A Tale of Symmetry and Duality in Neural Networks

Anindita Maiti

Based on arXiv:2106.00694 (with J. Halverson and K. Stoner)

String Data 2021

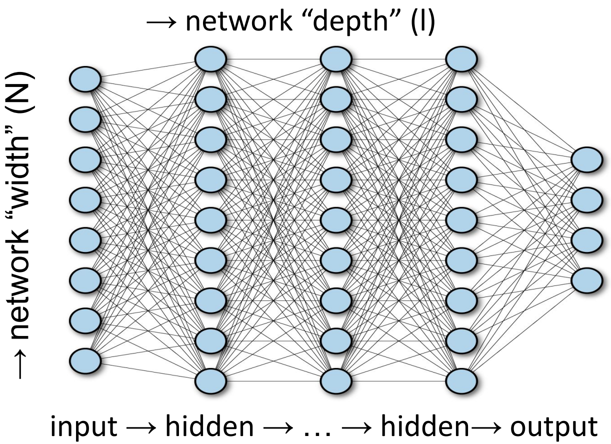
OVERVIEW

- Introduction to NN-QFT Correspondence
- Parameter Space - Function Space Duality
- Symmetry-via-Duality & Examples
- Symmetry Breaking & Deep Learning

Introduction to NN-QFT Correspondence

What are Neural Networks?

Neural Networks (NN) are functions on inputs, learnable parameters θ and discrete hyperparameters N (width), l (depth).



Fully connected feedforward NN

$$f_i : \mathbb{R}^d \to \mathbb{R}^D$$

$$z_i^l(x) = b_i^l + \sum_{j=1}^{N} W_{ij}^l x_j^l(x)$$

$$x_j^l = \sigma(z_j^{l-1}(x))$$

Schematic diagram

NNs are architectures on nodes and edges

Ensembles of NN outputs can be studied using statistical distributions.

NN-QFT Correspondence

At infinite N, outputs of initialized networks are sums over infinite i.i.d. parameters.

[Neal], [Williams], 1990's,
[Lee et al., 2017],
[Matthews et al., 2018,],
[Yang, 2019]

Central Limit Theorem (CLT): such sum is drawn from Gaussian distributions. Output function space well described by free Scalar Field theory.

[Novak et al., 2018]
[Garriga-Alonso et al. 2018],
[Jacot et al., 2018],
[Lee et al., 2019]

Close to GP limit, ensembles of NN outputs (for most architectures) are well described by perturbative field theory.

[Halverson, A.M., Stoner 2008.08601]

Model for GP action: $P[f] \sim \exp\left[-\frac{1}{2}\int d^dx\, d^dx'\, f(x)\Xi(x,x')f(x')\right]$ w/ $\int d^dx' K(x,x')\,\Xi(x',x'') = \delta^{(d)}(x-x'')$

K(x,x') : kernel or 2-pt function of NNGP

Model for NGP action: $S = S_{\text{GP}} + \Delta S$ with $\Delta S = \int d^dx\, [g f(x)^3 + \lambda f(x)^4 + \alpha(x)^5 + \kappa f(x)^6 + \cdots]$

Parameter Space - Function Space Duality

Two different ways of computing same functions in NN

Symmetries of NN Gaussian Processes

Symmetries of the action is essential to any field theory description

NNGP symmetries are fixed by symmetries of 2-pt function (by Wick's theorem).

$$G^{(2)}_{i_1 i_2}(x_1, x_2) = \delta_{i_1 i_2} K(x_1, x_2)$$

i_n : output space indices.

Parameters drawn from SO(D) invariant distributions $\to$ SO(D) invariant NNGP action.

$$G^{(2n)}_{i_1,\ldots,i_{2n}}(x_1, \ldots, x_{2n}) = \sum_{P \in \mathrm{Wick}(2n)} \delta_{i_{a_1} i_{b_1}} \ldots \delta_{i_{a_n} i_{b_n}} K(x_{a_1}, x_{b_1}) \ldots K(x_{a_n}, x_{b_n})$$

$$G^{(2n+1)}(x_1, \ldots, x_n) = 0$$

$$\mathrm{Wick}(n) = \{P \in \mathrm{Partitions}(1, \ldots, n) \mid |p| = 2 \ \forall p \in P\}$$
$$P = \{(a_1, b_1), \ldots, (a_n, b_n)\}$$

Transformations by R $\in$ SO(D) $\to$ output transforms as $f_i \mapsto R_{ij} f_j$.

Correlators are invariant: $\delta_{ik} \mapsto R_{ij} R_{kl} \delta_{jl} = (R\, R^T)_{ik} = \delta_{ik}.$

Symmetries at Non-Gaussian Process

At finite width, n>2 correlators receive EFT corrections (2-pt function is exact at all N).

Full action unknown → symmetries of correlators can't be deduced in field space.
Exact correlators can be studied in parameter space.

Exploding number of parameters at large N, but finding symmetries becomes easier.

Transformation $f'(x) = \Phi(f(x'))$
leaves functional density invariant

$$D[\Phi f]\, e^{-S[\Phi f]} = Df\, e^{-S[f]}$$

when n-pt functions are invariant:

$$\mathbb{E}[f(x_1)\dots f(x_n)] = \frac{1}{Z_f}\int Df\, e^{-S[f]}\, f(x_1)\dots f(x_n)$$
$$= \frac{1}{Z_f}\int Df'\, e^{-S[f']}\, f'(x_1)\dots f'(x_n) = \frac{1}{Z_f}\int D[\Phi f]\, e^{-S[\Phi f]}\, \Phi(f(x_1'))\dots\Phi(f(x_n'))$$
$$= \frac{1}{Z_f}\int Df\, e^{-S[f]}\, \Phi(f(x_1'))\dots\Phi(f(x_n')) = \mathbb{E}[\Phi(f(x_1'))\dots\Phi(f(x_n'))]$$

Absorb transformations of correlators into transformations of parameters $\theta_T \subset \theta$.

Invariance of P_{θ_T} leads to invariance of NN action S[f].

Symmetry-via-Duality
&
Examples

Symmetry-via-Duality Technique

"Symmetry-via-Duality": Use parameter space - function space duality to infer transformation group G that leaves NN action invariant.

$$f = f_\theta$$

Parameter Space

$$\begin{aligned} G^{(n)}(x_1, \cdots, x_n) &= \mathbb{E}_\theta[f(x_1)\cdots f(x_n)] \\ &= \frac{1}{Z_\theta}\int d\theta\, f(x_1)\cdots f(x_n) P_\theta \end{aligned}$$

$$Z_\theta = \int d\theta\, P_\theta$$

Function Space

$$\begin{aligned} G^{(n)}(x_1, \cdots, x_n) &= \mathbb{E}_f[f(x_1)\cdots f(x_n)] \\ &= \frac{1}{Z_f}\int Df\, f(x_1)\cdots f(x_n) P_f \end{aligned}$$

$$Z_f = \int Df\, P_f$$

Two different ways of studying correlation functions in Neural Nets

Symmetry-via-Duality Examples

(a) **SO(D) Output Symmetry**: Final linear layer parameters drawn from SO(D) invariant distributions $P_W = P_{R^{-1}\tilde{W}} = P_{\tilde{W}}$ & $P_b = P_{R^{-1}\tilde{b}} = P_{\tilde{b}}$, for $R \in SO(D)$, $|R| = 1$.

$$f_i(x) = W_{ij} g_j(x) + b_i \qquad f_i \mapsto R_{ij} f_j$$

(θ_g: all other parameters)

$$\begin{aligned} G'^{(n)}_{i_1 \ldots i_n}(x'_1, \ldots, x'_n) &= \mathbb{E}[R_{i_1 j_1} f_{j_1}(x_1) \ldots R_{i_n j_n} f_{j_n}(x_n)] \\ &= \frac{1}{Z_\theta} \int DW\, Db\, D\theta_g\, R_{i_1 j_1}(W_{j_1 k_1} g_{k_1}(x_1) + b_{j_1}) \ldots R_{i_n j_n}(W_{j_n k_n} g_{k_n}(x_n) + b_{j_n}) P_W P_b P_{\theta_g} \\ &= \frac{1}{Z_\theta} \int |R^{-1}|^2 D\tilde{W}\, D\tilde{b}\, D\theta_g\, (\tilde{W}_{i_1 k_1} g_{k_1}(x_1) + \tilde{b}_{i_1}) \ldots (\tilde{W}_{i_n k_n} g_{k_n}(x_n) + \tilde{b}_{i_n}) P_{R^{-1}\tilde{W}} P_{R^{-1}\tilde{b}} P_{\theta_g} \\ &= \mathbb{E}[f_{i_1}(x_1) \ldots f_{i_n}(x_n)] = G^{(n)}(x_1, \ldots, x_n) \end{aligned}$$

Invariant

(b) **SO(d) Input Symmetry**: First linear layer parameters drawn from SO(d) invariant distributions. $R \in SO(d)$, inputs transform as $x_i \mapsto x'_i = R_{ij} x_j$.

$$f_i(x) = g_{ij}(W_{jk} x_k)$$

Include bias trivially

Symmetry-via-Duality Examples

(c) **Translation Input Symmetry**: First linear layer with deterministic weights, bias $b \sim \mathcal{U}(S^1)$.

Translations $x_k \mapsto x_k + c_k$ transform output $f_i(x) = g_{ij}((W_{jk}x_k)\,\%\,1 + b_j)$ to $f'(x') = g_{ij}((W_{jk}x_k)\,\%\,1 + b'_j)$, $b'_j = (W_{jk}c_k)\,\%\,1 + b_j$.

$$G^{(n)}_{i_1,\ldots,i_n}(x_1 + c, \ldots, x_n + c) = \mathbb{E}[f'_{i_1}(x'_n) \cdots f'_{i_n}(x'_n)]$$

$$= \mathbb{E}[f_{i_1}(x_n) \cdots f_{i_n}(x_n)] = G^{(n)}_{i_1,\cdots,i_n}(x_1, \cdots, x_n)$$

(d) **SU(D) Output Symmetry:** Complex last linear layer parameters (all parts from identical SO(D) invariant dist.)

Appropriate choice of parameter distributions lead to other invariant NN densities.

Equivalence of Symmetries

Neural Nets	Field Theories
input x	space-time points
network output $f(x)$	free or interacting fields
input layer symmetries	space-time symmetries
output layer symmetries	internal symmetries

Internal symmetries from hidden layers give NN additional structures beyond those in field theory.

Symmetry Preservation & Deep Learning

Training can preserve initialization symmetries, if invariances of P_θ persist at all t.

Infinitesimal Gradient Descent: $\boxed{\frac{\partial P_\theta(t)}{\partial t} = \left(\frac{\partial}{\partial \theta_i}\frac{\partial}{\partial \theta_i}\mathcal{L}\right) P_\theta(t) + \frac{\partial P_\theta(t)}{\partial \theta_i}\frac{\partial \mathcal{L}}{\partial \theta_i}}$

SO(D) symmetry preservation example: $\frac{\partial P_\theta(t)}{\partial \theta_i} = I_P\, \theta_i$ & $\frac{\partial \mathcal{L}}{\partial \theta_i} = I_\mathcal{L}\, \theta_i\,.$

$a_j \in \mathbb{R}$

$$\mathcal{L} = \sum_{x,y}\left(f_i(x) f_i(x) - y_j y_j\right) \qquad P_\theta(0) = \exp[-\textstyle\sum_{j=1}^k a_j (\mathrm{Tr}(\theta^T\theta))^j]$$

Symmetry Invariant Correlated Parameters: Trained output ensembles at infinite N can still be modeled by EFT, If parameter mixing is close to Gaussian.

Symmetry properties can still persist too!

$$\mathcal{P}_\theta = e^{-\frac{1}{2\sigma_\theta^2}\theta_{\alpha\beta}^2 - \lambda_\theta\, \theta_{ab}\theta_{ab}\theta_{cd}\theta_{cd}}$$

Symmetry Breaking
&
Deep Learning

Training → Symmetry Breaking

Supervised Learning → NN output distribution flows to some nonzero mean

Training turns on nonzero mean, this breaks rotational invariance.

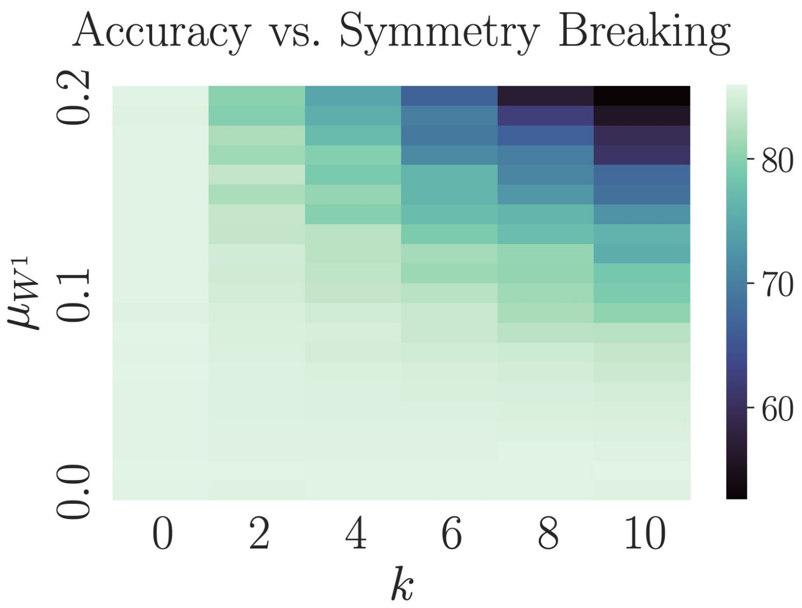
Thus, training causes symmetry breaking.

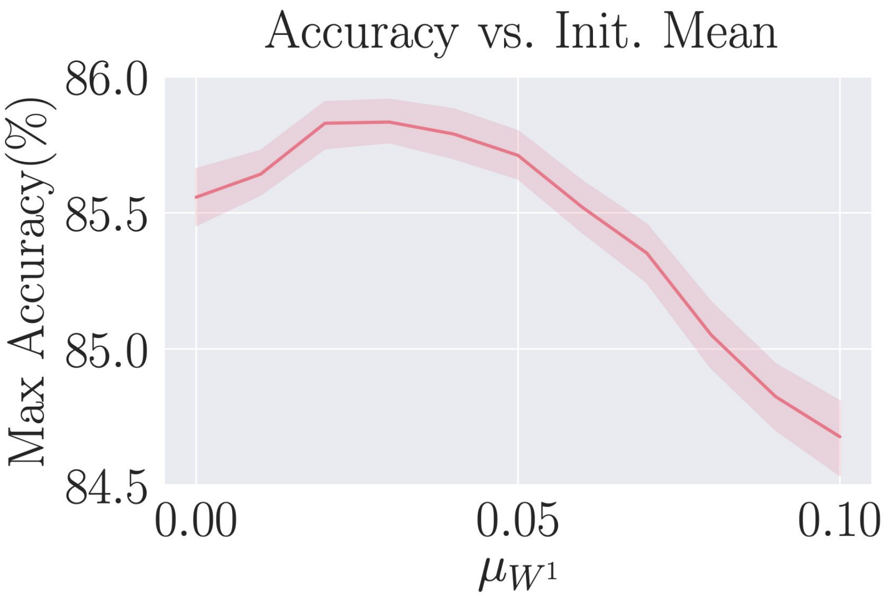
Q. Does broken symmetry at initialization result in better training?

Run simple experiments to check.

A. Symmetry breaking at initialization doesn't always improve training. Symmetry needs to be broken intelligently.

Symmetry Breaking Experiments





$$W_{ij}^l \sim \mathcal{N}(\mu_{W^l}, 1/\sqrt{N}), \quad \forall i < k+1 \qquad , \qquad W_{ij}^l \sim \mathcal{N}(0, 1/\sqrt{N}), \quad \forall i \geq k+1$$

Result: Nonzero initial means lead to better training when those are proportional to ground truth.

Conclusion

- Parameter Space - Function Space duality: potentially gives a way to approach field theories in parameter space.

- Symmetry-via-Duality: infer symmetries of NN action at all N through invariance of P_{θ_T}.

- NN input and output symmetries are equivalent to space-time and internal symmetries in QFT.

- Through judicious choice of initialization PDFs, loss functions & architectures, can obtain invariant network ensembles during training.

Thank You!

Questions?