ,$$

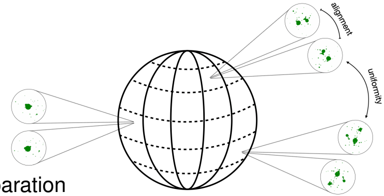
- training
  positive pairs aligned $s(z_i, z'_i) \to 1$
  negative pairs uniformly distributed

$$L_{\text{CLR}} = - \sum_{i \in \text{batch}} \frac{s(z_i, z'_i)}{\tau} + \sum_{i \in \text{batch}} \log \sum_{j \neq i \in \text{batch}} \left[e^{s(z_i, z_j)/\tau} + e^{s(z_i, z'_j)/\tau} \right]$$

$$= - \sum_{i \in \text{batch}} \log \frac{e^{s(z_i, z'_i)/\tau}}{\sum_{j \neq i \in \text{batch}} \left[e^{s(z_i, z_j)/\tau} + e^{s(z_i, z'_j)/\tau} \right]} \; ,$$

Contrastive learning

Self-supervised JetCLR

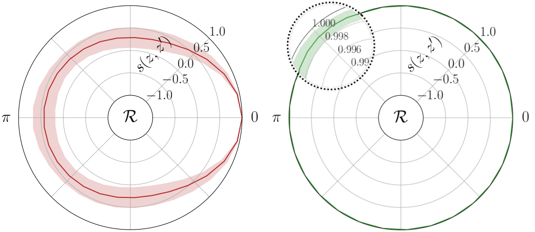


- original dataset $\{x_i\}$
  symmetry-augmented dataset $\{x_i'\}$
- 'positive' pairs $\{(x_i, x_i')\}$
  'negative' pairs $\{(x_i, x_j)\} \cup \{(x_i, x_j')\}$
- compact latent sphere with angular separation

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|} \in [-1, 1] \;,$$

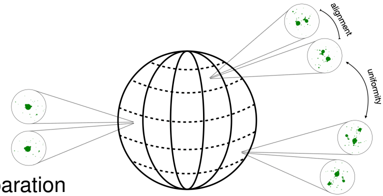
Particle-physics application

- jet symmetries rotation, translation, permutation



Contrastive learning

Self-supervised JetCLR



- original dataset $\{x_i\}$
  symmetry-augmented dataset $\{x_i'\}$
- 'positive' pairs $\{(x_i, x_i')\}$
  'negative' pairs $\{(x_i, x_j)\} \cup \{(x_i, x_j')\}$
- compact latent sphere with angular separation

$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|} \in [-1, 1] \; ,$$

Particle-physics application

- jet symmetries rotation, translation, permutation
- jet augmentations collinear merging, soft noise
- linear classifier test [top vs QCD]

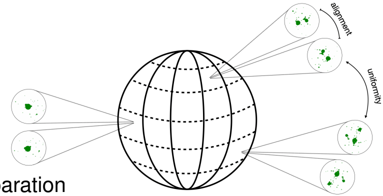
augmentation	$\epsilon_b^{-1}(\epsilon_s = 0.5)$	AUC
none	15	0.905
translations	19	0.916
rotations	21	0.930
soft+collinear	89	0.970
all combined (default)	181	0.980

Contrastive learning

Self-supervised JetCLR

- original dataset $\{x_i\}$
  symmetry-augmented dataset $\{x_i'\}$
- 'positive' pairs $\{(x_i, x_i')\}$
  'negative' pairs $\{(x_i, x_j)\} \cup \{(x_i, x_j')\}$
- compact latent sphere with angular separation

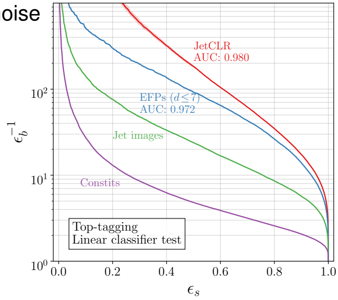
$$s(z_i, z_j) = \frac{z_i \cdot z_j}{|z_i||z_j|} \in [-1, 1] ,$$



Particle-physics application

- jet symmetries rotation, translation, permutation
- jet augmentations collinear merging, soft noise
- linear classifier test [top vs QCD]
- benchmarked with other methods

→ Symmetries putting theory into ML-tools



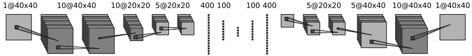
Autoencoders

Unsupervised classification

- train on background only
  extract unknown signal from reconstruction error
- reconstruct QCD jets $\rightarrow$ top jets hard to describe
- reconstruct top jets $\rightarrow$ QCD jets just simple top-like jet

$\rightarrow$ Symmetric performance $S \leftrightarrow B$?

Autoencoders



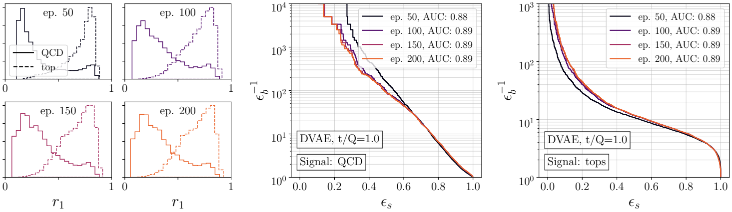
Unsupervised classification

- train on background only
  extract unknown signal from reconstruction error
- reconstruct QCD jets $\quad\to\quad$ top jets hard to describe
- reconstruct top jets $\quad\to\quad$ QCD jets just simple top-like jet

$\to$ Symmetric performance $S \leftrightarrow B$?

Moving to latent space

- anomaly score from latent space?
- VAE $\quad\to\quad$ does not work
  GMVAE $\quad\to\quad$ does not work
  Dirichlet VAE $\quad\to\quad$ works okay
  density estimation $\quad\to\quad$ does not work



Normalized autoencoder

Energy-based models

- goal penalize features away from background
- train on normalized probability
- Boltzmann-distribution with $x \to E_\theta$ =MSE

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta} \qquad \text{with} \qquad Z_\theta = \int_x dx e^{-E_\theta(x)}$$

$$L = -\Big\langle \log p_\theta(x) \Big\rangle_{p_\text{data}} = \Big\langle E_\theta(x) + \log Z_\theta \Big\rangle_{p_\text{data}}$$

$\to$ Small MSE for data, large MSE for model

Normalized autoencoder

Energy-based models

- goal   penalize features away from background
- train on normalized probability
- Boltzmann-distribution with $x \to E_\theta$ =MSE

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta} \qquad \text{with} \qquad Z_\theta = \int_x dx e^{-E_\theta(x)}$$

$$L = -\Big\langle \log p_\theta(x) \Big\rangle_{p_\text{data}} = \Big\langle E_\theta(x) + \log Z_\theta \Big\rangle_{p_\text{data}}$$

- gradient of loss with normalization term

$$\begin{aligned} -\nabla_\theta \log p_\theta(x) &= \nabla_\theta E_\theta(x) + \nabla_\theta \log Z_\theta \\ &= \nabla_\theta E_\theta(x) + \frac{1}{Z_\theta} \nabla_\theta \int_x dx e^{-E_\theta(x)} \\ &= \nabla_\theta E_\theta(x) - \int_x dx \frac{e^{-E_\theta(x)}}{Z_\theta} \nabla_\theta E_\theta(x) \\ &= \nabla_\theta E_\theta(x) - \Big\langle \nabla_\theta E_\theta(x) \Big\rangle_{p_\theta} \end{aligned}$$

- background metric for expectation value

$$\Big\langle -\nabla_\theta \log p_\theta(x) \Big\rangle_{p_\text{data}} = \Big\langle \nabla_\theta E_\theta(x) \Big\rangle_{p_\text{data}} - \Big\langle \nabla_\theta E_\theta(x) \Big\rangle_{p_\theta}$$

$\to$ Small MSE for data, large MSE for model

Normalized autoencoder

Energy-based models

- goal   penalize features away from background
- train on normalized probability
- Boltzmann-distribution with $x \to E_\theta$ =MSE

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta} \qquad \text{with} \qquad Z_\theta = \int_x dx e^{-E_\theta(x)}$$

$$L = -\Big\langle \log p_\theta(x) \Big\rangle_{p_\text{data}} = \Big\langle E_\theta(x) + \log Z_\theta \Big\rangle_{p_\text{data}}$$

$\to$ Small MSE for data, large MSE for model

Energy-based autoencoder

- still need to compute Z_θ
  integration over phase space x
- (Langevin) Markov Chain

$$x_{t+1} = x_t + \lambda_x \nabla_x \log p_\theta(x) + \sigma_x \epsilon_t \qquad \text{with} \qquad \epsilon_t \sim \mathcal{N}_{0,1}$$

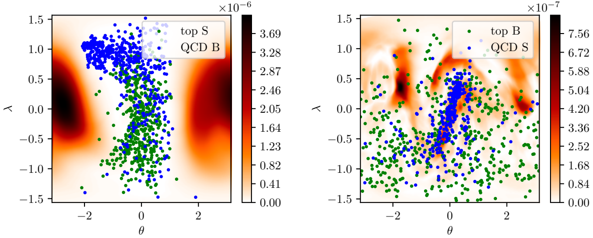
- problem x-space high-dimensional and hard to model
  autoencoder   sample in and around latent space   [physics manifold]
- MC abuse   100s of chains with 30 steps

$\to$ Autoencoder the perfect EBM

NAE performance

Top vs QCD autoencoding

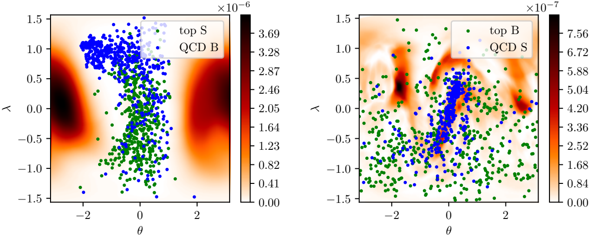
- regular autoencoder pre-training



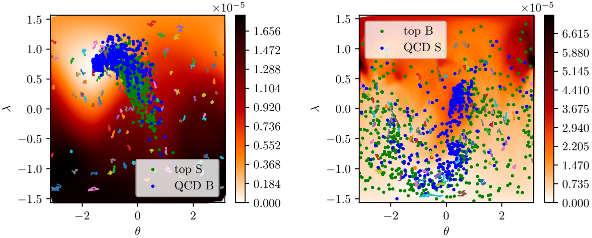
NAE performance

Top vs QCD autoencoding

- regular autoencoder pre-training



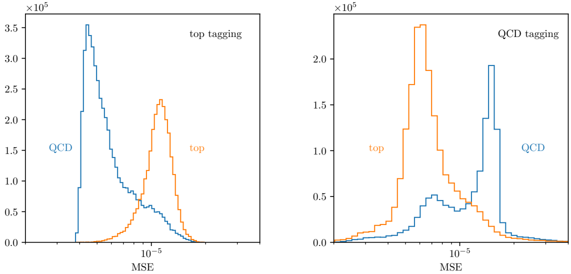
- normalized autoencoder training



NAE performance

Top vs QCD autoencoding

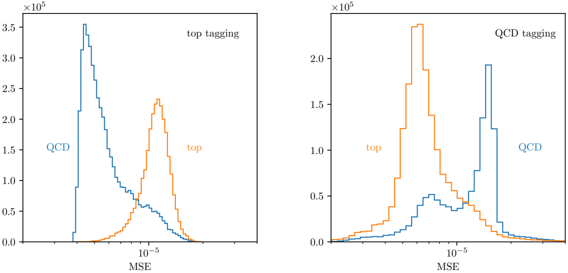
- regular autoencoder pre-training
- normalized autoencoder training
- MSE distributions for background and (unknown) signal



NAE performance

Top vs QCD autoencoding

- regular autoencoder pre-training
- normalized autoencoder training
- MSE distributions for background and (unknown) signal



→ Performance optimization next

→ Still simple autoencoder with better training

Outlook

ML and LHC

- big and fast dataset
- combined with precision simulations
- advanced inference with focus on uncertainties
- fundamental theory questions
- rich symmetry structure

Specific architectures we covered

- Bayesian regression and classification networks
- self-supervision
- autoencoders
- Bayesian INNs
- conditional INNs
- symbolic regression

  ...