Machine Learning in the String Landscape

Landscapia @ CEA Saclay

Jim Halverson

LVX VERITAS VIRTVS Northeastern University

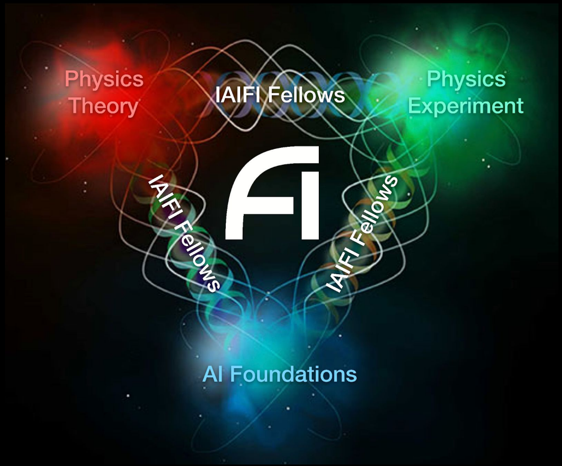
NSF

AI

Connecting
in this area.

NSF AI Institute for Artificial Intelligence and Fundamental Interactions (IAIFI /aɪ-faɪ/)

Advance physics knowledge—from the smallest building blocks of nature to the largest structures in the universe—and galvanize AI research innovation



iaifi@mit.edu

https://iaifi.org

@iaifi-news

Jobs @ Northeastern

Northeastern University, Department of Physics (CoS)

Position ID: Northeastern-Department of Physics (CoS)-PRA_IAIFI [#25427]
Position Title: Northeastern IAIFI Postdoctoral Research Associate
Position Type: Postdoctoral
Position Location: Boston, Massachusetts 02115, United States [map]
Subject Areas: Physics / hep-th, HEP-Theory (hep-th), High Energy Physics, high energy physics or mathematical physics, High Energy Theory, Particle Theory, Quantum Field Theory, quantum gravity, String Theory, Theoretical Particle Physics, Theoretical Physics
Appl Deadline: 2023/12/15 11:59PM (posted 2023/08/23, listed until 2024/02/23)
Position Description: Apply

https://academicjobsonline.org/ajo/jobs/25427

IAIFI (ML + strings) postdoctoral research associate, with me and Fabian Ruehle.

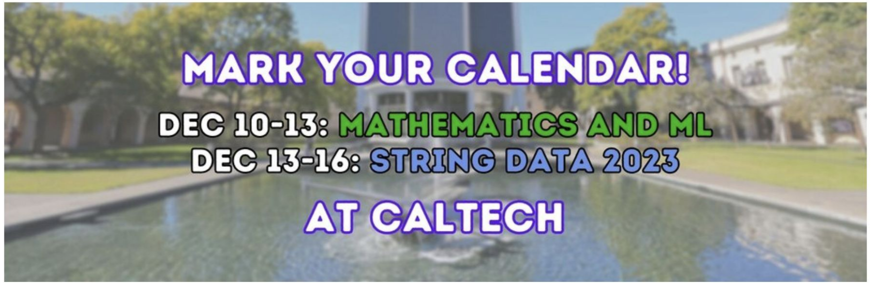
Northeastern University, Department of Physics (CoS)

Position ID: Northeastern-Department of Physics (CoS)-PRA_NEU [#25670]
Position Title: Northeastern Postdoctoral Research Associate
Position Type: Postdoctoral
Position Location: Boston, Massachusetts 02115, United States [map]
Subject Areas: Physics / Elementary Particle Physics, Elementary Particle Theory, Field Theory, hep-th, HEP-Theory (hep-th), High Energy Theory, Machine Learning, Mathematical Physics, Particle Theory, Physics - Mathematical Physics, Quantum Field Theory, String Theory, String Theory/Quantum Gravity/Field Theory, string-math, Theoretical high energy physics
Appl Deadline: 2023/12/15 11:59PM (posted 2023/09/12, listed until 2024/03/12)
Position Description: Apply

https://academicjobsonline.org/ajo/jobs/25670

2 jobs, postdoctoral research associates, with Fabian Ruehle and Sarah Harrison.

Upcoming Meetings



Mathematics and Machine Learning 2023
Dec 10-12, 2023, @ Caltech
https://mathml2023.caltech.edu/

String Data 2023
Dec 13-15, 2023, @ Caltech
https://stringdata2023.caltech.edu/

Jan 14 – 19, 2024
Aspen Center for Physics

Strings, Fields, and Deep Learning
January 14-19, 2023, @ Aspen Center for Physics
https://indico.cern.ch/event/1299185/

come join us! send me an e-mail.

String Theory, QG are Hard.

Ask an Oracle

1. Complete non-pert. definition of M-theory?
2. All the vacua? Decay rates, channels?
3. What is the cosmo measure?
4. How to make statistical predictions tractable?

Need Formal Theory.

Need Smart Computation.

Caveats and Comments from the Modern Era

- a number string pheno folks working on ML have interesting non-string-pheno off-shoots. will cite some refs!

- using LLMs for research, ML or otherwise, is cool! I won't talk about it, but feel free to ask me questions anyways.

- to those working in this area:
  my sincere apologies if I didn't adequately cite / cover / present your work. Please ***briefly*** raise your exciting results in Q&A.

- stats language: "sampling", "drawn from", etc.

  sounds gross, but if you've ever done a compuation in an actual string compactification, you yourself have sampled string data.

- numerical techniques can be rigorous.

  "rigor" is loaded. I don't mean statistical convergence, I mean zero-error results, applied ML or ***from ML theory***.

  see upcoming article with Gukov, Ruehle.

Outline:

ML Preliminaries

1) Neural Networks
2) Universal Approximation
3) Architectures
4) Famous Examples and Verbs

Cursory Glances

1) Prediction: Supervised Learning
2) Rigor: Conjecture Gen. + Theorems
3) Structure: Persistent Homology
4) Search: Reinforcement Learning, Gen. Algs, Quantum Annealers

In-Depth Looks

1) Auto-diff everything: Flux Vacua
2) Flow: CY-metrics and NN theory

Dreams

1) Generative Models: Simulating String Theory and Statistical Inference

Preliminaries

1. Neural Networks

2. Universal Approximation and Network Architectures

3. Famous Examples and their Verbs

Neural Networks

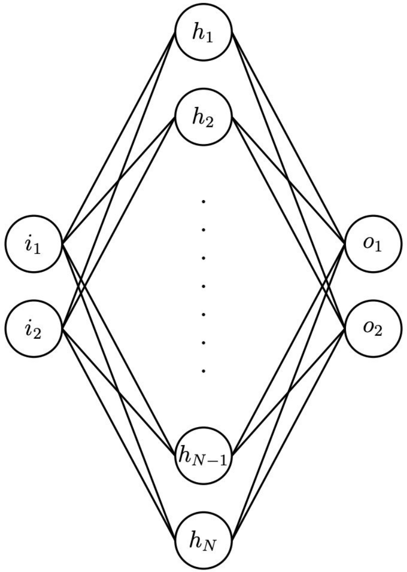
A **neural network** is a parameterized function, typically a "big" function composed out of many other simpler functions, with *many* parameters.

Therefore, map from param space to function space.

Parameters θ drawn θ ~ P(θ) at init.
Puts stats on NN function space,
defines a field theory, compute correlators, etc.

for FT, 2307 of [Demirtas, J.H., Maiti, Schwartz, Stoner] + refs therein

State of the art NNs have 500 billion parameters, cost millions of dollars to train.



e.g. the perceptron, building block for deep FFNN.

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

$b_0, b_1 \sim \mathcal{N}(\mu_b, \sigma_b^2)$ $\quad W_0 \sim \mathcal{N}(\mu_W, \sigma_W^2/d_{\text{in}})$ $\quad W_1 \sim \mathcal{N}(\mu_W, \sigma_W^2/N)$

Universal Approximation and Architectures

Universal Approximation Theorems:

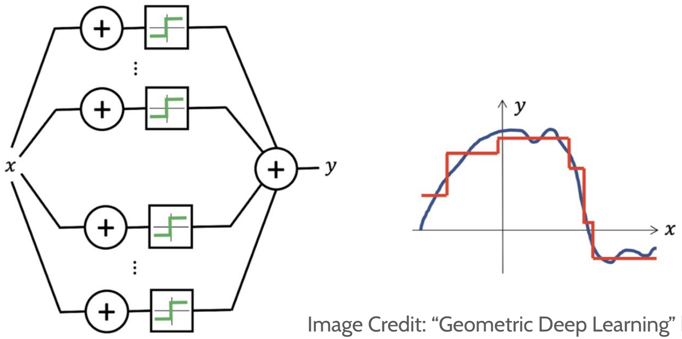


Image Credit: “Geometric Deep Learning” book

Figure 1: Multilayer Perceptrons (Rosenblatt, 1958), the simplest feed-forward neural networks, are universal approximators: with just one hidden layer, they can represent combinations of step functions, allowing to approximate any continuous function with arbitrary precision.

essence: approximation any function, error O(1/N), N = width.

Many versions, see Wiki.

Architecture:

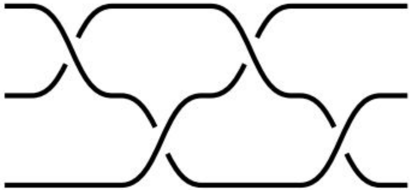
How you stitch simple functions into complex ones.
Architecture choice depends on data structure.

e.g. from before, FFNN. but there are others:

Convolutional Net

image data, identifies local features with conv kernels.

Transformer



language data, attention mechanism, “T” in GPT.

knots app [Gukov, J.H, Ruehle, Sulkowski]

Famous Examples, Quantified with Verbs

Predict: Supervised Learning, e.g. MNIST classification.

Generate / Simulate: Generative Models, e.g. Text2Img, GPT-4.

Search / Play: Reinforcement Learning, e.g. AlphaZero.

ChatGPT

In Paris, threads weave,

String theory's vast landscape,

Eiffel echoes deep.

e.g. Text2Text. GPT-4.

Prompt was iteration on strings and Paris.

Cursory Glances

Prediction: Supervised Learning

- **Data:** a set of input-output pairs, $(x,y) \in X \times Y$
- **Parameterized Model:** $f_\theta: X \to Y$
  e.g., lin. reg, neural net.
- **Loss Function:** $L: Y \times Y \to R$
  measures quality of preds of $f_\theta(x)$ rel. true labels y_i.
  e.g. $L_{MSE} = \boldsymbol{\Sigma} (y_i - f_\theta(x_i))^2$
- **Optimize:** change θ to minimize L on data.
  e.g. gradient descent.

Deep-Learning the Landscape

Yang-Hui He

[1] Department of Mathematics, City, University of London, EC1V 0HB, UK
[2] Merton College, University of Oxford, OX14JD, UK
[3] School of Physics, NanKai University, Tianjin, 300071, P.R. China

hey@maths.ox.ac.uk

Abstract

We propose a paradigm to deep-learn the ever-expanding databases which have emerged in mathematical physics and particle phenomenology, as diverse as the statistics of string vacua or combinatorial and algebraic geometry. As concrete examples, we establish multi-layer neural networks as both classifiers and predictors and train them with a host of available data ranging from Calabi-Yau manifolds and vector bundles, to quiver representations for gauge theories. We find that even a relatively simple neural network can learn many significant quantities to astounding accuracy in a matter of minutes and can also predict hithertofore unencountered results. This paradigm should prove a valuable tool in various investigations in landscapes in physics as well as pure mathematics.

Evolving neural networks with genetic algorithms to study the String Landscape

FABIAN RUEHLE [1]

Rudolf Peierls Centre for Theoretical Physics, Oxford University, 1 Keble Road, Oxford, OX1 3NP, UK

Abstract

We study possible applications of artificial neural networks to examine the string landscape. Since the field of application is rather versatile, we propose to dynamically evolve these networks via genetic algorithms. This means that we start from basic building blocks and combine them such that the neural network performs best for the application we are interested in. We study three areas in which neural networks can be applied: to classify models according to a fixed set of (physically) appealing features, to find a concrete realization for a computation for which the precise algorithm is known in principle but very tedious to actually implement, and to predict or approximate the outcome of some involved mathematical computation which performs too inefficient to apply it, e.g. in model scans within the string landscape. We present simple examples that arise in string phenomenology for all three types of problems and discuss how they can be addressed by evolving neural networks from genetic algorithms.

Machine Learning of Calabi-Yau Volumes

Daniel Krefl[a] and Rak-Kyeong Seong[b]
[a] *Theoretical Physics Department, CERN, Geneva 23, CH-1211 Switzerland*
[b] *Department of Physics and Astronomy, Uppsala University, SE-751 08 Uppsala, Sweden*

We employ machine learning techniques to investigate the volume minimum of Sasaki-Einstein base manifolds of non-compact toric Calabi-Yau 3-folds. We find that the minimum volume can be approximated via a second order multiple linear regression on standard topological quantities obtained from the corresponding toric diagram. The approximation improves further after invoking a convolutional neural network with the full toric diagram of the Calabi-Yau 3-folds as the input. We are thereby able to circumvent any minimization procedure that was previously necessary and find an explicit mapping between the minimum volume and the topological quantities of the toric diagram. Under the AdS/CFT correspondence, the minimum volumes of Sasaki-Einstein manifolds correspond to central charges of a class of $4d\ \mathcal{N}=1$ superconformal field theories. We therefore find empirical evidence for a function that gives values of central charges without the usual extremization procedure.

Machine Learning in the String Landscape

Jonathan Carifio,[a] James Halverson,[a] Dmitri Krioukov,[a,b,c] and Brent D. Nelson[a]
[a] *Department of Physics, Northeastern University, Boston, MA 02115, USA*
[b] *Department of Mathematics, Northeastern University, Boston, MA 02115, USA*
[c] *Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115, USA*

ABSTRACT:

We utilize machine learning to study the string landscape. Deep data dives and conjecture generation are proposed as useful frameworks for utilizing machine learning in the landscape, and examples of each are presented. A decision tree accurately predicts the number of weak Fano toric threefolds arising from reflexive polytopes, each of which determines a smooth F-theory compactification, and linear regression generates a previously proven conjecture for the gauge group rank in an ensemble of $\frac{4}{3} \times 2.96 \times 10^{755}$ F-theory compactifications. Logistic regression generates a new conjecture for when E_6 arises in the large ensemble of F-theory compactifications, which is then rigorously proven. This result may be relevant for the appearance of visible sectors in the ensemble. Through conjecture generation, machine learning is useful not only for numerics, but also for rigorous results.

Early String / ML papers were usually supervised

Rigor: Conjecture Generation, My Path Into ML

[Submitted on 3 Jul 2017]

Machine Learning in the String Landscape

Jonathan Carifio, James Halverson, Dmitri Krioukov, Brent D. Nelson

We utilize machine learning to study the string landscape. Deep data dives and conjecture generation are proposed as useful frameworks for utilizing machine learning in the landscape, and examples of each are presented. A decision tree accurately predicts the number of weak Fano toric threefolds arising from reflexive polytopes, each of which determines a smooth F-theory compactification, and linear regression generates a previously proven conjecture for the gauge group rank in an ensemble of $\frac{4}{3} \times 2.96 \times 10^{755}$ F-theory compactifications. Logistic regression generates a new conjecture for when E_6 arises in the large ensemble of F-theory compactifications, which is then rigorously proven. This result may be relevant for the appearance of visible sectors in the ensemble. Through conjecture generation, machine learning is useful not only for numerics, but also for rigorous results.

Theorem: related to prevalence of E_6 grand unification in ensemble of 10^{755} string geometries.

Big Data: there, 10^{755} < # geoms < ∞.

[J.H., Long, Sung] [Di Cerbo, Svaldi]

The Idea:

1) use ML, e.g. supervised, on interesting problem.

2) "open the box" to discover key decision variables.
e.g. decision tree, automatic
e.g. gradient saliency in NNs (see DeepMind results)

3) bring human expert into the ML loop.
think hard. conjecture. maybe iterate.

4) prove theorem.

More Conjecture Generation and Theorems in Strings

Machine Learning Line Bundle Cohomology

Callum R. Brodie[1], Andrei Constantin[2], Rehan Deen[1], Andre Lukas[1]

arXiv: 1906.08730

Machine Learning and Algebraic Approaches towards Complete Matter Spectra in 4d F-theory

Martin Bies[1], Mirjam Cvetič[2,3,4], Ron Donagi[3,2],
Ling Lin[5], Muyang Liu[2], Fabian Ruehle[5,6]

arXiv: 2007.00009

We investigate different approaches to machine learning of line bundle cohomology on complex surfaces as well as on Calabi-Yau three-folds. Standard function learning based on simple fully connected networks with logistic sigmoids is reviewed and its main features and shortcomings are discussed. It has been observed recently that line bundle cohomology can be described by dividing the Picard lattice into certain regions in each of which the cohomology dimension is described by a polynomial formula. Based on this structure, we set up a network capable of identifying the regions and their associated polynomials, thereby effectively generating a conjecture for the correct cohomology formula. For complex surfaces, we also set up a network which learns certain rigid divisors which appear in a recently discovered master formula for cohomology dimensions.

piecewise polynomials for LB Hodge numbers

Motivated by engineering vector-like (Higgs) pairs in the spectrum of 4d F-theory compactifications, we combine machine learning and algebraic geometry techniques to analyze line bundle cohomologies on families of holomorphic curves. To quantify jumps of these cohomologies, we first generate 1.8 million pairs of line bundles and curves embedded in dP3, for which we compute the cohomologies. A white-box machine learning approach trained on this data provides intuition for jumps due to curve splittings, which we use to construct additional vector-like Higgs-pairs in an F-Theory toy model. We also find that, in order to explain quantitatively the full dataset, further tools from algebraic geometry, in particular Brill--Noether theory, are required. Using these ingredients, we introduce a diagrammatic way to express cohomology jumps across the parameter space of each family of matter curves, which reflects a stratification of the F-theory complex structure moduli space in terms of the vector-like spectrum. Furthermore, these insights provide an algorithmically efficient way to estimate the possible cohomology dimensions across the entire parameter space.

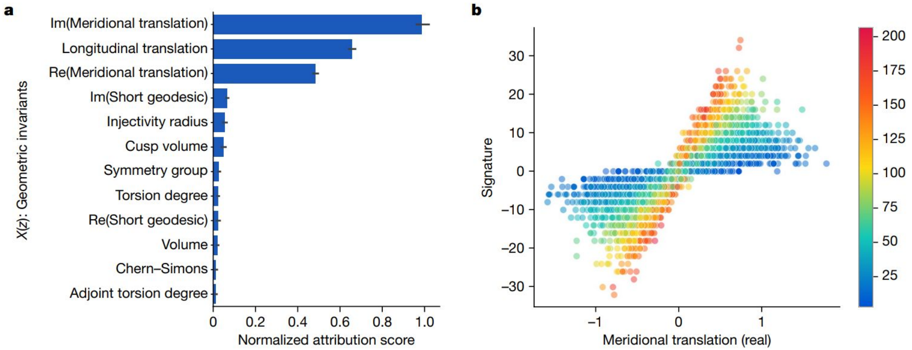
ML-inspired intuition for vector pairs.

Conjecture Generation in Knot Theory: DeepMind Results

z: **Knot**	*X*(*z*): **Geometric invariants** Volume	Chern–Simons	Meridional translation	...	*Y*(*z*): **Algebraic invariants** Signature	Jones polynomial	...
	2.0299	0	i	...	0	$t^{-2} - t^{-1} + 1 - t + t^2$	...
	2.8281	–0.1532	$0.7381 + 0.8831i$	...	–2	$t - t^2 + 2t^3 - t^4 + t^5 - t^6$	...
	3.1640	0.1560	$-0.7237 + 1.0160i$	...	0	$t^{-2} - t^{-1} + 2 - 2t + t^2 - t^3 + t^4$	...

- predict knot sig. from geom invariants.
- interpreted with “attribution” via gradient saliency.

$$\text{importance of } x_i \propto \frac{\partial f(x)}{\partial x_i}$$



- some features clearly correlated with sig.
- given knowledge of crucial features, made conjecture and proved that:

Theorem: There exists a constant c such that, for any hyperbolic knot K,

$$|2\sigma(K) - \mathrm{slope}(K)| \leq c\,\mathrm{vol}(K)\,\mathrm{inj}(K)^{-3} \tag{2}$$

[Davies et al] Nature 2021. Also have representation theory results!

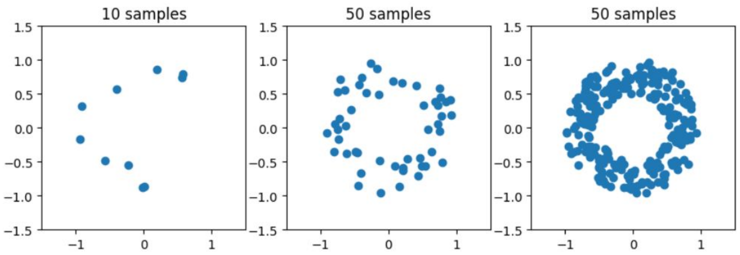
Structure: Persistent Homology

- **Data manifold.** M in ambient space A.
  e.g. Calabi-Yau manifolds
  e.g. stabilized vacua in moduli space.

- **Structure.** M has geometry, topology, e tc.

- **Samples.** Xs ~ P(M) yield point cloud in A.

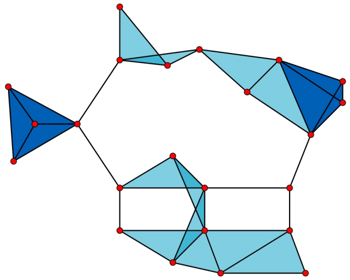
- **Question:** topology of M from samples?



- **Persistent Homology.**
  A one-parameter family of simplicial complexes yields homology family $H_k(Xs; \delta)$.

- **Vietoris-Rips Complex.** To each k-point subset of Xs within δ-ball, a (k-1)-simplex.

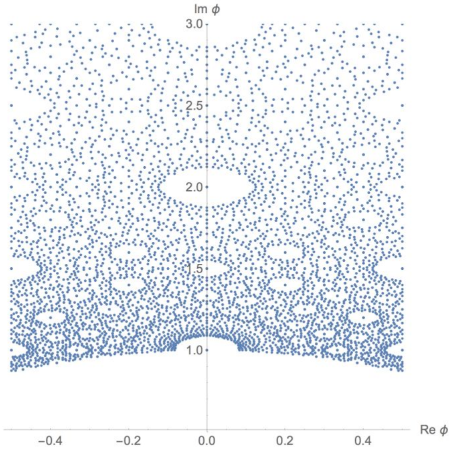
  e.g.

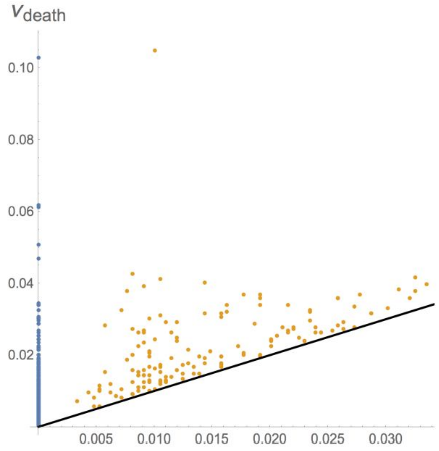


- **Life and death.** Cycles are born and die.
  Real cycles persist, have $\delta_{death} >> \delta_{birth}$.

Structure: Persistent Homology

also great [Shiu et. al.] on TDA for CMB data!
see also: [Cirafici] for early TDA and strings





Axiodilaton values, weak IIB on rigid CY, with flux cutoff.

Persistent homology detects topology,
birth and death of cycles.

Topological Data Analysis for the String Landscape

Alex Cole and Gary Shiu

acole4@wisc.edu, shiu@physics.wisc.edu

Department of Physics, University of Wisconsin, Madison, WI 53706, USA

Abstract

Persistent homology computes the multiscale topology of a data set by using a sequence of discrete complexes. In this paper, we propose that persistent homology may be a useful tool for studying the structure of the landscape of string vacua. As a scaled-down version of the program, we use persistent homology to characterize distributions of Type IIB flux vacua on moduli space for three examples: the rigid Calabi-Yau, a hypersurface in weighted projective space, and the symmetric six-torus $T^6 = (T^2)^3$. These examples suggest that persistence pairing and multiparameter persistence contain useful information for characterization of the landscape in addition to the usual information contained in standard persistent homology. We also study how restricting to special vacua with phenomenologically interesting low-energy properties affects the topology of a distribution.

Search: Reinforcement Learning

- **key idea:** “winning the game” is an exact solution to problem. Zero error.

- an *agent* interacts in an *environment*.
- it perceives a *state* from state space.
- its *policy* picks an action, given state.

- arrives in new state, receives *reward*.
- successive rewards accum. to *return*.
  future rewards penalized by *discount*.

- *state-* and *action-value* functions.

$$\pi : \mathcal{S} \to \mathcal{A}$$

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

$$v(s) = \mathbb{E}[G_t | S_t = s]$$

$$q(s, a) = \mathbb{E}[G_t | S_t = s, A_t = a]$$

Search: Reinforcement Learning

see also, [Gukov, J.H., Manolescu, Ruehle] studied knots, ruled out > 800 potential counterexamples to smooth 4d Poincare conjecture.

Branes with Brains: Exploring String Vacua with Deep Reinforcement Learning

James Halverson,[a] Brent Nelson,[a] Fabian Ruehle[b,c]

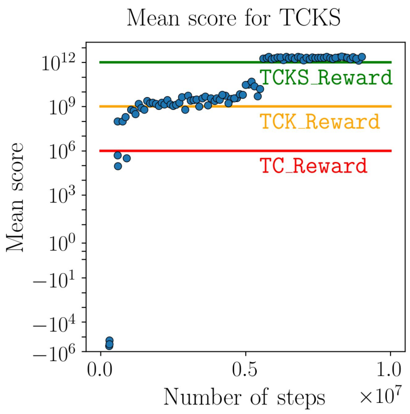
[a] *Department of Physics, Northeastern University, Boston, MA 02115, USA*

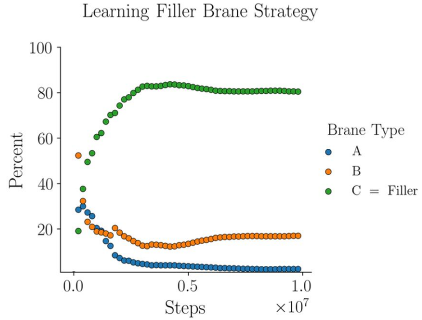
[b] *CERN, CERN, Theoretical Physics Department 1 Esplanade des Particules, Geneva 23, CH-1211, Switzerland*

[c] *Rudolf Peierls Centre for Theoretical Physics, Oxford University, 1 Keble Road, Oxford, OX1 3NP, UK*

E-mail: j.halverson@northeastern.edu, b.nelson@northeastern.edu, fabian.ruehle@cern.ch

Abstract: We propose deep reinforcement learning as a model-free method for exploring the landscape of string vacua. As a concrete application, we utilize an artificial intelligence agent known as an asynchronous advantage actor-critic to explore type IIA compactifications with intersecting D6-branes. As different string background configurations are explored by changing D6-brane configurations, the agent receives rewards and punishments related to string consistency conditions and proximity to Standard Model vacua. These are in turn utilized to update the agent's policy and value neural networks to improve its behavior. By reinforcement learning, the agent's performance in both tasks is significantly improved, and for some tasks it finds a factor of $\mathcal{O}(200)$ more solutions than a random walker. In one case, we demonstrate that the agent learns a human-derived strategy for finding consistent string models. In another case, where no human-derived strategy exists, the agent learns a genuinely new strategy that achieves the same goal twice as efficiently per unit time. Our results demonstrate that the agent learns to solve various string theory consistency conditions simultaneously, which are phrased in terms of non-linear, coupled Diophantine equations.





- IIA, intersecting D6.
- learning realizes punctuated equilibria in search of tadpole, K-theory, SUSY reward.
- learned filled brane strategy, but also a better one.

Search: Genetic Algorithms and Quantum Annealers

Genetic Algorithms:
Simulate natural selection by evolve solutions using mutation, crossover, selection.
Survival of the fittest.

Quantum Annealers: a.k.a. use DWave

exploit quantum superposition and tunneling for specific optimization problems.

different from general quantum computers.

Right: early literature in SUSY (2004) and strings (2014).

Genetic Algorithms and Experimental Discrimination of SUSY Models

B.C. Allanach
LAPTH, 9 chemin Bellevue, BP110, Annecy 74941, France
E-mail: benjamin.allanach@cern.ch

D. Grellscheid
Physikalisches Institut der Universität Bonn, Nussallee 12, 53115 Bonn, Germany
E-mail: grelli@th.physik.uni-bonn.de

F. Quevedo
DAMTP, Centre for Mathematical Sciences University of Cambridge, Cambridge, CB3 0WA, United Kingdom
E-mail: f.quevedo@damtp.cam.ac.uk

Genetic Algorithms and the Search for Viable String Vacua

Steven Abel[◇] **and John Rizos**[†]

[◇] *IPPP, Durham University, South Road, Durham, DH1 3LE, UK*
[†] *Physics Department, University of Ioannina, GR45110, Greece*

Abstract: Genetic Algorithms are introduced as a search method for finding string vacua with viable phenomenological properties. It is shown, by testing them against a class of Free Fermionic models, that they are orders of magnitude more efficient than a randomised search. As an example, three generation, exophobic, Pati-Salam models with a top Yukawa occur once in every 10^{10} models, and yet a Genetic Algorithm can find them after constructing only 10^5 examples. Such non-deterministic search methods may be the only means to search for Standard Model string vacua with detailed phenomenological requirements.

Search: Genetic Algorithms and Quantum Annealers

Manifold	h	$\lvert\Gamma\rvert$	Range	GA	Scan	Found	Explored
7862	4	2	[-7,8]	5	5	100%	10^{-10}
7862	4	4	[-7,8]	30	31	97%	10^{-10}
7447	5	2	[-7,8]	38	38	100%	10^{-14}
7447	5	4	[-7,8]	139	154	90%	10^{-14}
5302	6	2	[-7,8]	403	442	93%	10^{-19}
5302	6	4	[-7,8]	722	897	80%	10^{-19}
4071	7	2	[-3,4]	11,937	N/A	N/A	10^{-14}

Decoding Nature with Nature's Tools:
Heterotic Line Bundle Models of Particle Physics with Genetic Algorithms and Quantum Annealing

Steve A. Abel[1,2,*] Andrei Constantin[3,†] Thomas R. Harvey[3,‡] Andre Lukas[3,§] and Luca A. Nutricati[1,¶]

[1] *IPPP, Durham University, Durham, DH1 3LE UK*
Department of Mathematical Sciences, Durham University, Durham DH1 3LE, UK
[2] *CERN, Theoretical Physics Department, CH 1211 Geneva 23 Switzerland*
[3] *Rudolf Peierls Centre for Theoretical Physics, University of Oxford, Parks Road, Oxford OX1 3PU, UK*

The string theory landscape may include a multitude of ultraviolet embeddings of the Standard Model, but identifying these has proven difficult due to the enormous number of available string compactifications. Genetic Algorithms (GAs) represent a powerful class of discrete optimisation techniques that can efficiently deal with the immensity of the string landscape, especially when enhanced with input from quantum annealers. In this letter we focus on geometric compactifications of the $E_8 \times E_8$ heterotic string theory compactified on smooth Calabi-Yau threefolds with Abelian bundles. We make use of analytic formulae for bundle-valued cohomology to impose the entire range of spectrum requirements, something that has not been possible so far. For manifolds with a relatively low number of Kähler parameters we compare the GA search results with results from previous systematic scans, showing that GAs can find nearly all the viable solutions while visiting only a tiny fraction of the solution space. Moreover, we carry out GA searches on manifolds with a larger numbers of Kähler parameters where systematic searches are not feasible.

Heterotic E_8 x E_8 + physics constraints,
(anomaly cancellation, spectrum, poly-stability, etc).

realize most found via systematic scans,
searched a fraction of the space.

GA vs RL, e.g.:

Evolving Heterotic Gauge Backgrounds:
Genetic Algorithms versus Reinforcement Learning

Steven Abel[a,1], Andrei Constantin[b,2], Thomas R. Harvey[b,3], Andre Lukas[b,]

[a] *IPPP, Durham University, Durham DH1 3LE, UK*

[b] *Rudolf Peierls Centre for Theoretical Physics, University of Oxford Parks Road, Oxford OX1 3PU, UK*

Probing the Structure of String Theory Vacua with Genetic Algorithms and Reinforcement Learning

Alex Cole
University of Amsterdam
a.e.cole@uva.nl

Sven Krippendorf
Arnold Sommerfeld Center for Theoretical Physics
LMU Munich
sven.krippendorf@physik.uni-muenchen.de

Andreas Schachner
Centre for Mathematical Sciences
University of Cambridge
as2673@cam.ac.uk

Gary Shiu
University of Wisconsin-Madison
shiu@physics.wisc.edu

GA + Tadpoles: Our Gracious Hosts!

Algorithmically solving the Tadpole Problem

Iosif Bena[a], Johan Blåbäck[b], Mariana Graña[a] and Severin Lüst[c,d]

[a] *Institut de Physique Théorique, Université Paris Saclay, CEA, CNRS*
Orme des Merisiers
91191 Gif-sur-Yvette Cedex, France

[b] *Dipartimento di Fisica, Università di Roma "Tor Vergata" & INFN - Sezione di Roma2*
Via della Ricerca Scientifica 1, 00133 Roma, Italy

[c] *Jefferson Physical Laboratory, Harvard University, Cambridge, MA 02138, USA*

[d] *Centre de Physique Théorique, Ecole Polytechnique, CNRS*
91128 Palaiseau Cedex, France

Abstract

The extensive computer-aided search applied in [1] to find the minimal charge sourced by the fluxes that stabilize all the (flux-stabilizable) moduli of a smooth K3×K3 compactification uses differential evolutionary algorithms supplemented by local searches. We present these algorithms in detail and show that they can also solve our minimization problem for other lattices. Our results support the Tadpole Conjecture: The minimal charge grows linearly with the dimension of the lattice and, for K3×K3, this charge is larger than allowed by tadpole cancelation.

Even if we are faced with an NP-hard lattice-reduction problem at every step in the minimization process, we find that differential evolution is a good technique for identifying the regions of the landscape where the fluxes with the lowest tadpole can be found. We then design a "Spider Algorithm," which is very efficient at exploring these regions and producing large numbers of minimal-tadpole configurations.

evidence that average-case is better than worst-case complexity here?

String Field Theory and ML

Characterizing 4-string contact interaction using machine learning

Harold Erbin[1,2,3] and Atakan Hilmi Fırat[1,2]

[1] *Center for Theoretical Physics*
Massachusetts Institute of Technology
Cambridge MA 02139, USA

[2] *NSF AI Institute for Artificial Intelligence and Fundamental Interactions*

[3] *Université Paris Saclay, CEA, LIST*
Gif-sur-Yvette, F-91191, France

erbin@mit.edu, firat@mit.edu

Abstract

The geometry of 4-string contact interaction of closed string field theory is characterized using machine learning. We obtain Strebel quadratic differentials on 4-punctured spheres as a neural network by performing unsupervised learning with a custom-built loss function. This allows us to solve for local coordinates and compute their associated mapping radii numerically. We also train a neural network distinguishing vertex from Feynman region. As a check, 4-tachyon contact term in the tachyon potential is computed and a good agreement with the results in the literature is observed. We argue that our algorithm is manifestly independent of number of punctures and scaling it to characterize the geometry of n-string contact interaction is feasible.

Deeper Looks

at some recent progress.

1. Autodiff Everything: Flux Vacua

2. Flows: CY Metrics and Neural Network Theory

Autodiff Everything:
A Slogan for Differentiable Science

A theme in ML, since differentiability opens up new algorithms, analyses.

Here: a pipeline for IIB flux vacua.

Autodiff Everything

JH **You**

write me 2 sentences explaining the basics of the slogan "Autodiff everything" in the context of differentiable science. be clear on why this is interesting and important.

ChatGPT

The slogan "Autodiff everything" in differentiable science advocates for the widespread use of automatic differentiation, a method that computationally derives precise gradients of complex functions. This approach is crucial as it enables more efficient and accurate modeling in various scientific and engineering fields, significantly advancing research and development through improved optimization and analysis techniques.

Big idea: many more things are diff'able than you think, e.g. certain complex simulations, deep NNs. really any computational graph with diff'able comps

e.g. differentiate through numerical ODE solutions, with Diffrax, built on JAX.

JAX, M.D.
A Framework for Differentiable Physics

Samuel S. Schoenholz
Google Research: Brain Team
schsam@google.com

Ekin D. Cubuk
Google Research: Brain Team
cubuk@google.com

Abstract

We introduce JAX MD, a software package for performing differentiable physics simulations with a focus on molecular dynamics. JAX MD includes a number of physics simulation environments, as well as interaction potentials and neural networks that can be integrated into these environments without writing any additional code. Since the simulations themselves are differentiable functions, entire trajectories can be differentiated to perform meta-optimization. These features are built on primitive operations, such as spatial partitioning, that allow simulations to scale to hundreds-of-thousands of particles on a single GPU. These primitives are flexible enough that they can be used to scale up workloads outside of molecular dynamics. We present several examples that highlight the features of JAX MD including: integration of graph neural networks into traditional simulations, meta-optimization through minimization of particle packings, and a multi-agent flocking simulation. JAX MD is available at www.github.com/google/jax-md.

JAX MD: differentiable molecular dynamics simuls.

Differentiable Cosmological Stasis: WIP with Pandya.

Autodiff IIB Flux Vacua with JAXVacua

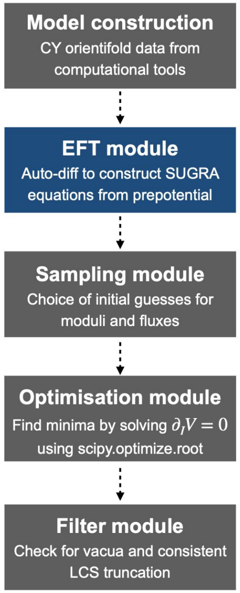
JAX: a Google package, diff'able numpy, NN library extensions sit on top.

A number of nice features (other libs too), including:

- **autodiff.** what we've been talking about.
- **jit.** just-in-time compilation, converts functions to machine code at run-time, gives speedups.
- **vmap, pmap.** automatic vectorization allows for clean implementation across many CPU cores.

JAXVacua: pipeline for finding vacua with JAX.

[Dubey, Krippendorf, Schachner]



Model construction
CY orientifold data from computational tools

↓

EFT module
Auto-diff to construct SUGRA equations from prepotential

↓

Sampling module
Choice of initial guesses for moduli and fluxes

↓

Optimisation module
Find minima by solving $\partial_I V = 0$ using scipy.optimize.root

↓

Filter module
Check for vacua and consistent LCS truncation

JAXvacua

EFT module

- Flexible code (i.e. re-use for different CY data) for EFT properties with JAX

CY data		Prepotential		Periods		EFT properties
κ_{ijk}, c_2, GVs	$\dashrightarrow$	F	$\xrightarrow{\text{auto-diff}}$	Π, K, W	$\xrightarrow{\text{auto-diff}}$	$V, \partial_I \partial_J V, M_{3/2}, \ldots$
		Only hardcoded input!				

- Auto-diff: machine precision derivatives; easy to implement and adapt to different properties.

Code example:

```
def prepot(moduli):

    return np.einsum("ijk,i,j,k",kappa,moduli,moduli,moduli) + ...

gradient_prepot = jax.grad(prepot,holomorphic=True)(moduli)
```

On paper: $F = -\frac{1}{6}\kappa_{ijk} Z^i Z^j Z^k + \ldots \; , \quad \partial_i F = -\frac{1}{2}\kappa_{ijk} Z^j Z^k + \ldots$

thanks @ Sven for sharing slides.

JAXvacua

Numerical results — $h^{1,2} \leq 25$

Scaling behaviour at larger $h^{1,2}$

$h^{1,1}$	$h^{1,2}$	Q_{D3}	success rate	$\sharp$vacua
213	5	220	50%	1,370,842
244	10	256	16%	498,545
399	15	416	7%	168,291
350	20	372	$< 1\%$	36
245	25	272	$< 1\%$	1

Success rate decreases rapidly because

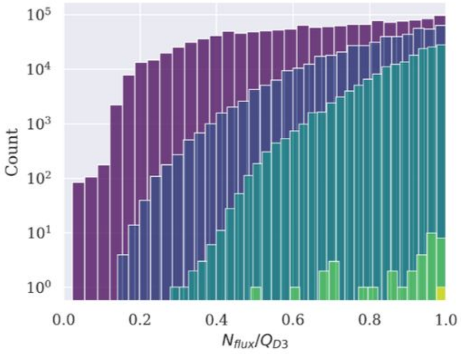
- high dimensionality means slower evaluation time
- harder to perform numerical optimisation
- phase of Kähler cone becomes narrower `[Demirtas et al. 1808.01282]`

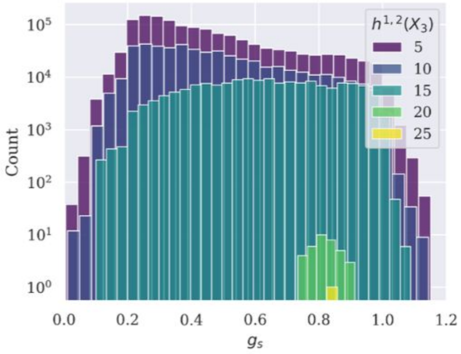
Important to stress: sampling with $N_{flux} \leq Q_{D3}$

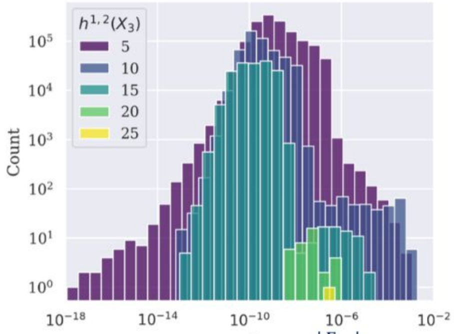
much harder than allowing $N_{flux} \to \infty$.

We actually looked at examples with $h^{1,2} > 100$

and found solutions with $N_{flux} \gg Q_{D3}$

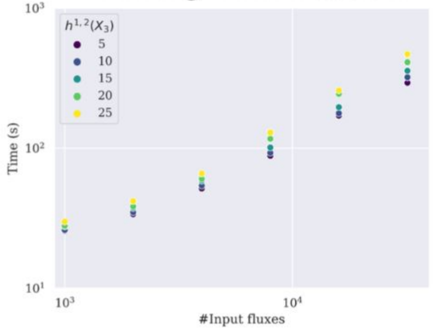




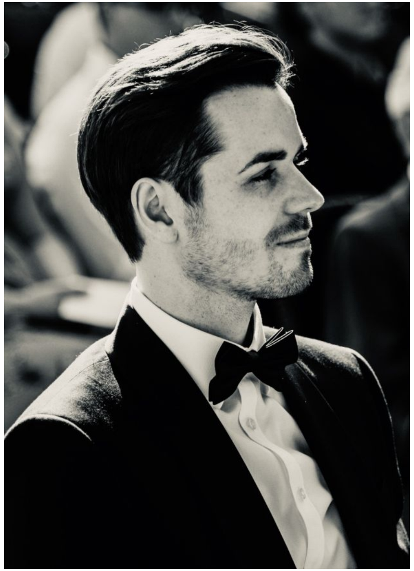


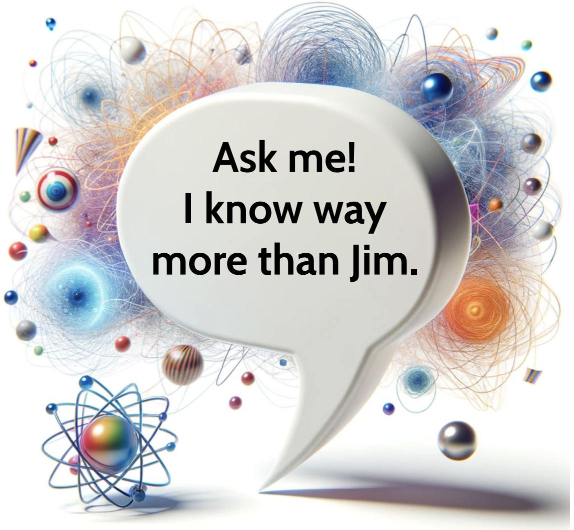
$\varepsilon = \frac{|F_{inst}|}{|F_{pert}|}$

Timing Minimisation:



thanks @ Sven for sharing slides.




Ask me!
I know way
more than Jim.

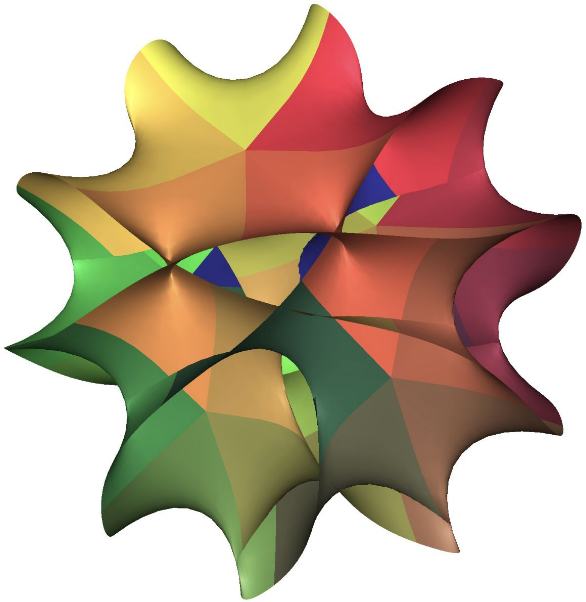
Flows: Calabi-Yau Metrics and Neural Network Theory

we know ***zero*** (non-trivial) compact CY metrics.
modeling them with NNs gives SOTA.

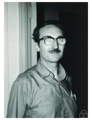
Training corresponds to *geometric flows*
and encompasses famous results in mathematical physics.

Similar apps. in cosmology!
[Krippendorf, Spannowsky]

Calabi-Yau Manifolds: A Fact We Don't Advertise



CY manifold: solutions of string theory, low energy physics depends on geometry and topology.



Yau's Miraculous Theorem:
let X be a complex (Kahler) manifold,
then if certain *topological* feature holds ($c_1 = 0$),
you get *geometry*, a unique Ricci-flat metric.

$$R_{ij} = 0$$

(Ricci curvature tensor)

Conditions often satisfied?
10^{755} < # elliptic CY 4-fold topological types < ∞.
$10^{200?}$ < # Kreuzer-Skarke CY3 < ∞.

How many metrics are known?
The proof of metric existence is non-constructive.
For X compact, *zero* are known. Need approximation.

Neural Network Calabi-Yau Metrics

[Anderson, Gerdes, Krippendorf, Raghuram, Ruehle]
[Douglas, Lakshminarasimhan, Qi]
[Jejjala, Mayorga Pena, Mishra]

see also: [Ashmore, He, Ovrut]
[Larfors, Lukas, Ruehle, Schneider][Krippendorf, Gerdes]

"Let the neural network be the metric!"
- above authors.

Neural network depends on parameters θ, which provide a variational ansatz:

$$g_{ij}(x) \mapsto g_{ij}(x, \theta)$$

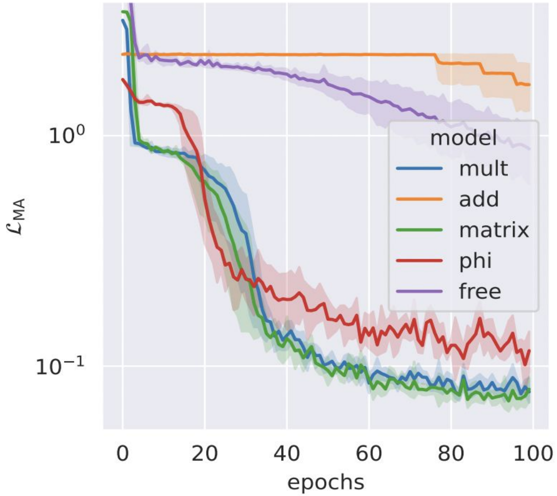
optimize parameters to minimize some objective, e.g. to drive the metric towards Ricci-flatness.

See also: NN as variational ansatz for quantum many-body wavefunction! Minimize energy, e.g.

[Carleo, Troyer] 2017 Infinite NN Context: [J.H., Luo]

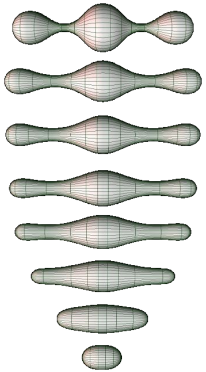
Why this is a good idea: NN's are powerful, universal approximation theorems, etc.

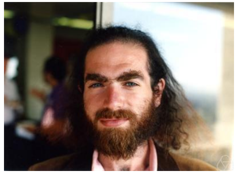
Evidence this is a good idea:



15 mins NN = 30 years w/ conventional techniques.

Flows in Metric Space





Perelman, 2000s: “Ricci flow is a gradient of a scalar functional, and I’ll use it to prove the Poincare conjecture.”

g_0 as a NN

NN metric flow

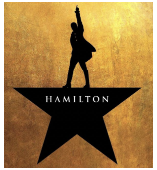
g_0 as a NN

NN metric flow

g_0 as a NN

“NN metric flow” via gradient of scalar functional L.

?



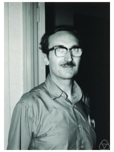
Hamilton, 1980s

g_0

Ricci Flow

$$\frac{dg_{ij}}{dt} = -2R_{ij}$$

g_{CY} fixed point



Theory of Neural Network Metric Flows

[J.H., Ruehle] 2310.19870

Network Dynamics: Neural Tangent Kernel

[Jacot et. al.] 2018

Let f be a NN with parameters θ.
Train via gradient descent of scalar loss.

$$
\begin{aligned}
\frac{df(x)}{dt} &= \frac{d\theta_I}{dt}\frac{df(x)}{d\theta_I} \\
&\overset{\text{G.D.}}{=} -\sum_{\text{data } x'} \frac{df(x)}{d\theta_I}\frac{df(x')}{d\theta_I}\frac{\delta\mathcal{L}}{\delta f(x')} \\
&= -\sum_{\text{data } x'} \Theta(x,x')\frac{\delta\mathcal{L}}{\delta f(x')}
\end{aligned}
$$

Neural Tangent Kernel:

$$\Theta(x,x') := \frac{df(x)}{d\theta_I}\frac{df(x')}{d\theta_I}$$

in infinite limit, becomes
deterministic and t-independent, a fixed function.

Neural Network Metric Flows:

$$\frac{dg_{ij}(x)}{dt} = -\int_X d\mu(x')\,\Theta_{ijkl}(x,x')\frac{\delta l(x')}{\delta g_{kl}(x')}$$

(continuous version. there is discrete, too)

Metric-NTK:

$$\Theta_{ijkl}(x,x') := \frac{\partial g_{ij}(x)}{\partial\theta_I}\frac{\partial g_{kl}(x')}{\partial\theta_I}$$

non-local, t-dependent, mixes components.

Perelman's Formulation:

local, no component mixing.
F: Einstein-dilaton theory.

$$\frac{dg_{ij}}{dt} = \frac{\delta\mathcal{F}[\phi,g]}{\delta g_{ij}(x)}$$

Question: do some architectures yield Perelman?
must overcome these differences. **Yes!**

Summary of Theory Results:

[J.H., Ruehle] 2310.19870

NN metric flow

∞-NN metric flow

Local Metric Flow

$$\frac{dg_{ij}(x)}{dt} = -\bar{\Omega}(x)\frac{\delta l(x)}{\delta g_{ij}(x)}$$

X Perelman's Ricci Flow

X State of the art Calabi-Yau Metrics good for a reason, feature learning.

$$\frac{dg_{ij}(x)}{dt} = -\int_X d\mu(x')\,\bar{\Theta}_{ijkl}(x,x')\frac{\delta l(x')}{\delta g_{kl}(x')}$$

$$\frac{dg_{ij}(x)}{dt} = -\int_X d\mu(x')\,\Theta_{ijkl}(x,x')\frac{\delta l(x')}{\delta g_{kl}(x')}$$

$$\Theta_{ijkl}(x,x') := \frac{\partial g_{ij}(x)}{\partial \theta_I}\frac{\partial g_{kl}(x')}{\partial \theta_I}$$

$$\lim_{N\to\infty} \Theta_{ijkl}(x,x') = \mathbb{E}_\theta[\alpha_{ijkl}(x,x')] =: \bar{\Theta}_{ijkl}(x,x')$$

Θ_{ijkl} **difficult:** stochastic, t-dependent, hard to compute.

$\bar{\Theta}_{ijkl}$ **easier:** deterministic, t-indep, fixed function.

$\bar{\Omega}$ **easiest:** deterministic, t-indep, local

Kernel Methods: Failing for Deep Reasons

We trained infinite NNs for CY-metrics.

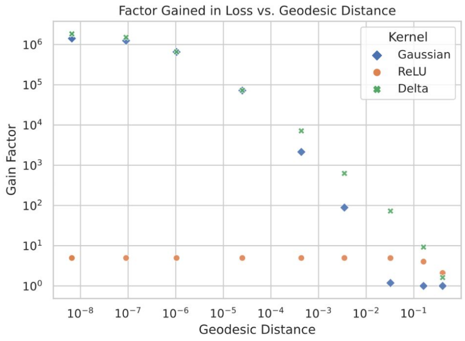
Results not great! Great on test, bad on train.

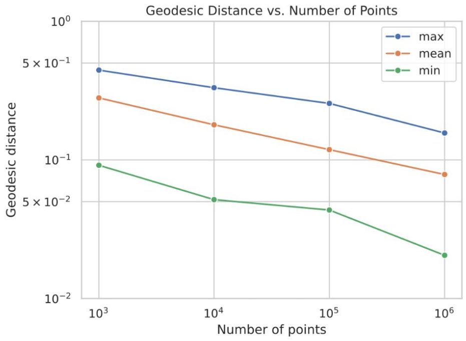
Kernel Learning and CY-Theorem:

- NNs in frozen-NTK regime don't learn features.
- therefore, different infinite NNs have different metric updates, due to different fixed kernels.

- the plethora of kernels and updates at odds with the uniqueness of the CY metric.

- **key:** finite-NNs have *evolving* kernels that can in principle learn the right kernel for the job. This idea is borne out in experiments.





$$\frac{dg_{ij}(x)}{dt} = -\int_X d\mu(x')\,\bar{\Theta}_{ijkl}(x,x')\,\frac{\delta l(x')}{\delta g_{kl}(x')}$$

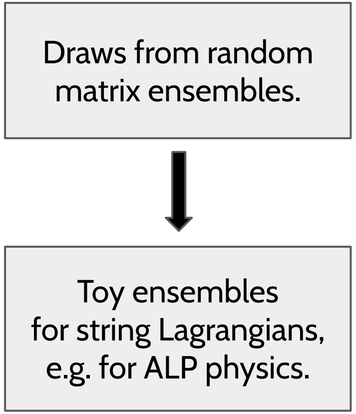
Dreams

come back to the oracle.
what might we really want?

how should we do
statistics in string theory?

Generative Models for String Theory

Random Matrix Theory for Strings:



A Global View on The Search for de-Sitter Vacua in (Type IIA) String Theory

Xingang Chen[1], Gary Shiu[2,3], Yoske Sumitomo[3], and S.-H. Henry Tye[3,4]

[1] Center for Theoretical Cosmology, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, CB3 0WA, UK
[2] Department of Physics, University of Wisconsin, Madison, WI 53706, USA
[3] Institute for Advanced Study, Hong Kong University of Science and Technology, Hong Kong
[4] Laboratory for Elementary-Particle Physics, Cornell University, Ithaca, NY 14853, USA

Abstract

The search for classically stable Type IIA de-Sitter vacua typically starts with an ansatz that gives Anti-de-Sitter supersymmetric vacua and then raises the cosmological constant by modifying the compactification. As one raises the cosmological constant, the couplings typically destabilize the classically stable vacuum, so the probability that this approach will lead to a classically stable de-Sitter vacuum is Gaussianly suppressed. This suggests that classically stable de-Sitter vacua in string theory (at least in the Type IIA region), especially those with relatively high cosmological constants, are very rare. The probability that a typical de-Sitter extremum is classically stable (i.e., tachyon-free) is argued to be Gaussianly suppressed as a function of the number of moduli.

The Wasteland of Random Supergravities

David Marsh, Liam McAllister, and Timm Wrase

Department of Physics, Cornell University, Ithaca, NY 14853

We show that in a general $\mathcal{N} = 1$ supergravity with $N \gg 1$ scalar fields, an exponentially small fraction of the de Sitter critical points are metastable vacua. Taking the superpotential and Kähler potential to be random functions, we construct a random matrix model for the Hessian matrix, which is well-approximated by the sum of a Wigner matrix and two Wishart matrices. We compute the eigenvalue spectrum analytically from the free convolution of the constituent spectra and find that in typical configurations, a significant fraction of the eigenvalues are negative. Building on the Tracy-Widom law governing fluctuations of extreme eigenvalues, we determine the probability P of a large fluctuation in which all the eigenvalues become positive. Strong eigenvalue repulsion makes this extremely unlikely: we find $P \propto \exp(-c\,N^p)$, with c, p being constants. For generic critical points we find $p \approx 1.5$, while for approximately-supersymmetric critical points, $p \approx 1.3$. Our results have significant implications for the counting of de Sitter vacua in string theory, but the number of vacua remains vast.

Pro: RMT universality results can be utilized.
Con: may make assumps not reflected by string data.

Alternative Question: can we use string data to learn a generative model that simulates string theory?

ML Perspective: an untrained generative model, it generates samples, but wasn't trained on data.

Use those samples to understand statistical preds?

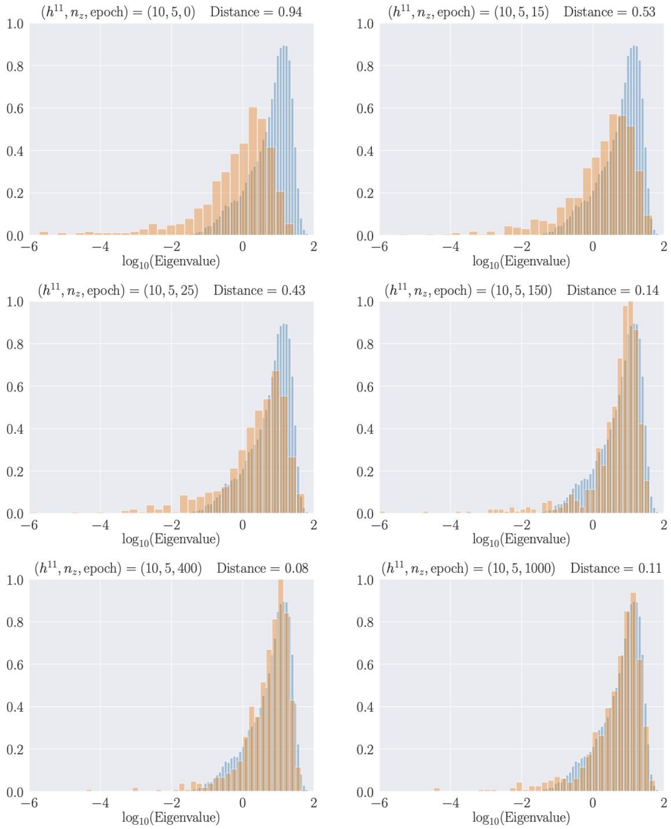
A Baby Step: Learning Random Matrices of String Data

Statistical Predictions in String Theory and Deep Generative Models

James Halverson and Cody Long
Department of Physics, Northeastern University
Boston, MA 02115-5000 USA
(Dated: January 3, 2020)

Generative models in deep learning allow for sampling probability distributions that approximate data distributions. We propose using generative models for making approximate statistical predictions in the string theory landscape. For vacua admitting a Lagrangian description this can be thought of as learning random tensor approximations of couplings. As a concrete proof-of-principle, we demonstrate in a large ensemble of Calabi-Yau manifolds that Kähler metrics evaluated at points in Kähler moduli space are well-approximated by ensembles of matrices produced by a deep convolutional Wasserstein GAN. Accurate approximations of the Kähler metric eigenspectra are achieved with far fewer than h^{11} Gaussian draws. Accurate extrapolation to values of h^{11} outside the training set are achieved via a conditional GAN. Together, these results implicitly suggest the existence of strong correlations in the data, as might be expected if Reid's fantasy is correct.

- **idea:** stat predictions by simulating string theory, i.e. samples from generative models.
- **data:** ensembles of Kahler metrics of KS CY3s at tip of stretched Kahler cone.
- **result:** learned random matrix approximation of the string data, see converging spectrum.
- extrap / interp analysis to other h^{11}.



Bayesian Inference: The Good, the Bad, and the Ugly

The method by which we use data to update our belief in a model has a name: **statistical inference**.

see e.g. [James, Witten, Hastie, Tibshirani]

A Statistical Pipeline for Strings: (akin to ML-stat pipelines in other fields)

1) ML-based methods to draw from approximate distribution on string data (ideally cosmo measure . . .)

2) use string techniques to compute observables.

3) use so-called simulation-based inference to approximate the posterior conditioned on all observed low energy data.

$$p(\theta|D) = \frac{p(D|\theta)\ p(\theta)}{p(D)}$$

4) make predictions in the posterior.

Statistical Inference and String Theory

Jonathan J. Heckman*

Jefferson Physical Laboratory, Harvard University, Cambridge, MA 02138, USA

Abstract

In this note we expose some surprising connections between string theory and statistical inference. We consider a large collective of agents sweeping out a family of nearby statistical models for an M-dimensional manifold of statistical fitting parameters. When the agents making nearby inferences align along a d-dimensional grid, we find that the pooled probability that the collective reaches a correct inference is the partition function of a non-linear sigma model in d dimensions. Stability under perturbations to the original inference scheme requires the agents of the collective to distribute along two dimensions. Conformal invariance of the sigma model corresponds to the condition of a stable inference scheme, directly leading to the Einstein field equations for classical gravity. By summing over all possible arrangements of the agents in the collective, we reach a string theory. We also use this perspective to quantify how much an observer can hope to learn about the internal geometry of a superstring compactification. Finally, we present some brief speculative remarks on applications to the AdS/CFT correspondence and Lorentzian signature spacetimes.

Concern: fundamental statistical learning theory constraint how much we can learn about UV data from IR experiments?

In Conclusion

ML Preliminaries

1) Neural Networks
2) Universal Approximation
3) Architectures
4) Famous Examples and Verbs

Cursory Glances

1) Prediction: Supervised Learning
2) Rigor: Conjecture Gen. + Theorems
3) Structure: Persistent Homology
4) Search: Reinforcement Learning, Gen. Algs, Quantum Annealers

In-Depth Looks

1) Flow: CY-metrics and NN theory
2) Auto-diff everything: Flux Vacua

Dreams

1) Generative Models: Simulating String Theory and Statistical Inference

In Conclusion

String theory and quantum gravity are extremely hard.

If we get to a final answer,
we need formal theory,
sophisticated compute,
and all of us.

New compute in *this* era: ML in the String Landscape.

Thanks!

Questions?

Feel free to get in touch:

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