

Neural Networks and Quantum Field Theory

Jim Halverson

Seminar in **Mathematics**, Physics, and Machine Learning
@ Instituto Superior Tecnico, Lisbon

Northeastern
University

Based on: 2008.08601 with Maiti and Stoner



The Gang



Anindita Maiti
targeting physics postdocs, Fall 2022



Keegan Stoner
targeting ML labs, Fall 2022

+ Three Connections @ Physics / ML Interface

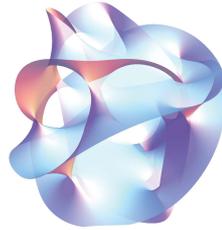


Institute for Artificial Intelligence and Fundamental Interactions (IAIFI)

one of five new NSF AI research institutes, this one at the interface with physics! MIT, Northeastern, Harvard, Tufts.

ML for physics / math discoveries?
Can physics / math help ML?

Colloquia begin in Spring!
www.iaifi.org.



Physics \cap ML

Physics Meets ML

virtual seminar series at the interface, “continuation” of 2019 meeting at Microsoft Research.

Bi-weekly seminars from physicists and CS, academia and industry.

Sign up at www.physicsmeetsml.org.



Feel free to contact me!

e-mail: jhh@neu.edu
Twitter: [@jhhalverson](https://twitter.com/jhhalverson)
web: www.jhhalverson.com

ML for Math:
e.g. “Learning to Unknot”: 2010.16263

ML for Strings:
e.g. “Statistical Predictions in String Theory and Deep Generative Models”: 2001.00555

What is learning?

Legend:

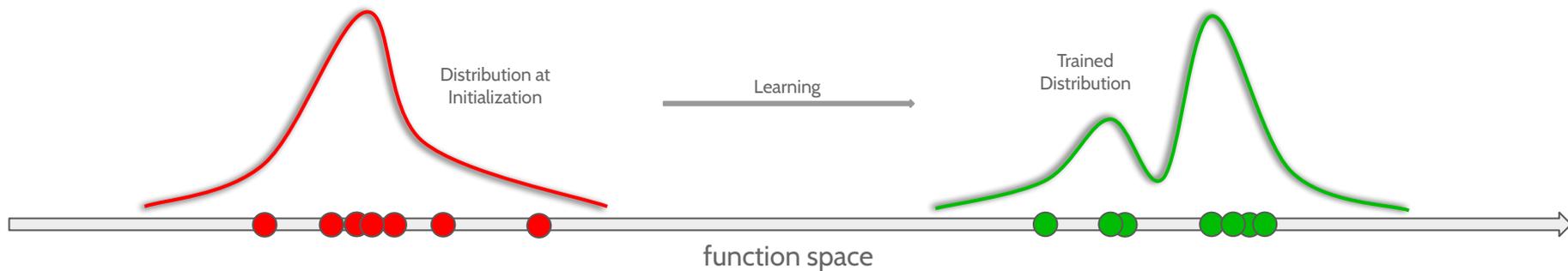
- Randomly initialized NN
- Trained NN

Physics Language:

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

Bayesian Language:

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



Then what is supervised learning?

the evolution of the 1-pt function $E[f]$ until convergence.

Outline

- **What is QFT?** (physically? origin of Feynman diagrams. statistically?)
- **NN-QFT Correspondence:** model NN distributions with QFT techniques
 - i) asymptotic NNs, GPs, and free field theory
 - ii) NNs, non-GPs and Wilsonian “effective” field theory.
 - iii) renormalization: removes divergences in higher correlators, simplifies NN dist.
- **Experiments:** verify approach in simple examples
- **Discussion and Outlook:** parameter-space / function-space duality, training

What is QFT?

physically?

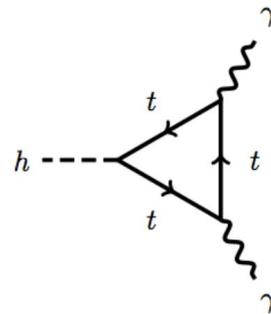
what are Feynman diagrams?

statistically?

What is QFT, physically?

- quantum theory of fields, and their particle excitations.
- for both fundamental particles, (e.g. Higgs) and quasiparticles (e.g. in superconductors)
- **a single QFT** predicts radioactive decay rates, strength of particle scattering, etc.
- two main perspectives:
“canonical quantization” (bra-ket approach)
Feynman’s path integral (today).
- **Many** Nobel prizes. (Could easily rattle off 5-10?)

Example: Higgs boson discovery



The QFT = Standard Model (SM) of Particle Phys.

2012: Discovered Higgs boson at CERN, e.g., in diphoton channel @ left.

Amazing science press.

2013: Nobel to Higgs, Englert.

Origin of Feynman diagrams?

Pictures useful for computing moments of Gaussian or near-Gaussian distributions

Example: Gaussian Moments

$$\langle x^{2n} \rangle = \frac{\int_{-\infty}^{+\infty} dx \exp(-\frac{1}{2}ax^2) x^{2n}}{\int_{-\infty}^{+\infty} dx \exp(-\frac{1}{2}ax^2)} = \frac{1}{a^n} (2n-1)!!$$

$$\begin{aligned} \langle x^4 \rangle &= \begin{array}{c} | \quad | \\ | \quad | \\ | \quad | \end{array} + \begin{array}{c} \text{---} \\ \text{---} \end{array} + \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} \\ &= \left(\frac{1}{a} \cdot \frac{1}{a}\right) + \left(\frac{1}{a} \cdot \frac{1}{a}\right) + \left(\frac{1}{a} \cdot \frac{1}{a}\right) = \frac{3}{a^2} \end{aligned}$$

Feynman rules: a picture-expression dictionary

Example: Near-Gaussian Moments via Perturbation Theory

$$\begin{aligned} \langle x^{2n} \rangle &= \frac{\int dx \exp(-\frac{1}{2}ax^2 + \lambda x^4) x^{2n}}{\int dx \exp(-\frac{1}{2}ax^2 + \lambda x^4)} \\ &= \frac{\sum_{k=0}^{\infty} \frac{\lambda^k}{k!} \int dx \exp(-\frac{1}{2}ax^2) x^{2n+4k}}{\sum_{k=0}^{\infty} \frac{\lambda^k}{k!} \int dx \exp(-\frac{1}{2}ax^2)} \cdot \frac{\int dx \exp(-\frac{1}{2}ax^2)}{\int dx \exp(-\frac{1}{2}ax^2)} \\ &= \frac{\sum_{k=0}^{\infty} \frac{\lambda^k}{k!} \langle x^{2n+4k} \rangle_G}{\sum_{k=0}^{\infty} \frac{\lambda^k}{k!} \langle x^{4k} \rangle_G} \quad \text{for small } \lambda, \text{ truncate} \end{aligned}$$

Additions and extra widgets may arise, but **Essence:** approximate non-Gaussian moments in terms of Gaussian moments, diagrammatically.

**Sounds like QFT is physics widgets
on top of a statistics backbone.**

What is QFT, statistically?

- defined by distribution on field space, the so-called Feynman path integral. log-probability $S[\Phi]$ is “action”
- Experiments measure n-pt correlation functions and amplitudes.
- **Free QFT:** no interactions, Gaussian.
- **Perturbative QFT:** distribution is near-Gaussian, compute approximate moments perturbatively.

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

$$G^{(n)}(x_1, \dots, x_n) = \frac{1}{Z} \int D\phi \phi(x_1) \dots \phi(x_n) e^{-S[\phi]}$$

NN-QFT Correspondence

- i) asymptotic neural nets, GPs, and free QFT
- ii) finite N neural nets, non-GPs, interacting QFT
- iii) Wilsonian renormalization

A way to model NN distributions
using QFT techniques

Asymptotic Neural Networks

neural network has a discrete hyperparameter N that enters into its architecture.

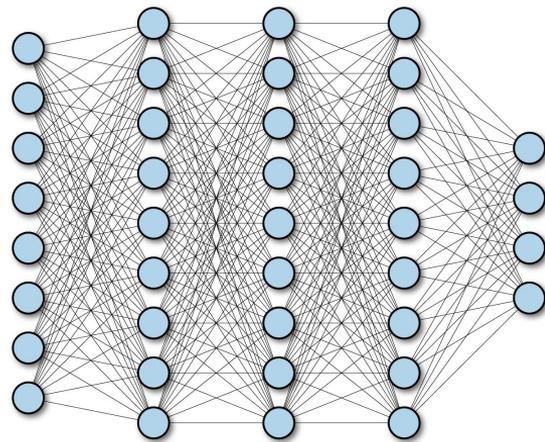
asymptotic limit = $N \rightarrow \infty$ limit

crucial property: want to add infinite number of parameters, which themselves are random variables!

example:

infinite width limit of single-layer or deep feedforward networks

$$f_{\theta, N} : \mathbb{R}^{d_{\text{in}}} \rightarrow \mathbb{R}^{d_{\text{out}}}$$



$$N \rightarrow \infty$$

NN-GP Correspondence and Central Limit Theorem

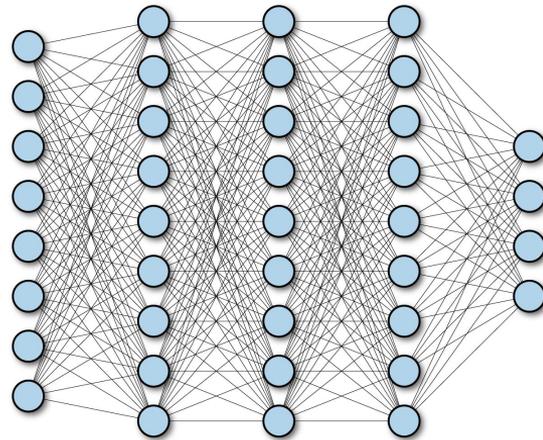
Add N iid random variables,
take $N \rightarrow \infty$,
sum is drawn from a Gaussian distribution.

If some step in a neural net does this,
that step drawn from Gaussian.

e.g., if NN output does, it's drawn from a Gaussian.

then NN is drawn from Gaussian distribution on
field space, known as a *Gaussian Process* (GP).

$$f_{\theta, N} : \mathbb{R}^{d_{\text{in}}} \rightarrow \mathbb{R}^{d_{\text{out}}}$$



$$N \rightarrow \infty$$

“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.

tons of examples cited
in our paper admit GP limits

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

Gaussian Processes 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:
$$G^{(n)}(x_1, \dots, x_n) = \frac{\int df f(x_1) \dots f(x_n) e^{-S}}{Z}$$

Crucial note:

P[f] can also have one or zero integrals, "local" and "ultra-local" cases, respectively.

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

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

GP / asymptotic NN	Free QFT
inputs (x_1, \dots, 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}

GP Predictions for Correlation Functions

if asymptotic NN drawn from GP and GP “=” free QFT, should be able to use Feynman diagrams for correlation functions.

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

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.

$$G_{\text{GP}}^{(2)}(x_1, x_2) = K(x_1, x_2) = \text{---} \text{---}$$

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

What about finite N nets?

Non-Gaussian Processes (NGPs), EFTs, and Interactions

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

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

where, e.g., could have:

$$\Delta S = \int d^{d_{\text{in}}} x [g f(x)^3 + \lambda f(x)^4 + \alpha f(x)^5 + \kappa f(x)^6 + \dots]$$

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

NGP / finite NN	Interacting QFT
inputs (x_1, \dots, 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

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

odd-pt functions vanish \rightarrow odd couplings vanish.

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

NGP Correlation Functions from Feynman Diagrams

Correlation functions defined by 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 $\bullet \cdots \cdots$.
- 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 $\bullet \cdots \cdots \bullet$.
- 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.

At this point you should object!

(very impressive attention to detail if you actually did.)

Input space integrals often diverge at large input.

QFT prescription: “regularization.”

Various varieties, we use a “hard cutoff” Λ , replace

$$S \rightarrow S_\Lambda$$

so any input integral is over a box of size Λ .

Making sense of divergences: Renormalization

Experiments: the central insight in renormalization.

[Zee] for beautiful textbook discussion.

Evaluate set of NNs on inputs

$$\mathcal{S}_{\text{in}} = \{x_1, \dots, x_{N_{\text{in}}}\}$$

and measure $\exp\left[-\int d^d x \mathcal{L}(x)\right]$ on functions,

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

Goal of theory is to explain them.

Theory: NGP action corrects GP action by

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

the old S had $\Lambda \rightarrow \infty$ and computing n-pt gives divergences. Λ finite regulates those divergences, input is now in a box.

For any Λ sufficiently big, measure couplings, make predictions, verify with experiments.

But there's an infinite number of S_{Λ} , and only one set of experiments for them to describe!

How does this make sense?

Essence of Renormalization

the infinity of effective actions must make the same experimental predictions, requiring, e.g.

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

Extracting β -functions from theory

NN effective actions (distributions) with different Λ may make the same predictions by absorbing the difference into couplings, “**running couplings.**”

$$\beta(g_{\mathcal{O}_i}) := \frac{d g_{\mathcal{O}_i}}{d \log \Lambda}$$

Encoded in the β -functions, which capture how the couplings vary with the cutoff.

Induces a “flow” in coupling space as Λ varies, **Wilsonian renormalization group flow.** (RG)

Extract from hitting n-pt functions with derivatives.

$$\begin{aligned} \frac{\partial G^{(4)}(x_1, x_2, x_3, x_4)}{\partial \log \Lambda} &= 0 = \frac{\partial \lambda}{\partial \log \Lambda} \int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{4,\lambda} + \varrho_{4,\lambda}) + \lambda \frac{\partial (\int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{4,\lambda} + \varrho_{4,\lambda}))}{\partial \log \Lambda} \\ &+ \frac{\partial \kappa}{\partial \log \Lambda} \int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{4,\kappa} + \varrho_{4,\kappa}) + \kappa \frac{\partial (\int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{4,\kappa} + \varrho_{4,\kappa}))}{\partial \log \Lambda}, \quad (4.13) \end{aligned}$$

$$\begin{aligned} \frac{\partial G^{(6)}(x_1, x_2, x_3, x_4, x_5, x_6)}{\partial \log \Lambda} &= 0 = \frac{\partial \lambda}{\partial \log \Lambda} \int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{6,\lambda} + \varrho_{6,\lambda}) + \lambda \frac{\partial (\int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{6,\lambda} + \varrho_{6,\lambda}))}{\partial \log \Lambda} \\ &+ \frac{\partial \kappa}{\partial \log \Lambda} \int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{6,\kappa} + \varrho_{6,\kappa}) + \kappa \frac{\partial (\int_{-\Lambda}^{\Lambda} d^{d_{\text{in}}} x (\gamma_{6,\kappa} + \varrho_{6,\kappa}))}{\partial \log \Lambda} \quad (4.14) \end{aligned}$$

Our examples:

κ more irrelevant than λ , in sense of Wilson.

Means as Λ gets large, κ goes to zero faster than λ , so you can ignore it.

Extract β -function for λ from deriv. of 4-pt.

Experiments

in single-layer networks

- i) sanity check fall-off to GP at large N
- ii) measure 4-pt non-Gaussianities, predict 6-pt and verify
- iii) cutoff dependence of 4-t non-Gaussianities verifies renormalization

Controlling correlators with $1/N$ -expansion

Subtlety of Networks with Linear Output: Independence

- any network with linear output looks like:

$$f = f_b + f_w$$

- the bias and weight terms are independent draws from different processes!

QFT language: two fields that don't interact with each other.

- For Gaussian bias, f_b is drawn from constant GP for all N .

However, f_w is from NGP at finite N , GP only in asymptotic limit.

QFT language: f_w has self-interactions

Therefore: must use NGP for f_w , but not f_b .

Experiments with single-layer networks

Erf-net: $\sigma(z) = \text{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z dt e^{-t^2}$ $K_{\text{Erf}}(x, x') = \sigma_b^2 + \sigma_W^2 \frac{2}{\pi} \arcsin \left[\frac{2(\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x x')}{\sqrt{\left(1 + 2(\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x^2)\right) \left(1 + 2(\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x'^2)\right)}} \right]$

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

ReLU-net: $\sigma(z) = \begin{cases} 0 & z < 0 \\ z & z \geq 0 \end{cases}$ $K_{\text{ReLU}}(x, x') = \sigma_b^2 + \sigma_W^2 \frac{1}{2\pi} \sqrt{(\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x \cdot x)(\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x' \cdot x')} (\sin \theta + (\pi - \theta) \cos \theta),$
 $\theta = \arccos \left[\frac{\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x \cdot x'}{\sqrt{(\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x \cdot x)(\sigma_b^2 + \frac{\sigma_W^2}{d_{\text{in}}} x' \cdot x')}} \right],$

Deviations from GP Predictions

$$\Delta G^{(n)}(x_1, \dots, x_n) = G^{(n)}(x_1, \dots, x_n) - G_{\text{GP}}^{(n)}(x_1, \dots, x_n)$$

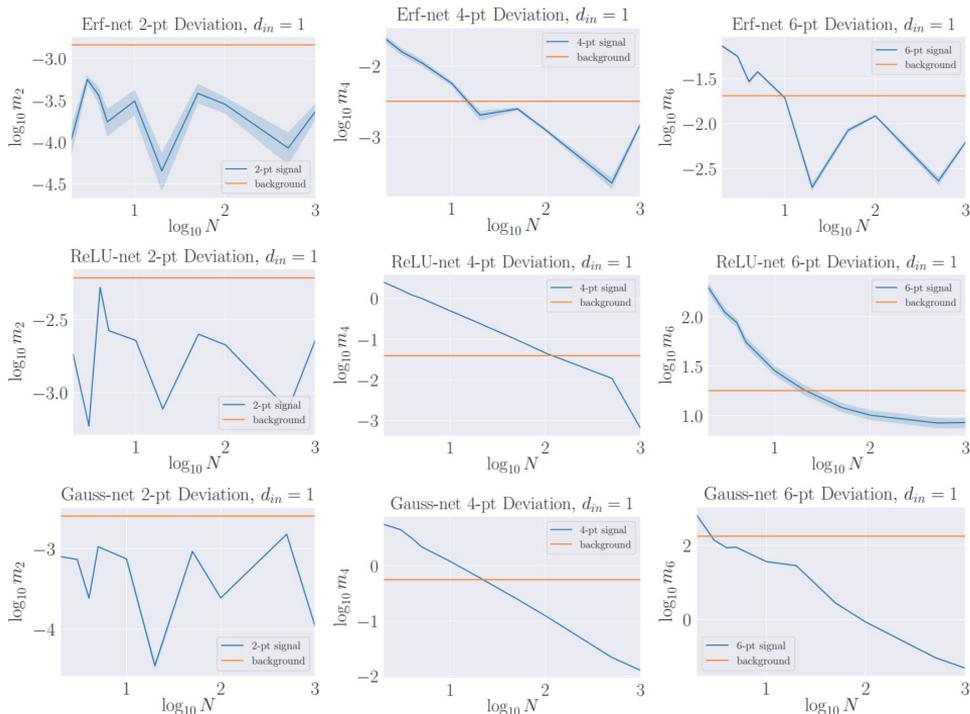
$$\Delta G^{(2)} = \frac{1}{n_{\text{nets}}} \sum_{\alpha}^{n_{\text{nets}}} f_{\alpha}(x_1)f_{\alpha}(x_2) - K(x_1, x_2)$$

$$\Delta G^{(4)} = \frac{1}{n_{\text{nets}}} \sum_{\alpha}^{n_{\text{nets}}} f_{\alpha}(x_1)f_{\alpha}(x_2)f_{\alpha}(x_3)f_{\alpha}(x_4) - \left[\begin{array}{cc} x_1 & x_3 \\ | & | \\ x_2 & x_4 \end{array} + \begin{array}{cc} x_1 & x_3 \\ \text{---} & \text{---} \\ x_2 & x_4 \end{array} + \begin{array}{cc} x_1 & x_3 \\ \diagdown & \diagup \\ x_2 & x_4 \end{array} \right]$$

$$\begin{aligned} \Delta G^{(6)} &= \frac{1}{n_{\text{nets}}} \sum_{\alpha}^{n_{\text{nets}}} f_{\alpha}(x_1)f_{\alpha}(x_2)f_{\alpha}(x_3)f_{\alpha}(x_4)f_{\alpha}(x_5)f_{\alpha}(x_6) \\ &- \left[K_{12}K_{34}K_{56} + K_{12}K_{35}K_{46} + K_{12}K_{36}K_{45} + K_{13}K_{24}K_{56} + K_{13}K_{25}K_{46} + K_{13}K_{26}K_{45} \right. \\ &+ K_{14}K_{23}K_{56} + K_{14}K_{25}K_{36} + K_{14}K_{26}K_{35} + K_{15}K_{23}K_{46} + K_{15}K_{24}K_{36} + K_{15}K_{26}K_{34} \\ &\left. + K_{16}K_{23}K_{45} + K_{16}K_{24}K_{35} + K_{16}K_{25}K_{34} \right]. \end{aligned} \quad (2.31)$$

Question: can we measure the experimental falloff to the GP prediction as $N \rightarrow \infty$?

Measuring Falloff to GP Predictions @ Large N



$$m_n = \Delta G^{(n)} / G_{GP}^{(n)}$$

Some details of the experiments:

$$N \in \{2, 3, 4, 5, 10, 20, 50, 100, 500, 1000\}$$

	inputs $\{x_i\}$	(σ_W^2, σ_b^2)
Gauss-net	$\{-0.01, -0.006, -0.002, +0.002, +0.006, +0.01\}$	(1, 1)
Erf-net	$\{-1, -0.6, -0.2, +0.2, +0.6, +1\}$	(1, 1)
ReLU-net	$\{+0.2, +0.4, +0.6, +0.8, +1.0, +1.2\}$	(1, 0)

10 experiments of 10^6 neural nets each, compute ensemble average to get correlation functions.

Background := average (across the 10 expts) standard deviation of m_n

Experimentally determined scaling: $\Delta G^{(n)} \propto N^{-1}$

Simple Extracting of Couplings from Experiments

intentionally left the coupling λ inside the interaction integrals. Only pull out if constants!

but aren't couplings constant?

e.g., physics: proton collisions have same interactions at Fermilab (Illinois) and CERN (Switzerland).

but Standard Model is T-inv. NGPs? not always.

Case, λ constant: measure from 4-pt function expts

$$\lambda = \frac{K(x_1, x_2)K(x_3, x_4) + K(x_1, x_3)K(x_2, x_4) + K(x_1, x_4)K(x_2, x_3) - G^{(4)}(x_1, x_2, x_3, x_4)}{24 \int d^{d_{\text{in}}} y K(x_1, y)K(x_2, y)K(x_3, y)K(x_4, y)}$$

call denominator integrand Δ_{1234y} .

Case, λ function: write as constant + space varying

$$\lambda(y) = \bar{\lambda} + \delta\lambda(y)$$

then we have

$$\bar{\lambda} = \frac{K_{12}K_{34} + K_{13}K_{24} + K_{14}K_{23} - G^{(4)}(x_1, x_2, x_3, x_4)}{24 \int d^{d_{\text{in}}} y \Delta_{1234y}} - \frac{\int d^{d_{\text{in}}} y \delta\lambda(y) \Delta_{1234y}}{\int d^{d_{\text{in}}} y \Delta_{1234y}}$$

and expression from before not constant,

$$\lambda_m(x_1, x_2, x_3, x_4) := \frac{K_{12}K_{34} + K_{13}K_{24} + K_{14}K_{23} - G^{(4)}(x_1, x_2, x_3, x_4)}{24 \int d^{d_{\text{in}}} y \Delta_{1234y}}$$

but when variance is small relative to mean have

$$\lambda \simeq \bar{\lambda} \simeq \text{mean}(\lambda_m(x_1, x_2, x_3, x_4))$$

our definition of “measuring λ .”

Include 4-pt Contribution to 6-pt functions

The use of EFT:

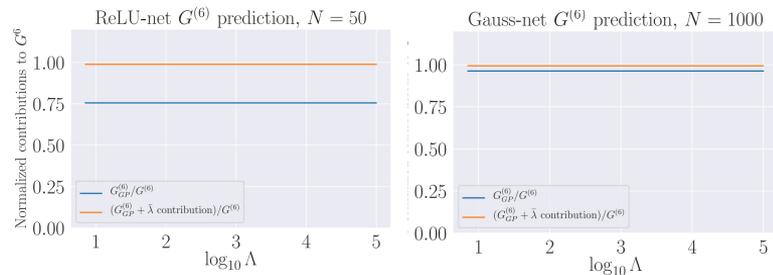
- Give a candidate ΔS for the NGP.
- Fix coefficients of operators in ΔS with experiments.
- Once fixed, make predictions for other experiments and verify them.

i.e. effective corrections from including 4-pt contr. to 6-pt.

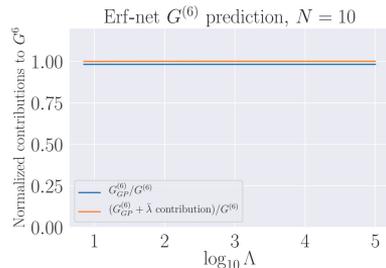
Effectiveness: Expt 6-pt - GP prediction = NGP correction

$$\begin{aligned} \delta'(x_1, \dots, x_6) &:= G^{(6)}(x_1, \dots, x_6) - \sum_{15 \text{ combinations}} \left[K(x_i, x_j) K(x_k, x_l) K(x_m, x_n) \right. \\ &\quad \left. - 24 \int d^{d_{\text{in}}} y \lambda K(x_i, y) K(x_j, y) K(x_k, y) K(x_l, y) K(x_m, x_n) \right] \\ &= -\kappa \left[720 \begin{array}{c} \bullet \quad \bullet \quad \bullet \quad \bullet \\ \diagdown \quad \diagup \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} + 5400 \begin{array}{c} \bullet \quad \bullet \quad \bullet \quad \bullet \\ \diagdown \quad \diagup \\ \bullet \quad \bullet \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \end{array} \right] \end{aligned}$$

Experimental verification:



measured λ enters
6-pt prediction,
corrects bad GP
prediction to very
close to experiment.



EFT of NN is effective!

More Precision: Quartic Coupling Predictions

Three-parameter model:

$$\Delta S_1 = \int d^{d_{\text{in}}} x (\lambda_0 + \lambda_2 x^2) f(x)^4 + \int d^{d_{\text{in}}} x d^{d_{\text{in}}} y \lambda_{\text{NL}} f(x)^2 f(y)^2$$

Three interactions:

- Constant local quartic
- Input-dependent local quartic
- Constant quartic with symmetric non-locality

Fitting the models:

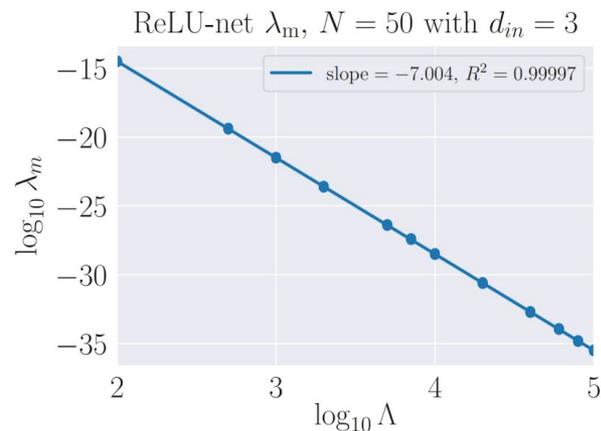
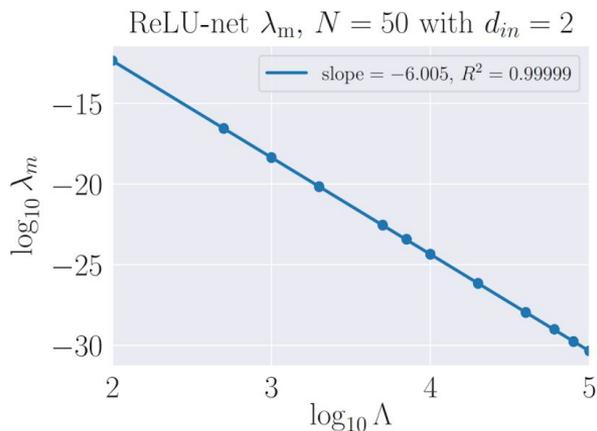
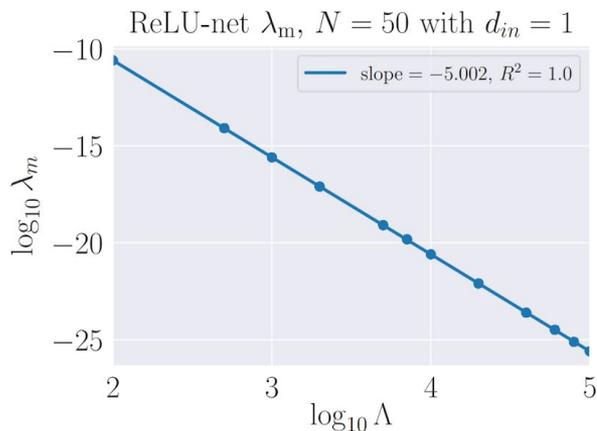
Inputs form a grid.

Train on interior, **Test** on exterior.

	$(\lambda_0, \lambda_2, \lambda_{\text{NL}})$	Test (MAPE, MSE)
Gauss M0	(0.0, 0.0, 0.0)	(100, 0.018)
Gauss M1	(0.0044, 0.0, 0.0)	(.013, 4.7×10^{-10})
Gauss M2	(0.0042, 0.001, 0.0)	(.012, 4.5×10^{-10})
Gauss M3	(0.0006, 0.0001, 0.0015)	(.016, 5.9×10^{-10})
ReLU M0	(0.0, 0.0, 0.0)	(100, 0.003)
ReLU M1	(6.3×10^{-11} , 0.0, 0.0)	(0.047, 1.6×10^{-9})
ReLU M2	(1.2×10^{-18} , 8.8×10^{-15} , 0.0)	(0.0015, 3.3×10^{-12})
ReLU M3	(1.2×10^{-18} , 8.8×10^{-15} , 6.8×10^{-17})	(0.0014, 2.8×10^{-12})

Results are **fresh**, see details in journal version.

Renormalization Theory vs. Experiment: ReLU-net



$$\beta(\lambda) : = \frac{\partial \lambda}{\partial \log(\Lambda)} = -(d_{in} + 4)\lambda$$

experimentally measured d_{in} -dependent slope matches theory
predictions from Wilsonian RG

Discussion and Outlook

summary,
parameter-space / function-space duality,
supervised learning in QFT language

Summary of Results

asymptotic NN's "=" Free QFT

GP / asymptotic NN	Free QFT
inputs (x_1, \dots, 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, \dots, 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

central idea: model NGP / NN distribution using Wilsonian effective field theory. (EFT)

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.

EFT treatment of NN distribution yields:

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

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

What does this treatment 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 6-pt corrections, despite losing an ∞ number of params in moving 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.

Thanks!

Questions?

And seriously, feel free to get in touch:

e-mail: jhh@neu.edu

Twitter: @jhhaverson

web: www.jhhaverson.com

Constants or Functions? Use Technical Naturalness

Recalling that

$$\lambda(x) = \bar{\lambda} + \delta\lambda(x)$$

when should spatially varying $\delta\lambda(x)$ be small?

Because sometimes QFT corrections to an arbitrary coupling g give:

$$g \rightarrow g + \Delta g \quad \text{with} \quad \frac{\Delta g}{g} \gg 1$$

but sometimes

$$\Delta g = 0 \text{ if } S \text{ has a symmetry.}$$

or at least in that case it might be small.

A principle of 't Hooft:

Technical Naturalness: a coupling g appearing in ΔS may be small relative to Λ if a symmetry is restored when g is set to zero.

it's true sometimes in physics, e.g. $\Delta m_e \propto m_e$ is small relative to weak scale, protected by chiral symmetry that is restored when e^- mass goes to zero.

Applied to our case? i.e., when is $\frac{\delta\lambda}{\lambda} \ll 1$ for NNs?

Conjecture: couplings in NGPs associated to neural network architectures are constants (or nearly constants) if the kernel $K(x, y)$ associated with their GP limit is translationally invariant.

holds true in our cases, GP kernel of Gauss-net is the only T-inv't one, and only example with coupling *constants*.