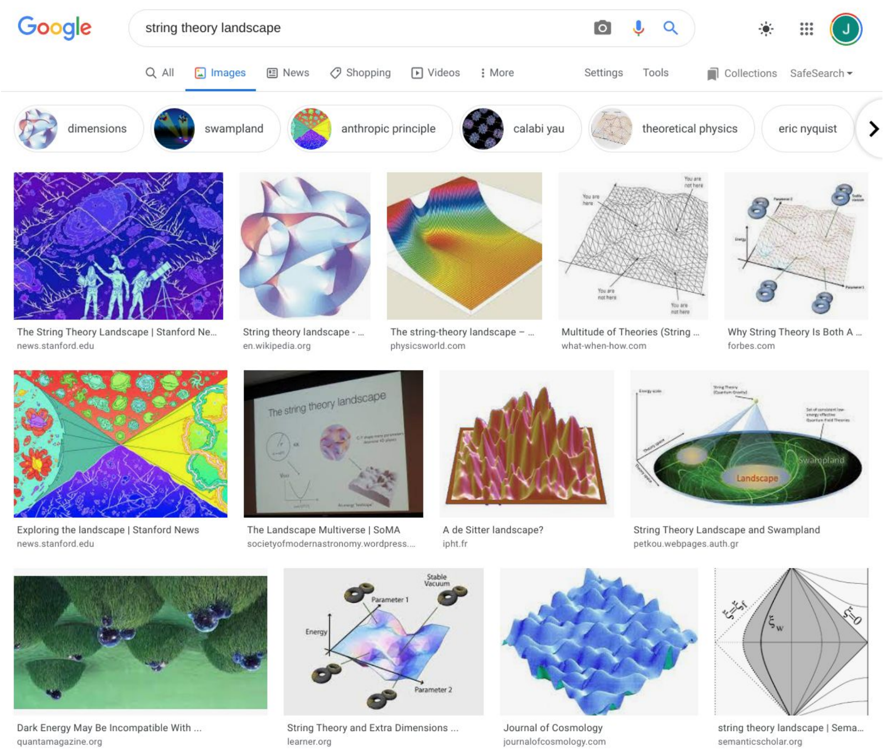
Deep Learning Landscapes

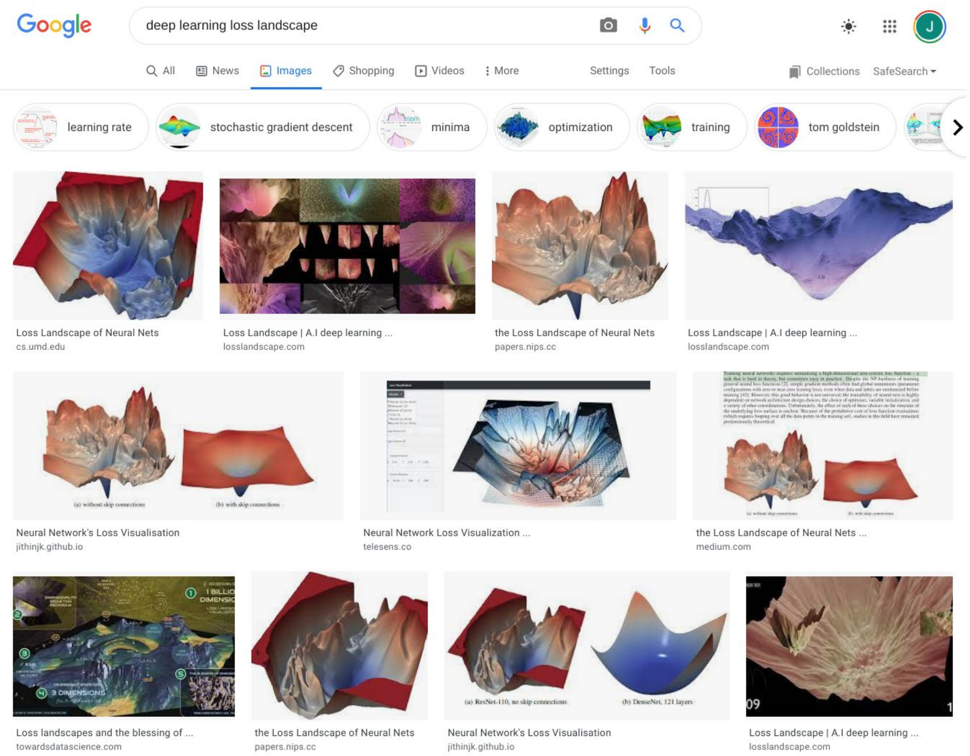
Jim Halverson

Northeastern University

PHENO 2021

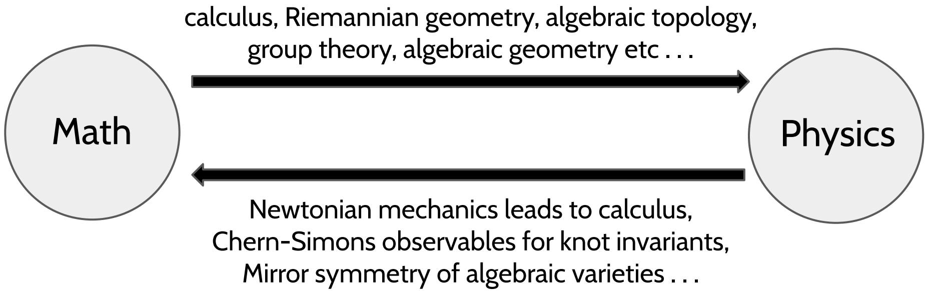
Deep Learning Landscapes: 10,000 Foot View



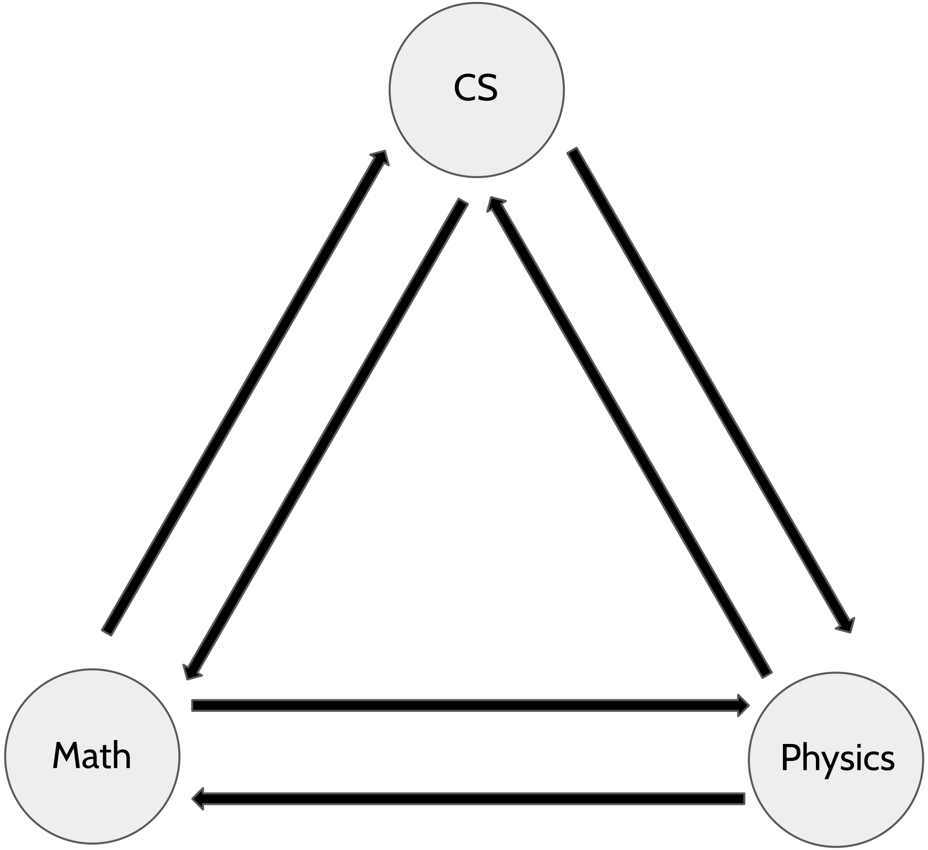


Landscapes are ubiquitous, important, and hard.

hat tip also @ spin glasses, protein folding.



The Usual Story: Math and Physics



CS: The New Kid in Town

New Developments in Deep Learning
and Quantum Information

Connecting @ Physics / ML Interface

Physics ∩ ML

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?

Sign up for our mailing list: www.iaifi.org.

Physics Meets ML

virtual seminar series, "continuation" of 2019 meeting at Microsoft Research.

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

Organizers: Bahri (Google), Krippendorf (LMU Munich), J.H., Paganini (DeepMind), Ruehle (CERN), Shiu (Madison), Yang (MSR)

Sign up at www.physicsmeetsml.org.

Feel free to contact me!

e-mail: jhh@neu.edu
Twitter: @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

But this is PHENO 2021

and so I'll focus on PHENO aspects of these subjects.

Best takeaways from the string landscape in 2021?

Can we use pheno to understand neural networks?

This Talk: Two Main Points

Part 1) String Pheno in 2021

Takeaway: draw vacua V ~ U(known string constructions), many ALPs, many gauge sectors, light and weakly coupled when controlled.

Potentially detectable remnant DOF are everywhere → problems and opportunities.

Part 2) Neural Network Pheno

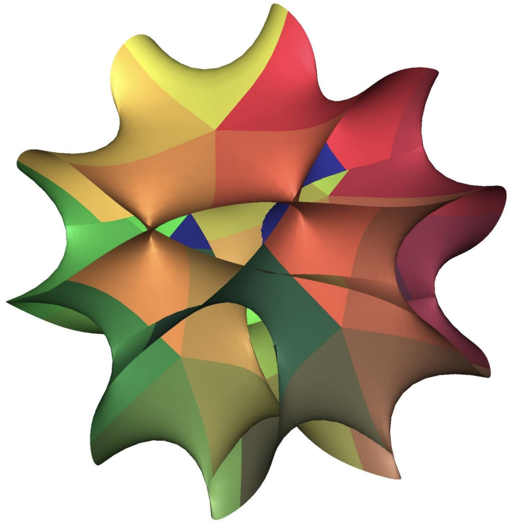
Takeaway: neural networks are random functions from nearly-Gaussian dists. *This is like in particle physics!* Model non-Gaussianities via QFT.

Use duality to determine symmetries of neural net effective actions.

Part 1) String Pheno in 2021

a broad view of what we know about the String Landscape.

The String Landscape



- quantum theory of gravity, candidate TOE.
  can give rise to semi-realistic particle-cosmo.

- extra space dimensions → compactify.
  geometry and topology determine 4d physics.

- **Landscape:** many solutions / vacua of theory.
  e.g. Kreuzer-Skarke CY3s, or 10^{755} F-theory geometries.
  e.g. $10^{272,000}$ fluxes on single geometry.

- Bubble nucleation → predictions are statistical.
  Idea: dynamics affect statistics. anthropics, too?

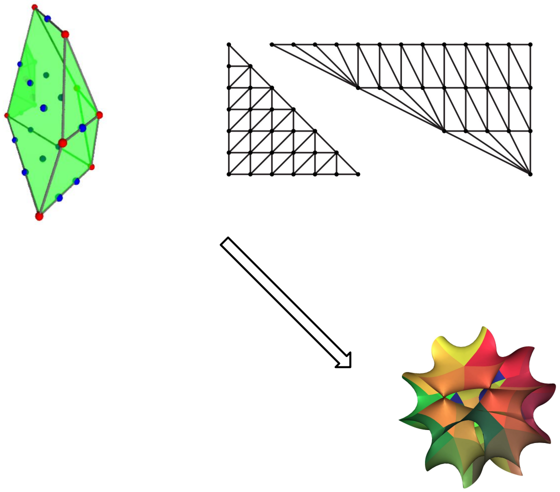
Don’t know the right distribution on vacua.

But if we drew from a uniform distribution,
given what we know in 2021, what would we find?

(Caveat: the broader we go, the fewer details are worked out.
Think impressionism, not hi-res photos).

Largest Ensemble of Concrete Visible Sectors

A Quadrillion "Standard Models"



Result: 10^{15} provably distinct, fully-consistent F-theory compactifications with exact chiral spectrum of the MSSM

[Cvetic, J.H., Lin, Liu, Tian] 2019

But They're Constrained

The following cannot all be true:

1) our vacuum is lives in this (or related) ensemble.

2) the theory is controlled (SUGRA approx.)

3) No additional dark sectors on seven-branes.

Crucial to 1) is the correct SM gauge couplings.

Upshot: There's no free lunch, we can't just tune the string EFTs at will to whatever we want.

[Cvetic, J.H., Lin, Liu, Tian] 2019

It's a big world out there: F-theory

Geometry with the Most Flux Vacua

from precision knowledge of elliptic CY4, we think we know the geometry with the most flux vacua.

$$\mathcal{O}(10^{272,000})$$

Far eclipses original 10^{500}!

But (before fluxes), minimal geometric gauge group

$$G_{\max} = E_8^9 \times F_4^8 \times (G_2 \times SU(2))^{16}$$

and also many ALPs. **Complicated cosmology.**

[Taylor, Wang] 2015

Largest Concrete Ensemble of Geometries

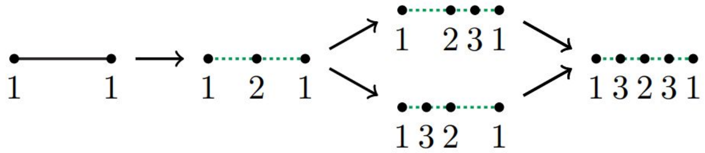
On Algorithmic Universality in F-theory Compactifications

James Halverson, Cody Long, and Benjamin Sung
Department of Physics, Northeastern University
Boston, MA 02115-5000 USA

(Dated: June 9, 2017)

We study universality of geometric gauge sectors in the string landscape in the context of F-theory compactifications. A finite time construction algorithm is presented for $\frac{4}{3} \times 2.96 \times 10^{755}$ F-theory geometries that are connected by a network of topological transitions in a connected moduli space. High probability geometric assumptions uncover universal structures in the ensemble without explicitly constructing it. For example, non-Higgsable clusters of seven-branes with intricate gauge sectors occur with probability above $1 - 1.01 \times 10^{-755}$, and the geometric gauge group rank is above 160 with probability .999995. In the latter case there are at least 10 E_8 factors, the structure of which fixes the gauge groups on certain nearby seven-branes. Visible sectors may arise from E_6 or $SU(3)$ seven-branes, which occur in certain random samples with probability $\simeq 1/200$.

Title with a point: concrete construction algorithm.



Random draw? Geometry ~ Unif(This Ensemble).

Has ~ 750 gauge sectors, # of ALPs ~ Thousands

PHENO Takeaway: Remnant Degrees of Freedom

35 Years of String Pheno → String Remnants

Remnants: new DOF, apparent accidental consequences from the UV, not motivated by shortcomings of SMs of particle physics or cosmology.

from TASI Lectures: [J.H., Langacker] 2018

Since Then: Implications of Control

Generic vacua arise at large # topological cycles.

The Kreuzer-Skarke Axiverse

Mehmet Demirtas,[a] Cody Long,[b] Liam McAllister,[a] and Mike Stillman[c]

[a]*Department of Physics, Cornell University, Ithaca, NY 14853, USA*
[b]*Department of Physics, Northeastern University, Boston, MA 02115, USA*
[c]*Department of Mathematics, Cornell University, Ithaca, NY 14853, USA*

md775@cornell.edu, co.long@northeastern.edu, mcallister@cornell.edu, mike@math.cornell.edu

Result: Demanding control pushes you out in the "stretched Kahler cone", ***making remnants more important*** (light ALPs, weak gauge couplings).

Illustrative Example: ALP-Photon Couplings

A Model Narrative from String Theory

Goal: well-motivated string ideas for PHENO.

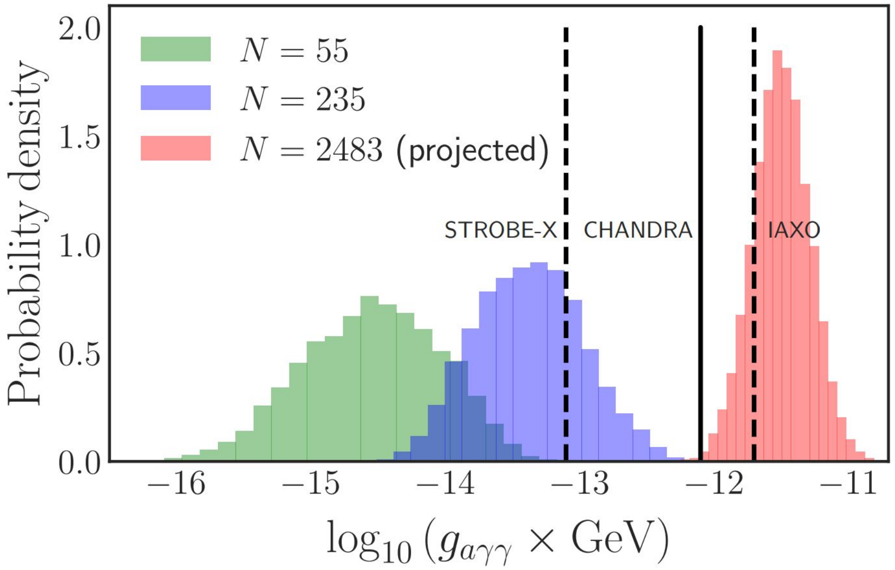
If string theory is true:

1) it has a photon.
2) ALPs are a good bet, with # ALPs = N large.
   if controlled, many ALPs should be light.
3) no symmetry forbids $g_{a\gamma\gamma}$ in EFT,
many ALPs couple to photon!

Q: how goes $E[g_{a\gamma\gamma}]$ scale with N in the landscape?

Large N is where most of it lives.

[J.H., Long, Nelson, Salinas] 2019



$$\mathrm{mean}(g_{a\gamma\gamma}) = 2.73 \times 10^{-18} \times N^{1.77}\ \mathrm{GeV}^{-1}$$

Note well: used a computationally precise but nonetheless toy model for photon.

Part 2) Neural Network Pheno

Deep relationship between NNs and QFT.
Opens up an avenue in theoretical ML.

based on 2008.08601 and work to appear this week,
both with my amazing students, A. Maiti and K. Stoner.

The Linchpin of the Revolution: Neural Networks

Supervised:

NN is powerful function that predicts outputs (e.g. class labels), given input.

Generative Models:

NN is powerful function that maps draws from noise distribution to draws from data distribution.

Reinforcement:

NN is powerful function that, e.g., picks intelligent state-dependent actions.

Natural Language:

NN is powerful function that, e.g., extends a sequence, given a prompt.

Neural Networks = Powerful Functions

A neural network is just a function:

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

with continuous learnable parameters θ
and discrete hyperparameters N.
Training updates θ to improve performance.

Crucial for today: $\theta \sim P_{\theta}$,
parameters are draws from some distribution.

So, fire up your code

and have it create a neural network.

It’s a random function!

Do it again.

It’s another random function.

Again and again and again.

All different, all random functions.

But from what distribution?

We normally think of NNs as having an architecture with random params.

But they're also random functions, and we can study them instead ***in function space.***

What Distribution? Sharpening with the Simplest Example

[Neal], 90s

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.

“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]

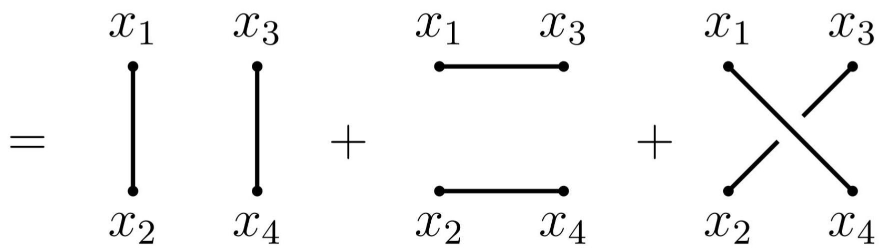
Free Field Theory is a Gaussian Process

$$Z = \int D\phi \, e^{-S[\phi]} \qquad S[\phi] = \int d^d x \, \phi(x)(\Box + m^2)\phi(x)$$

So infinite neural networks are like free field theory!

Statistics entirely determined by one-point function (mean)
and two-point function (GP kernel),
compute correlators in terms of Feynman diagrams.

$$
\begin{aligned}
G^{(4)}_{\mathrm{GP}}(x_1, x_2, x_3, x_4) \quad &= \quad K(x_1, x_2)K(x_3, x_4) \\
&+ K(x_1, x_3)K(x_2, x_4) + K(x_1, x_4)K(x_2, x_3)
\end{aligned}
$$



What about finite-N networks?

The function space distribution is generally non-Gaussian.
But non-Gaussianities $\to 0$ as $N \to \infty$.

Large-but-finite N?
Weakly coupled interactions from the small non-Gaussianities.

a bit different, but see also:
[Dyer, Gur-Ari], [Yaida]

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

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

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

where, e.g., could have a **model**:

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

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

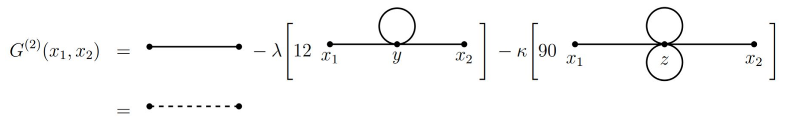
$$S = S_{\mathrm{GP}} + \int d^{d_{\mathrm{in}}}x \; \left[\lambda\, f(x)^4 + \kappa\, f(x)^6 \right]$$

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

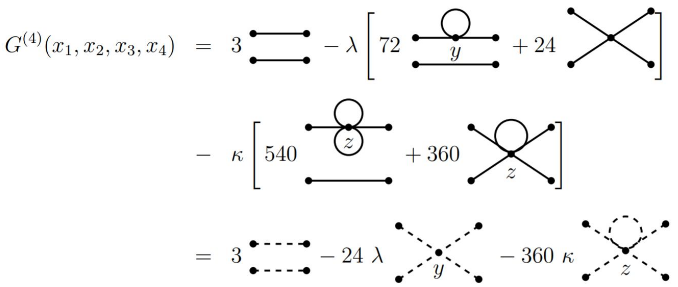
κ is 1/N suppressed rel. λ, somes more irrelevant (Wilsonian sense), gives *even simpler NGP distribution.*

Once Again, Feynman Diagrams for NNs

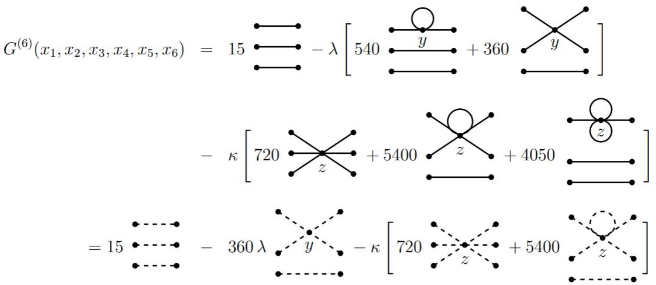
point: theory equations that actually enter our NN codes.



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



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



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

Neural Network Pheno

Modeling interactions with ΔS, constraining with experiments, making new predictions, and verifying?

This is neural network phenomenology!

Hope to do it for state-of-the-art networks (transformers)?

Could lead to real ML breakthroughs.

(e.g., the one-point function of the trained network distribution is the central object in supervised learning.)

One Crisp Result

To prove you can actually do things with this.

(See also: backup slides).

Symmetries of NN distribution?

[Maiti, Stoner, J.H.] to appear this week.

Clearly any NN pheno would benefit from knowledge of the expected symmetries of the dist.

We could use experiments to determine them.

But in fact **we can use duality:** parameter space and function space give two dual perspectives on NN!

Specifically: correlation functions are fundamental in the NN system, but can be computed in parameter or function space duality frames.

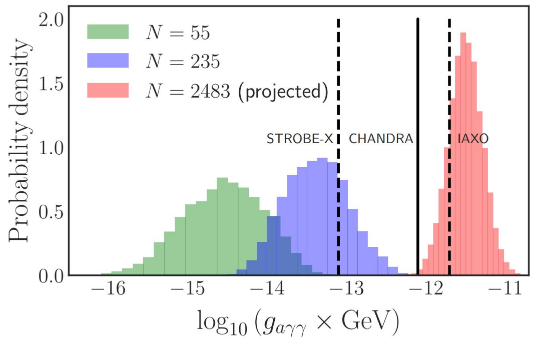
Can deduce symmetry properties of function space description from their symmetries.

$$G^{(n)}(x_1, \ldots, x_n) = \mathbb{E}[f(x_1)\ldots f(x_n)]$$
$$= \frac{1}{Z_\theta}\int d\theta\; f(x_1)\ldots f(x_n)\; P_\theta$$
$$= \frac{1}{Z_f}\int Df\; f(x_1)\ldots f(x_n)\; P_f$$

Example below: SO(D) output symmetry of NN distribution, assuming linear output layer with invariant weights and biases (e.g. mean 0 Gaussian).

$$\begin{aligned}
&G'^{(n)}_{i_1\ldots i_n}(x'_1,\ldots,x'_n) = \mathbb{E}[R_{i_1j_1}f_{j_1}(x_1)\ldots R_{i_nj_1}f_{j_n}(x_n)] \\
&= \frac{1}{Z_\theta}\int DW\,Db\,D\theta_g\; R_{i_1j_1}(W_{j_1k_1}g_{k_1}(x_1)+b_{j_1})\ldots R_{i_nj_n}(W_{j_nk_n}g_{k_n}(x_n)+b_{j_n})P_W P_b P_{\theta_g} \\
&= \frac{1}{Z_\theta}\int |R^{-1}|^2 D\tilde{W}\,D\tilde{b}\,D\theta_g\,(\tilde{W}_{i_1k_1}g_{k_1}(x_1)+\tilde{b}_{i_1})\ldots(\tilde{W}_{i_nk_n}g_{k_n}(x_n)+\tilde{b}_{i_n})P_{R^{-1}\tilde{W}}P_{R^{-1}\tilde{b}}P_{\theta_g} \\
&= \mathbb{E}[f_{i_1}(x_1)\ldots f_{i_n}(x_n)] = G^{(n)}(x_1,\ldots,x_n), \qquad (8)
\end{aligned}$$

Once Again: The Two Main Points



Part 1) String Pheno in 2021

Takeaway: draw vacua V ~ U(known string constructions), many ALPs, many gauge sectors, light and weakly coupled when controlled.

Potentially detectable remnant DOF are everywhere $\longrightarrow$ problems and opportunities. Complements shifting paradigms (?) in pheno.

Part 2) Neural Network Pheno

Takeaway: neural networks are random functions from nearly-Gaussian dists. *This is like in particle phenomenology!* Model non-Gaussianities via EFT.

Use duality to determine symmetries of neural net effective actions.

General Conclusion:

What's happening in computer science is special.
Relative to math and physics, it's in its infancy,
and it will likely be woven into the math / physics story.

Our most cherished physics problems are often unwieldy,
but we have new opportunities to use deep learning for progress.

For me: that's the string landscape and mathematics,
but the toolbox is general and it's a great time to dive in.

Thanks!

Questions?

Or get in touch after:

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

A Flash of Some NN-QFT Experimental Results

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

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

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

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

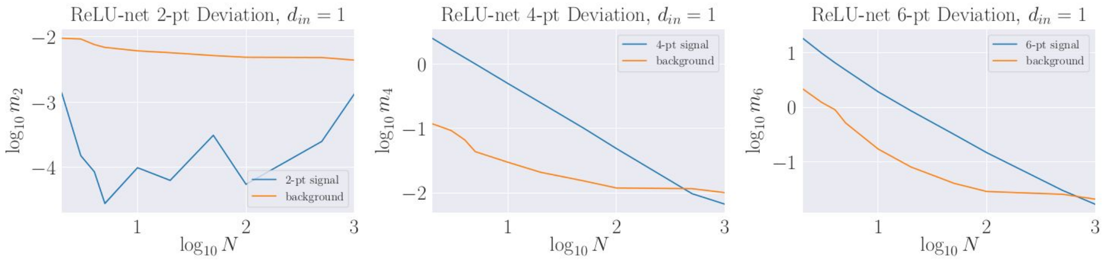
A Flash of Some NN-QFT Experimental Results

Experimental description

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

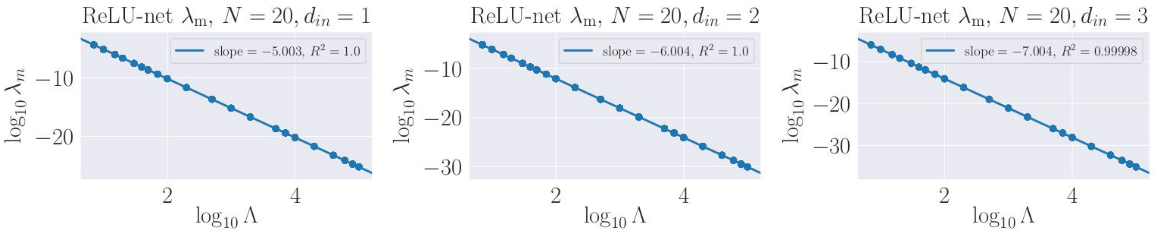
Drew millions of models and evaluated on fixed sets of input to do experiments with correlators and the EFT description of NN distribution.

NGP correlators become GP correlators as $N \to \infty$



Note also: $G6_{con} \sim 1/N^2$

Dependence of Quartic Coupling on Cutoff



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

Depends on input dimension, as expected from QFT.

Verification of EFT Predictions

	$(\lambda_0, \lambda_2, \lambda_{\text{NL}})$	Test (MAPE, MSE)
Gauss M_0	$(0.0, 0.0, 0.0)$	$(100, 0.019)$
Gauss M_1	$(0.0046, 0.0, 0.0)$	$(0.0145, 6.8 \times 10^{-10})$
Gauss M_2	$(0.0043, 0.0011, 0.0)$	$(0.0144, 6.7 \times 10^{-10})$
Gauss M_3	$(0.00062, 0.00016, 0.0015)$	$(0.0156, 7.5 \times 10^{-10})$
ReLU M_0	$(0.0, 0.0, 0.0)$	$(100, 0.003)$
ReLU M_1	$(6.2 \times 10^{-11}, 0.0, 0.0)$	$(0.0035, 7.6 \times 10^{-12})$
ReLU M_2	$(1.2 \times 10^{-18}, 8.7 \times 10^{-15}, 0.0)$	$(0.0013, 1.5 \times 10^{-12})$
ReLU M_3	$(1.2 \times 10^{-18}, 8.7 \times 10^{-15}, 6.8 \times 10^{-17})$	$(0.0012, 1.2 \times 10^{-12})$
Erf M_0	$(0.0, 0.0, 0.0)$	$(100, 0.006)$
Erf M_1	$(0.039, 0.0, 0.0)$	$(0.030, 8.3 \times 10^{-10})$
Erf M_2	$(0.040, -0.00043, 0.0)$	$(0.0042, 1.9 \times 10^{-11})$
Erf M_3	$(0.0019, -0.0054, 0.0063)$	$(0.037, 1.1 \times 10^{-9})$

Test / train split on connected 4-pt function to verify predictions of measured couplings.