NN-QFT Correspondence

Jim Halverson

Northeastern University

String Data 2020, CERN

Based on: 2008.08601 with Maiti and Stoner

The Gang

Anindita Maiti
targeting physics postdocs, Fall 2022

Keegan Stoner
targeting ML labs, Fall 2022

What is learning?

Legend:

🔴 Randomly initialized NN

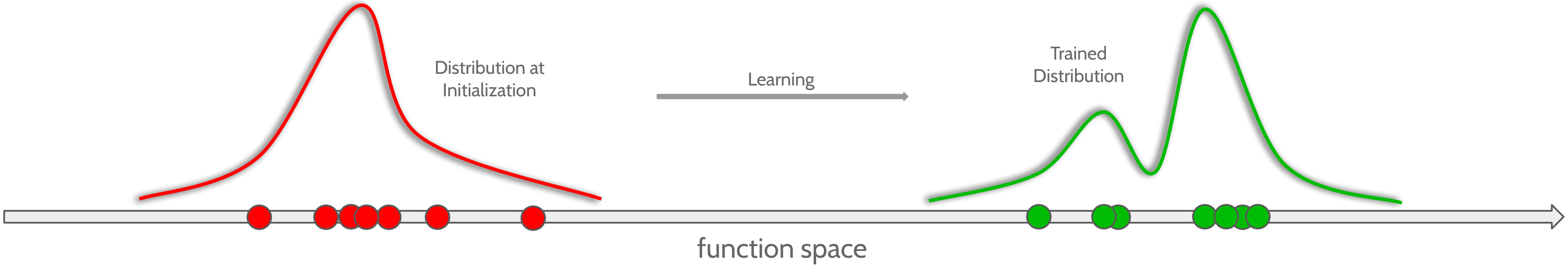
🟢 Trained NN

Physics Language:

Learning is a data-induced flow from an initialization function-space distribution to a trained distribution.

Bayesian Statistics:

Learning is approximating the posterior over functions given a prior, a likelihood, and data.



Function-space distributions are central objects in machine learning.

Already sounds a little like the path integral of QFT.

What can we say about the function-space distributions of NNs?

Start Simple: Single-Layer Networks

A single-layer feedforward network is just

$$f_{\theta,N} : \mathbb{R}^{d_{\text{in}}} \xrightarrow{W_0,\, b_0} \mathbb{R}^N \xrightarrow{\sigma} \mathbb{R}^N \xrightarrow{W_1,\, b_1} \mathbb{R}^{d_{\text{out}}}$$

$$f(x) = W_1(\sigma(W_0 x + b_0)) + b_1$$

parameters drawn as $b_0, b_1 \sim \mathcal{N}(\mu_b, \sigma_b^2)$

$W_0 \sim \mathcal{N}(\mu_W, \sigma_W^2/d_{\text{in}})$ $\quad W_1 \sim \mathcal{N}(\mu_W, \sigma_W^2/N)$

Limit of interest: infinite width N → ∞.

Then output adds an infinite number of i.i.d. entries from W_1 matrix, so CLT applies, output drawn from Gaussian!
Language: the neural network f is drawn from a ***Gaussian process***, i.e. Gaussian function-space distribution.

∞ width single-layer networks drawn from GP

Was just a simple consequence of CLT.
Surely this must be more general!

Just need a discrete hyperparameter N such that as N → ∞,
the associated ***asymptotic*** NN adds infinite number of i.i.d. variables,
then apply CLT.

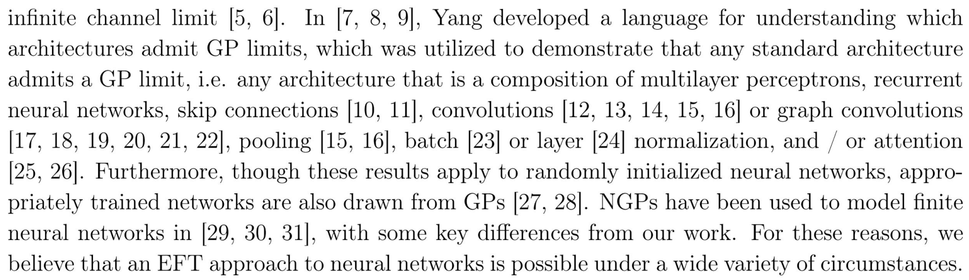
Most architectures admit GP limit

Single-layer infinite width feedforward networks are GPs. [Neal], [Williams] 1990's

Deep infinite width feedforward networks are GPs. [Lee et al., 2017], [Matthews et al., 2018]

Infinite channel CNNs are GPs. [Novak et al., 2018] [Garriga-Alonso et al. 2018]

Tensor programs show any ***standard*** architecture admits GP limit. [Yang, 2019]



infinite channel limit [5, 6]. In [7, 8, 9], Yang developed a language for understanding which architectures admit GP limits, which was utilized to demonstrate that any standard architecture admits a GP limit, i.e. any architecture that is a composition of multilayer perceptrons, recurrent neural networks, skip connections [10, 11], convolutions [12, 13, 14, 15, 16] or graph convolutions [17, 18, 19, 20, 21, 22], pooling [15, 16], batch [23] or layer [24] normalization, and / or attention [25, 26]. Furthermore, though these results apply to randomly initialized neural networks, appropriately trained networks are also drawn from GPs [27, 28]. NGPs have been used to model finite neural networks in [29, 30, 31], with some key differences from our work. For these reasons, we believe that an EFT approach to neural networks is possible under a wide variety of circumstances.

Greg's "tensor programs" (next talk?) are a language for showing, amongst other things, how general GP limits are.

GP property persists under appropriate training. [Jacot et al., 2018] [Lee et al., 2019]

NN-QFT correspondence: the essential logic

Asymptotic NNs are drawn from Gaussian as $N \to \infty$.

@ Large (but finite) N: close-to-Gaussian NN distribution,
with non-Gaussianities 1/N-suppressed.

This structure is the backbone of perturbative QFT.

Asymptotic NNs, GPs, and Free Field Theory

Gaussian Process:

distribution: $$P[f] \sim \exp\left[-\frac{1}{2}\int d^{d_{\text{in}}}x\, d^{d_{\text{in}}}x'\, f(x)\Xi(x,x')f(x')\right]$$

where: $$\int d^{d_{\text{in}}}x'\, K(x,x')\,\Xi(x',x'') = \delta^{(d_{\text{in}})}(x-x'')$$

K is the *kernel* of the GP.

log-likelihood: $$S = \frac{1}{2}\int d^{d_{\text{in}}}x\, d^{d_{\text{in}}}x' \;\, f(x)\Xi(x,x')f(x')$$

n-pt correlation functions: $$G^{(n)}(x_1,\ldots,x_n) = \frac{\int df \;\, f(x_1)\ldots f(x_n)\, e^{-S}}{Z}$$

Free Field Theory:

“free” = non-interacting

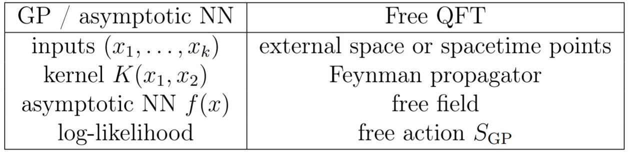
Feynman path integral:

$$Z = \int D\phi\, e^{-S[\phi]}$$

From P.I. perspective, free theories are Gaussian distributions on field space.

e.g., free scalar field theory

$$S[\phi] = \int d^dx\, \phi(x)(\Box + m^2)\phi(x)$$



GP / asymptotic NN	Free QFT
inputs $(x_1,\ldots,x_k)$	external space or spacetime points
kernel $K(x_1,x_2)$	Feynman propagator
asymptotic NN $f(x)$	free field
log-likelihood	free action S_{GP}

Large N Neural Networks, NGPs, and Interacting QFT

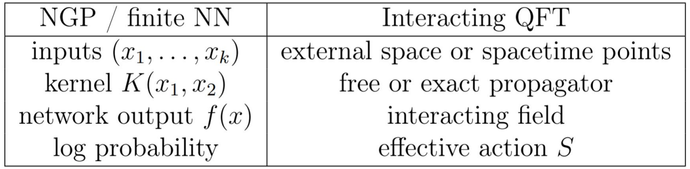
Punchline: finite N networks that admit a GP limit should be drawn from non-Gaussian process. (NGP)

$$S = S_{\mathrm{GP}} + \Delta S$$

where, e.g., could have:

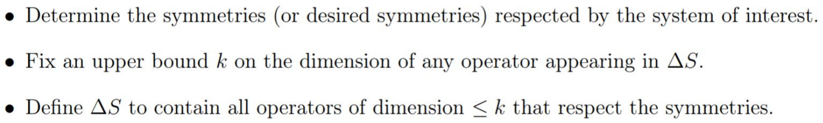
$$\Delta S = \int d^{d_{\mathrm{in}}}x \; \left[g\, f(x)^3 + \lambda\, f(x)^4 + \alpha\, f(x)^5 + \kappa\, f(x)^6 + \ldots \right]$$

such non-Gaussian terms are interactions in QFT. their coefficients = "couplings."



NGP / finite NN	Interacting QFT
inputs $(x_1, \ldots, x_k)$	external space or spacetime points
kernel $K(x_1, x_2)$	free or exact propagator
network output $f(x)$	interacting field
log probability	effective action S

Wilsonian EFT for NGPs:



- Determine the symmetries (or desired symmetries) respected by the system of interest.
- Fix an upper bound k on the dimension of any operator appearing in ΔS.
- Define ΔS to contain all operators of dimension $\leq k$ that respect the symmetries.

determines NGP "effective action" = log likelihood. Some art in this, but done for decades by physicists.

Experiments below: single-layer finite width networks

$$S = S_{\mathrm{GP}} + \int d^{d_{\mathrm{in}}}x \;\; \left[\lambda\, f(x)^4 + \kappa\, f(x)^6 \right]$$

odd-pt functions vanish $\longrightarrow$ odd couplings vanish.

In fact, κ more irrelevant than λ (in Wilsonian sense), can be ignored in expts. *even simpler NGP distribution.*

Given NN-QFT, how should we determine + model NN distributions?

in the real world, we compare experiments to the ***moments*** of the distribution, use to infer or approximate the NGP distribution.

i.e., we compute and measure correlation functions.

GP Predictions for Correlation Functions

if asymptotic NN drawn from GP
use Feynman diagrams for correlators.

$$G^{(n)}(x_1,\ldots,x_n) = \frac{\int df\;\, f(x_1)\ldots f(x_n)\, e^{-S}}{Z}$$

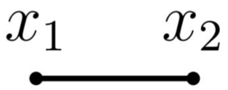
Right: analytic and Feynman diagram expressions for n-pt correlations of asymptotic NN outputs.

Physics analogy: mean-free GP is totally determined by 2-pt statistics, i.e. the GP kernel.

kernel = propagator, so GP = a QFT where all diagrams rep particles flying past each other.

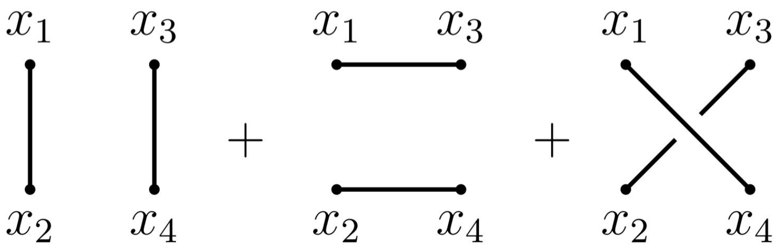
$$G^{(2)}_{\text{GP}}(x_1,x_2) \;=\; K(x_1,x_2)$$

$$=$$



$$G^{(4)}_{\text{GP}}(x_1,x_2,x_3,x_4) \;=\; K(x_1,x_2)K(x_3,x_4)$$
$$+\,K(x_1,x_3)K(x_2,x_4)+K(x_1,x_4)K(x_2,x_3)$$

$$=$$



NGP Correlation Functions from Feynman Diagrams

Correlation functions of neural network outputs defined by associated NGP distribution.

$$G^{(n)}(x_1, \dots, x_n) = \frac{\int df \;\, f(x_1) \dots f(x_n)\, e^{-S}}{Z_0}$$

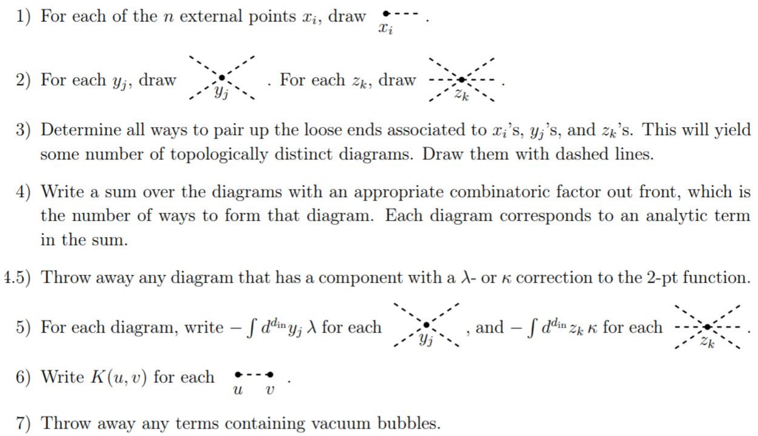
use usual physics trick

$$= \frac{\int df \;\, f(x_1) \dots f(x_n) \left[1 - \int d^{d_{\text{in}}}x \, g_k f(x)^k + O(g_k^2)\right] \; e^{-S_{\text{GP}}}/Z_{\text{GP},0}}{\int df \;\; \left[1 - \int d^{d_{\text{in}}}x \, g_k f(x)^k + O(g_k^2)\right] e^{-S_{\text{GP}}}/Z_{\text{GP},0}}$$

to compute diagrammatically as Feynman diagrams.

Essentials from QFT reviewed in paper, e.g. cancellation of “vacuum bubbles” (components with no external points) by expanding the denominator.

Feynman Rules:



1) For each of the n external points x_i, draw .

2) For each y_j, draw . For each z_k, draw .

3) Determine all ways to pair up the loose ends associated to x_i's, y_j's, and z_k's. This will yield some number of topologically distinct diagrams. Draw them with dashed lines.

4) Write a sum over the diagrams with an appropriate combinatoric factor out front, which is the number of ways to form that diagram. Each diagram corresponds to an analytic term in the sum.

4.5) Throw away any diagram that has a component with a λ- or κ correction to the 2-pt function.

5) For each diagram, write $-\int d^{d_{\text{in}}}y_j \, \lambda$ for each , and $-\int d^{d_{\text{in}}}z_k \, \kappa$ for each .

6) Write $K(u,v)$ for each .

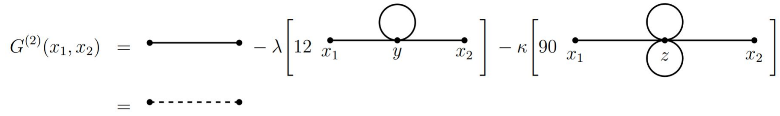
7) Throw away any terms containing vacuum bubbles.

these rules are a picture to analytic expression dictionary.

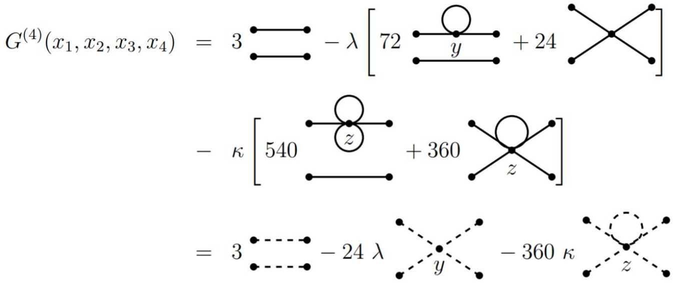
note: in our experiments, GP kernel happens to be exact all-width 2-pt function.

2-pt, 4-pt, and 6-pt Correlation Functions

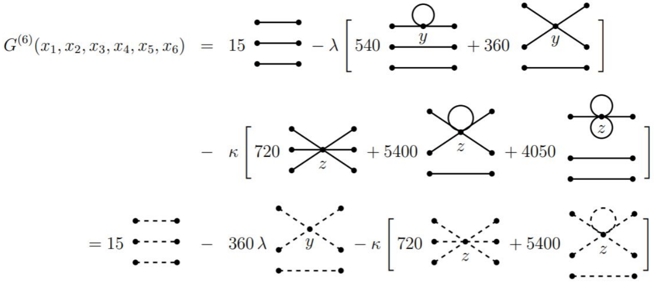
point: theory equations that actually enter our NN codes.



$$= \quad K(x_1, x_2), \qquad (3.17)$$



$$
\begin{aligned}
&= K(x_1,x_2)K(x_3,x_4) + K(x_1,x_3)K(x_2,x_4) + K(x_1,x_4)K(x_2,x_3) \\
&- 24 \int d^{d_{\text{in}}}y \, \lambda \, K(x_1,y)K(x_2,y)K(x_3,y)K(x_4,y) \\
&- 360 \int d^{d_{\text{in}}}z \, \kappa \, K(x_1,z)K(x_2,z)K(x_3,z)K(x_4,z)K(z,z) \qquad (3.18)
\end{aligned}
$$



$$
\begin{aligned}
= \; & \Big[K_{12}K_{34}K_{56} + K_{12}K_{35}K_{46} + K_{12}K_{36}K_{45} + K_{13}K_{24}K_{56} + K_{13}K_{25}K_{46} + K_{13}K_{26}K_{45} + K_{14}K_{23}K_{56} \\
+ \; & K_{14}K_{25}K_{36} + K_{14}K_{26}K_{35} + K_{15}K_{23}K_{46} + K_{15}K_{24}K_{36} + K_{15}K_{26}K_{34} + K_{16}K_{23}K_{45} + K_{16}K_{24}K_{35} \\
+ \; & K_{16}K_{25}K_{34}\Big] - 24 \int d^{d_{\text{in}}}y \, \lambda \Big[K_{1y}K_{2y}K_{3y}K_{4y}K_{56} + K_{1y}K_{2y}K_{3y}K_{5y}K_{46} + K_{1y}K_{2y}K_{4y}K_{5y}K_{36} \\
+ \; & K_{1y}K_{3y}K_{4y}K_{5y}K_{26} + K_{2y}K_{3y}K_{4y}K_{5y}K_{16} + K_{1y}K_{2y}K_{3y}K_{6y}K_{45} + K_{1y}K_{2y}K_{4y}K_{6y}K_{35} \\
+ \; & K_{1y}K_{3y}K_{4y}K_{6y}K_{25} + K_{2y}K_{3y}K_{4y}K_{6y}K_{15} + K_{1y}K_{2y}K_{5y}K_{6y}K_{34} + K_{1y}K_{3y}K_{5y}K_{6y}K_{24} \\
+ \; & K_{2y}K_{3y}K_{5y}K_{6y}K_{14} + K_{1y}K_{4y}K_{5y}K_{6y}K_{23} + K_{2y}K_{4y}K_{5y}K_{6y}K_{13} + K_{3y}K_{4y}K_{5y}K_{6y}K_{12}\Big] \\
- \; & 720 \int d^{d_{\text{in}}}z \, \kappa \, K_{1z}K_{2z}K_{3z}K_{4z}K_{5z}K_{6z} - 360 \int d^{d_{\text{in}}}z \, \kappa \Big[K_{zz}K_{1z}K_{2z}K_{3z}K_{4z}K_{56} \\
+ \; & K_{zz}K_{1z}K_{2z}K_{3z}K_{5z}K_{46} + K_{zz}K_{1z}K_{2z}K_{4z}K_{5z}K_{36} + K_{zz}K_{1z}K_{3z}K_{4z}K_{5z}K_{26} \\
+ \; & K_{zz}K_{2z}K_{3z}K_{4z}K_{5z}K_{16} + K_{zz}K_{1z}K_{2z}K_{3z}K_{6z}K_{45} + K_{zz}K_{1z}K_{2z}K_{4z}K_{6z}K_{35} \\
+ \; & K_{zz}K_{1z}K_{3z}K_{4z}K_{6z}K_{25} + K_{zz}K_{2z}K_{3z}K_{4z}K_{6z}K_{15} + K_{zz}K_{1z}K_{2z}K_{5z}K_{6z}K_{34} \\
+ \; & K_{zz}K_{1z}K_{3z}K_{5z}K_{6z}K_{24} + K_{zz}K_{2z}K_{3z}K_{5z}K_{6z}K_{14} + K_{zz}K_{1z}K_{4z}K_{5z}K_{6z}K_{23} \\
+ \; & K_{zz}K_{2z}K_{4z}K_{5z}K_{6z}K_{13} + K_{zz}K_{3z}K_{4z}K_{5z}K_{6z}K_{12}\Big], \qquad (3.19)
\end{aligned}
$$

When Correlators Diverge: Wilsonian RG in NN-QFT

Experiments: the central insight in renormalization.

[Zee] for beautiful textbook discussion.

Evaluate set of NNs on inputs

$$S_{\text{in}} = \{x_1, \ldots, x_{N_{\text{in}}}\}$$

$$|x_i| \ll \Lambda$$

and measure experimental correlation functions,

$$G^{(n)}(x_1, \ldots, x_n) = \frac{1}{n_{\text{nets}}} \sum_{\alpha \in \text{nets}}^{n_{\text{nets}}} f_\alpha(x_1) \ldots f_\alpha(x_n)$$

these just are what they are! One set of corr fns.

Goal of theory is to explain them.

Theory: put cutoff in NGP corrections

$$\Delta S_\Lambda = \int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}}x \sum_{l \leq k} g_{\mathcal{O}_l}(\Lambda)\, \mathcal{O}_l$$

Λ finite puts input in box, regulates divergences. For any Λ sufficiently big, measure couplings, make predictions, verify with experiments.

Infinite number of S_Λ that are supposed to work, but only one set of experiments. **Requires:**

$$\frac{dG^{(n)}(x_1, \ldots, x_n)}{d\Lambda} = 0$$

allows for extraction of β-functions.

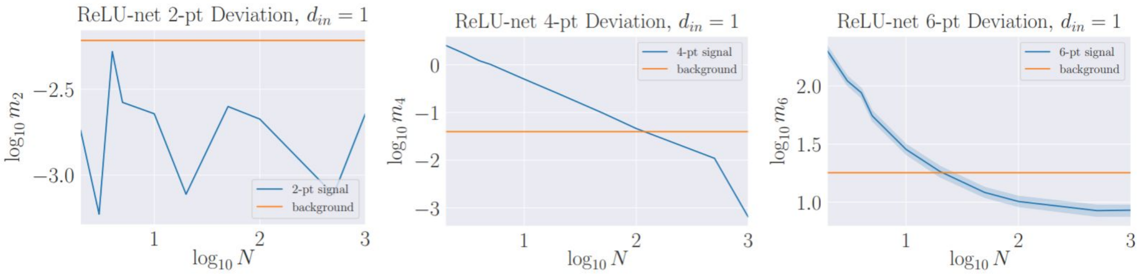
A Flash of Some Experimental Results

Experimental description

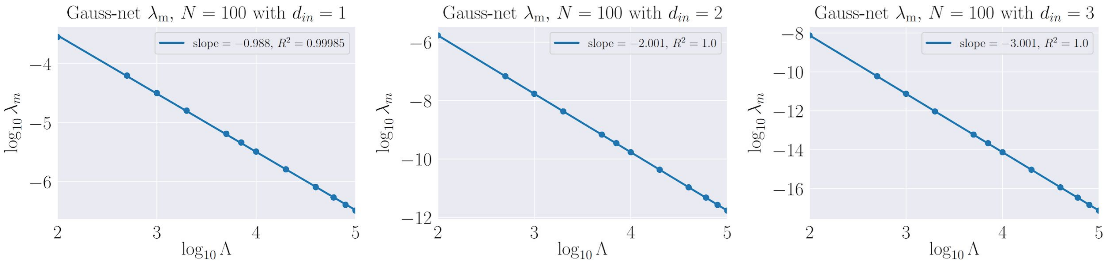
Experiments in three different single-layer networks, with ReLU, Erf, and a custom “GaussNet” activation.

Drew millions of models and evaluated on fixed sets of input to do experiments with correlators and the EFT description of NN distribution.

NGP correlators become GP correlators as N → ∞



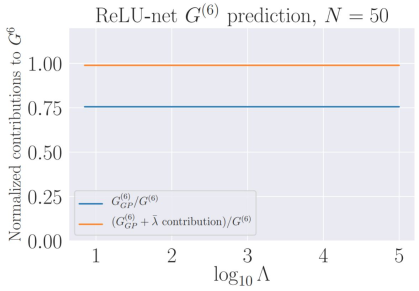
Dependence of Quartic Coupling on Cutoff



$$\beta(\lambda) := \frac{\partial \lambda}{\partial \log \Lambda} = -\lambda \, d_{\text{in}}$$

Depends on input dimension.
See quartic is **asymptotically free.**

Verification of EFT Predictions

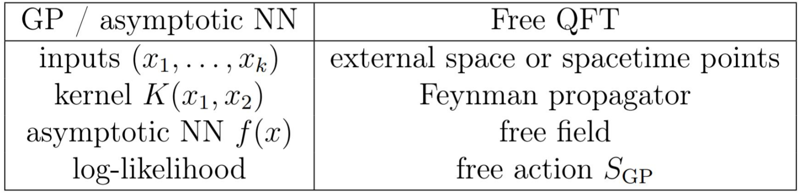


Perturbative correction in measured quartic coupling bring 6-pt function closer to experiment.

Summary of Results from First Paper

central idea: model NGP / NN distribution using QFT techniques, e.g. Wilsonian EFT.

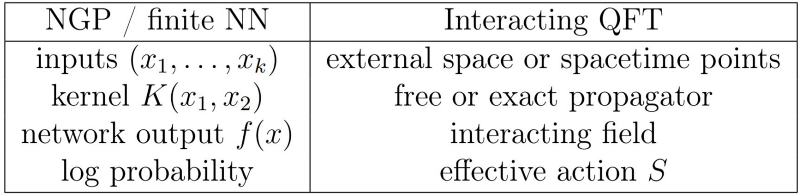
asymptotic NN's "=" Free QFT



GP / asymptotic NN	Free QFT
inputs $(x_1, \ldots, x_k)$	external space or spacetime points
kernel $K(x_1, x_2)$	Feynman propagator
asymptotic NN $f(x)$	free field
log-likelihood	free action S_{GP}

b/c drawn from GPs

NNs "=" QFT



NGP / finite NN	Interacting QFT
inputs $(x_1, \ldots, x_k)$	external space or spacetime points
kernel $K(x_1, x_2)$	free or exact propagator
network output $f(x)$	interacting field
log probability	effective action S

b/c drawn from NGPs

fairly general: any "standard architecture" (Yang) admits a GP limit. persists under some training.

therefore, away from limit, NGP. use EFT to model. import QFT ideas directly into NNs.

QFT treatment of NN distribution yields:

1) output correlation functions as Feynman diagrams.
2) measure couplings in experiments, predict, verify.
3) Wilsonian RG induces flow in couplings, simplifies the model of the NN distribution.

Verified experimentally, single layer networks, indeed QFT gives function-space perspective on NNs.

What does this treatment of NNs get you?

Duality:

In physics, means two perspectives on a single system, where certain things are easier from one.

Parameter-space / function-space duality:
at large N, parameter-space complexity explodes.

but in function-space complexity decreases due to renorm. and 1/N suppression of non-Gaussianities.

Acute example: *single number* in NGP dist. was sufficient to approximate NGP corrections, despite losing an ∞ number of parameters in moving away from GP.

Training:

Our formalism only requires being “close” to GP, where measure of closeness determined experimentally and in examples is relatively low N.

Some training preserves GP at large N, in principle allowing QFT treatment of NGP during training.

Supervised learning:
in QFT language, it is just learning the 1-pt function.

in general this will break symmetry of NGP (see paper next week for priors), bring in even more QFT.

Outlook and Motivation

Most NN’s at moderate N are drawn from non-Gaussian processes (NGPs).
Perturbative QFT = NGP + extra widgets / axioms.

Which NNs, if any, satisfy the extra axioms?

Which techniques that we love from QFT apply
even in the absence of the axioms, i.e. to general NGPs / NNs?

Thanks!

Questions?

Please feel free to get in touch:

e-mail: jhh@neu.edu
Twitter: @jhhalverson
web: www.jhhalverson.com

Our examples and their kernels

Erf-net: $\sigma(z) = \mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z dt\, e^{-t^2}$

$$K_{\mathrm{Erf}}(x, x') = \sigma_b^2 + \sigma_W^2\, \frac{2}{\pi} \arcsin\left[\frac{2(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x x')}{\sqrt{\left(1 + 2(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x^2)\right)\left(1 + 2(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x'^2)\right)}}\right]$$

Gauss-net: $\sigma(x) = \frac{\exp\left(W\, x + b\right)}{\sqrt{\exp\left[2(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}} x^2)\right]}}$

$$K_{\mathrm{Gauss}}(x, x') = \sigma_b^2 + \sigma_W^2\; \exp\left[-\frac{\sigma_W^2 |x - x'|^2}{2 d_{\mathrm{in}}}\right]$$

ReLU-net: $\sigma(z) = \begin{cases} 0 & z < 0 \\ z & z \geq 0 \end{cases}$

$$\begin{aligned} K_{\mathrm{ReLU}}(x, x') &= \sigma_b^2 + \sigma_W^2\, \frac{1}{2\pi} \sqrt{(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x \cdot x)(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x' \cdot x')}(\sin\theta + (\pi - \theta)\cos\theta), \\ \theta &= \arccos\left[\frac{\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x \cdot x'}{\sqrt{(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x \cdot x)(\sigma_b^2 + \frac{\sigma_W^2}{d_{\mathrm{in}}}\, x' \cdot x')}}\right], \end{aligned}$$