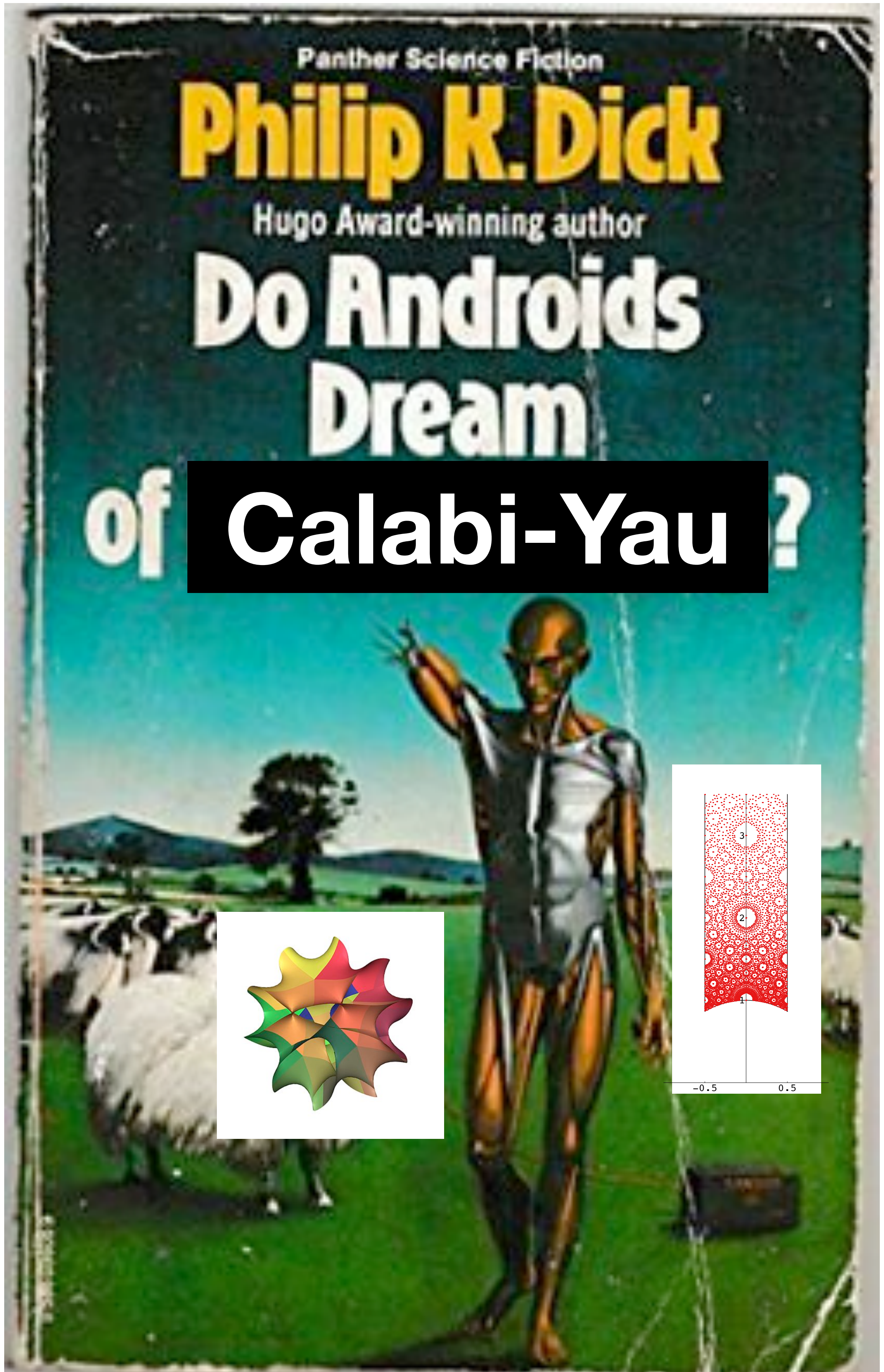


*or: **Machine Learning for String Theory***

13 Sep, ASC school ’22

Alex Cole (U. of Amsterdam)
a.e.cole2@uva.nl
@a_e_cole

Outline

1. What is string data, and why is it interesting?

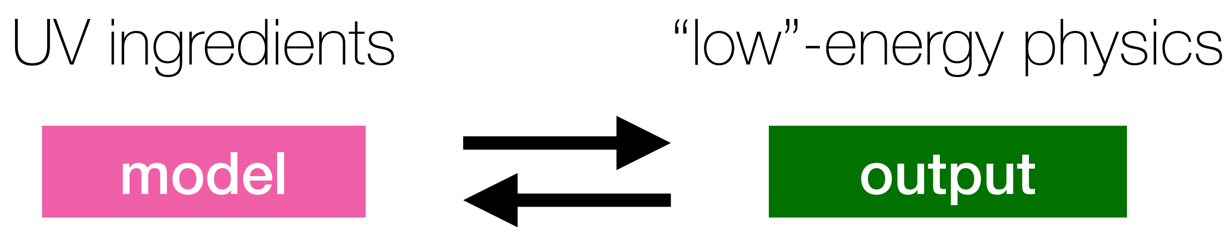
2. Applications of ML methods:

    1. Optimizing vacua via genetic algorithms, reinforcement learning; optimization and representation.

    2. Discovering symmetries in near-optimal vacua, from forward maps.

    3. Learning CY metrics; differentiable programming for strings, supersymmetry.

Forward, Reverse



- Today we will see that ML can help both “forward” and “reverse” directions of studying string theory.
- Forward map: advances in computing Calabi-Yau metrics, identifying symmetries in systems…
- Inverse map: how to pick UV ingredients that give specified low-energy physics?

I. What is string data?

and why is it interesting?

Why is string data interesting?

- **Unifying framework**: string models for both particle physics and cosmology

- **Plenty of data**! In various corners, 10^{500} or $10^{272,000}$ metastable ground states.

- **Computational challenges**: integers $\implies$ NP-hard problems!

- **Data is “pure”**: analytic/noiseless descriptions of systems. Symbolic regression a possibility?

- Data has **interesting mathematical structure**! Opportunity to develop new models, explore exotic symmetries.

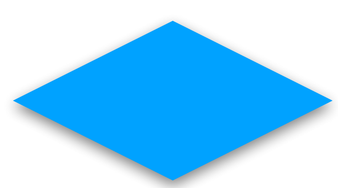
Why is string data interesting?

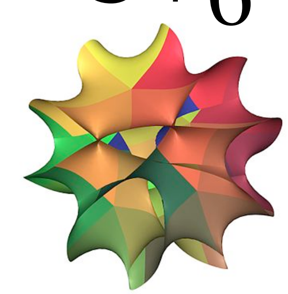
- **Naïveté:** computational methods have been often overlooked in the string theory community, so it is likely significant advances can still be made.

String Data

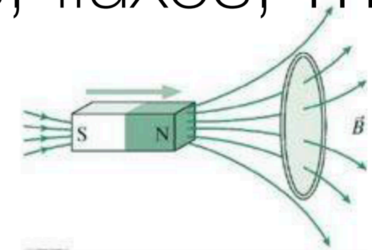
- In 11D, string theory is unique. When compactified to 10D, we have a handful of (related) string theories.
- There are many 4D vacua specified by a compactification:

$$\mathscr{M}_{10} = M_4 \quad \times \quad \text{CY}_6$$





and *discrete* ingredients (branes, fluxes, …) on the background.

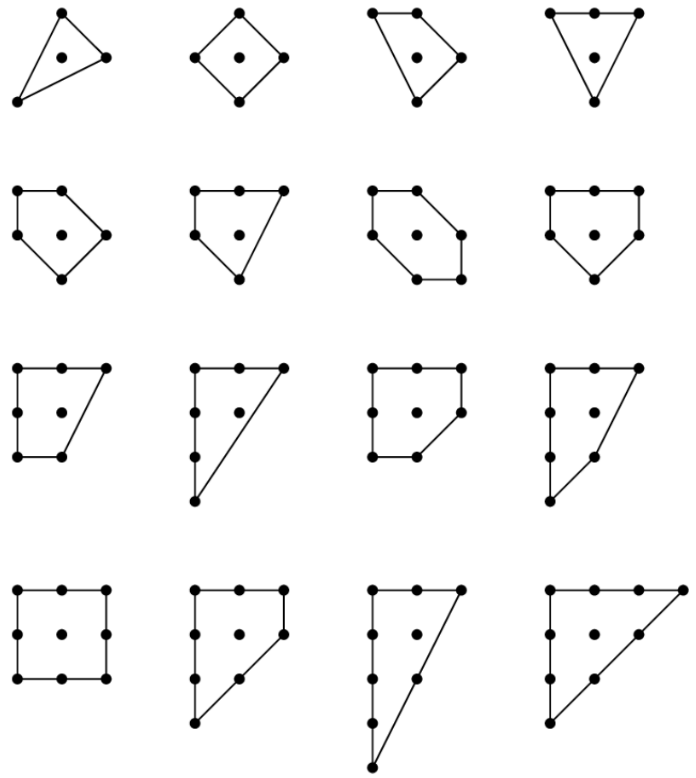


Landscape of Geometries

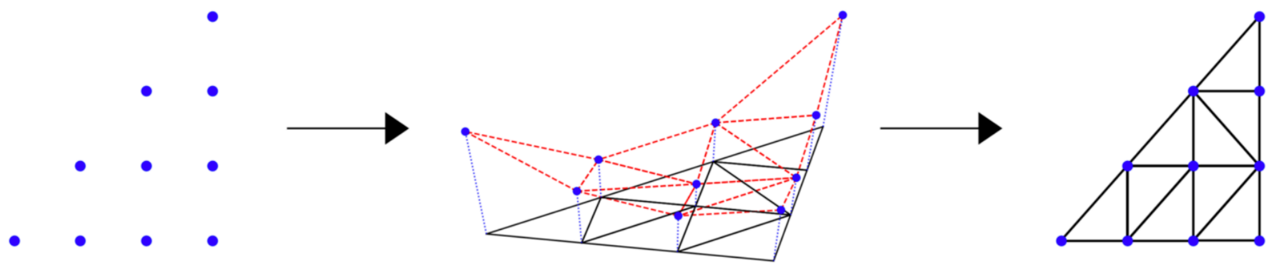
- The exact number of distinct CY manifolds is unknown, but it is at the least huge.
- Various classes of constructions. For example, toric varieties.
    - In [Kreuzer, Skarke ’00], all $473{,}800{,}776$ 4d reflexive polytopes $\Delta \in \mathbb{Z}^4$ were constructed/classified.
    - Each “fine regular star triangulation” of such a Δ gives a toric variety in which the anticanonical hypersurface is a smooth Calabi-Yau.

Triangulations

- Fine: uses all points
- Star: all simplices contain origin
- Regular: descends from higher-dimensional convex hull



The 16 2d reflexive polytopes



regularity

Landscape of Geometries

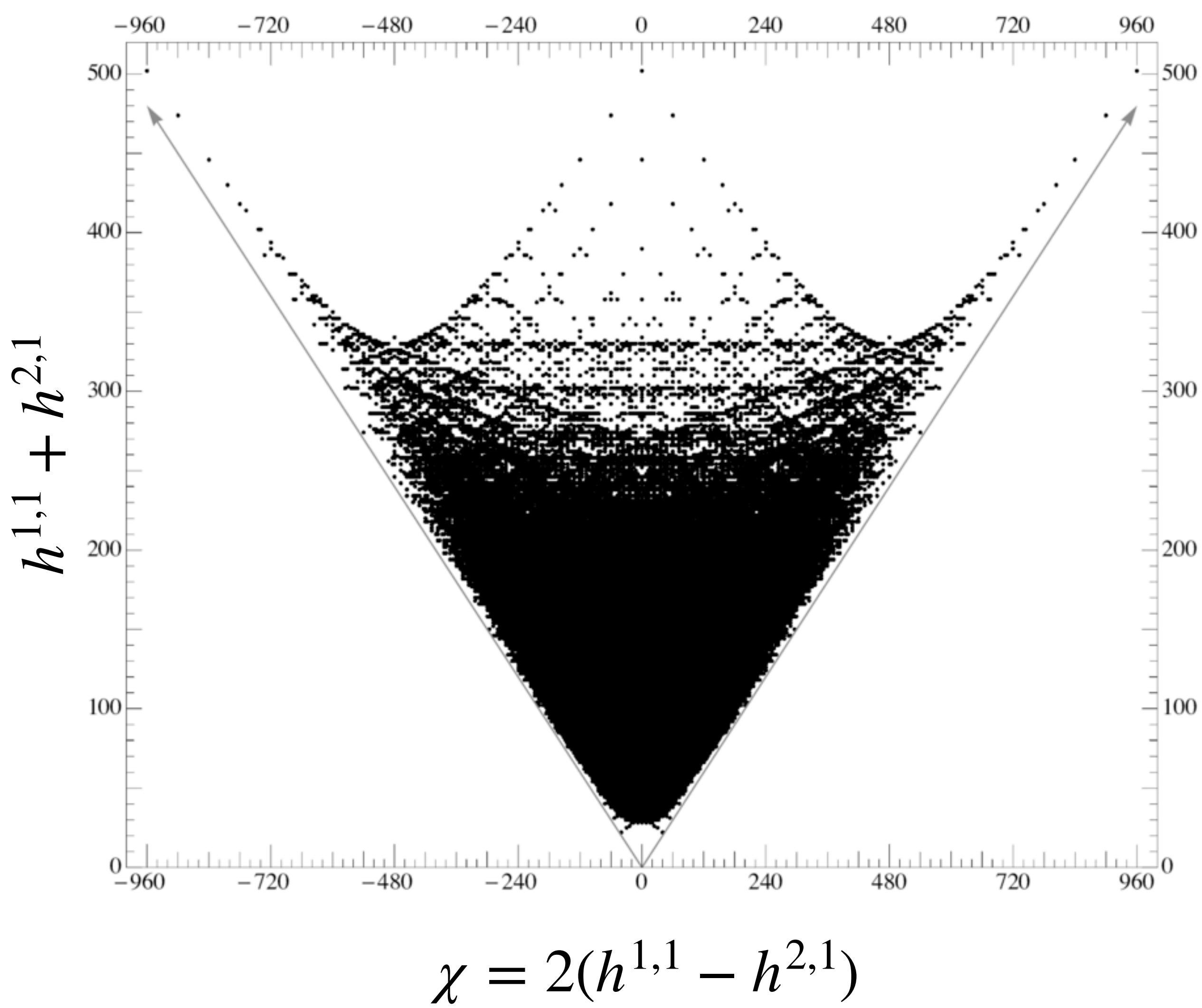
- Number of points in polytope: $h^{1,1} + 4$

- Number of triangulations grows combinatorially:

$$N_{FRST} \leq \binom{4V-1}{h^{1,1}+3}$$

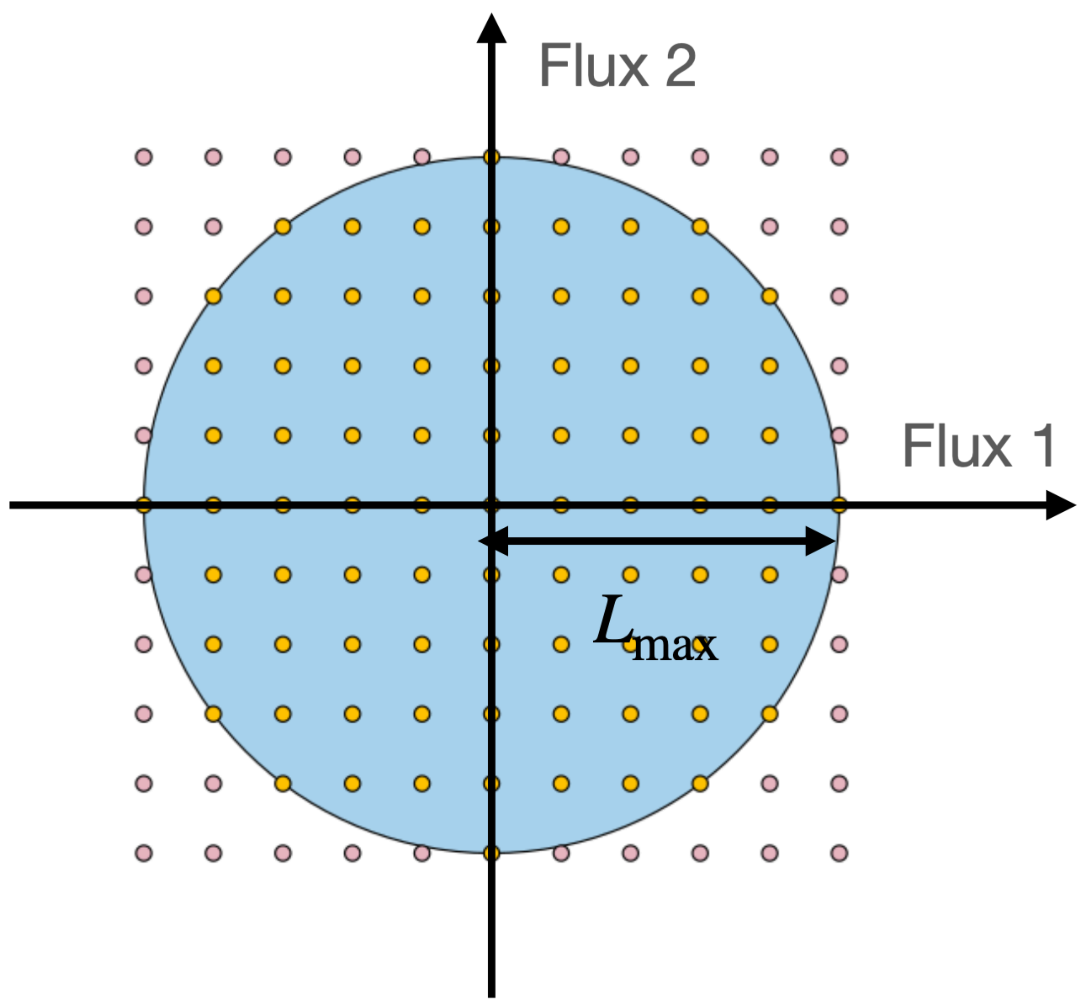
[Demirtas, McAllister, Rios-Tascon]

- Symmetry was a hint of mirror symmetry.



$$h^{1,1}_{\max} = 419 \implies N_{FRST} \leq \binom{14{,}111}{494} \approx 1.53 \times 10^{928}$$

Landscape of Fluxes



- Fix the internal space.
- Integer fluxes are subject to boundedness condition $N_{\text{flux}} < L_{\max} \sim O(10 - 100)$
- Number of fluxes $\sim O(100)$
- So $\left(L_{\max}\right)^{\#\text{ fluxes}} \sim 10^{500}$

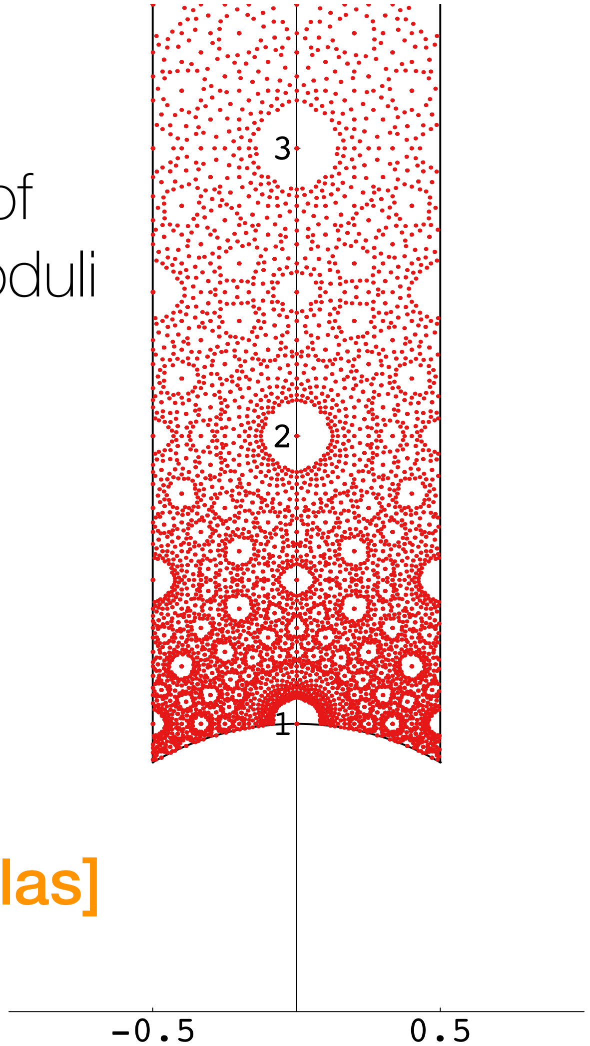
[Ashok, Douglas '03]

F-theory: ∃ geometry with $10^{272,000}$ flux choices [Taylor,Wang]

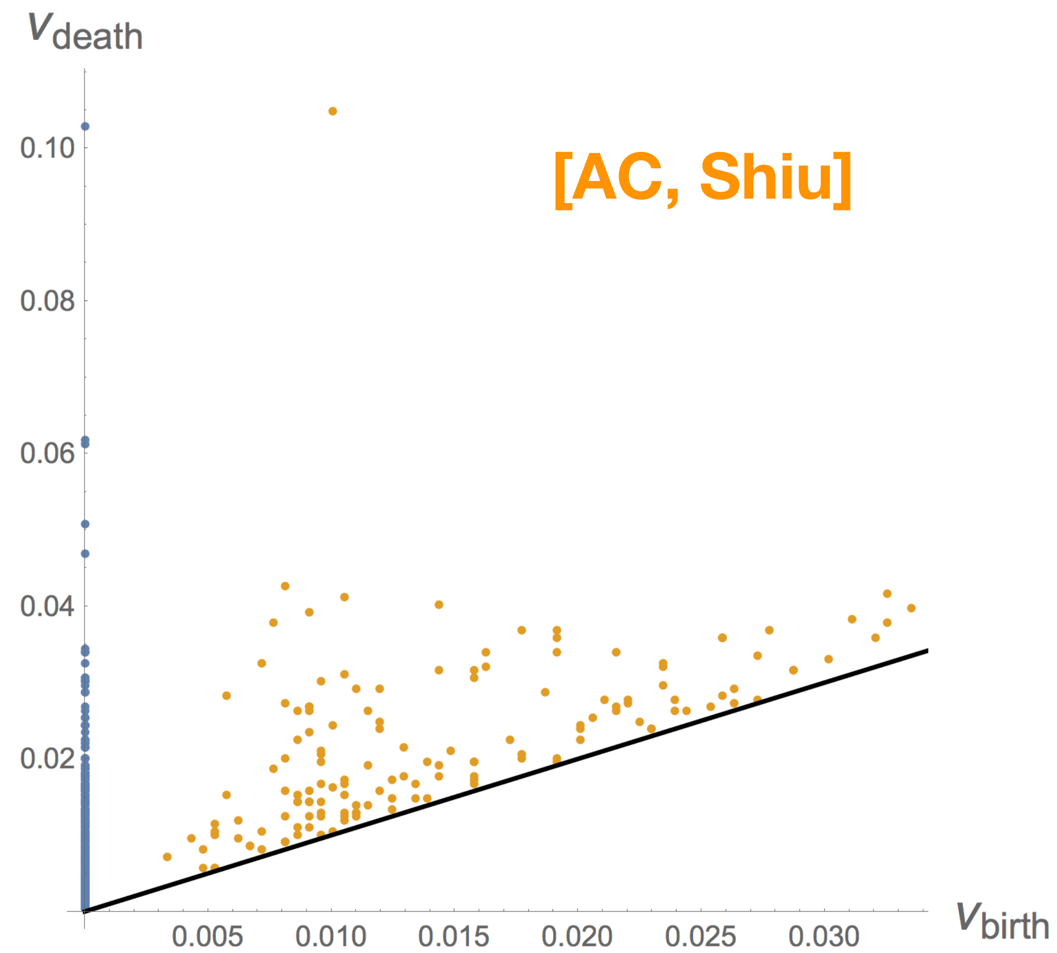
The Shape of Flux Vacua

- Despite the enormous number of vacua, existence of states with specific properties is not guaranteed. Most regions are “voids”

distribution of vacua over moduli space



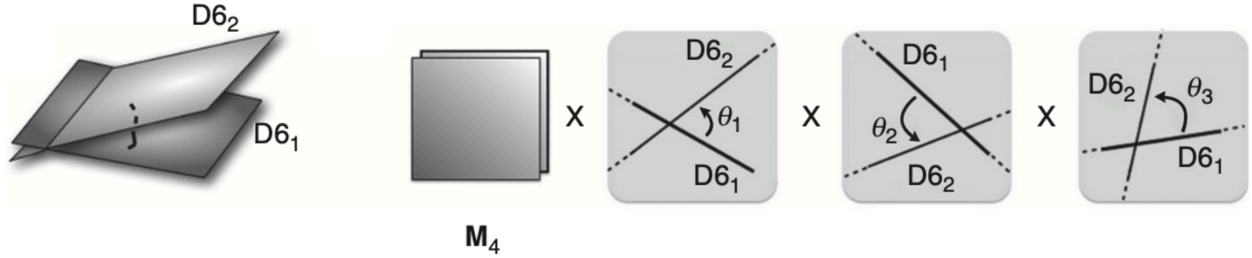
[Denef, Douglas]

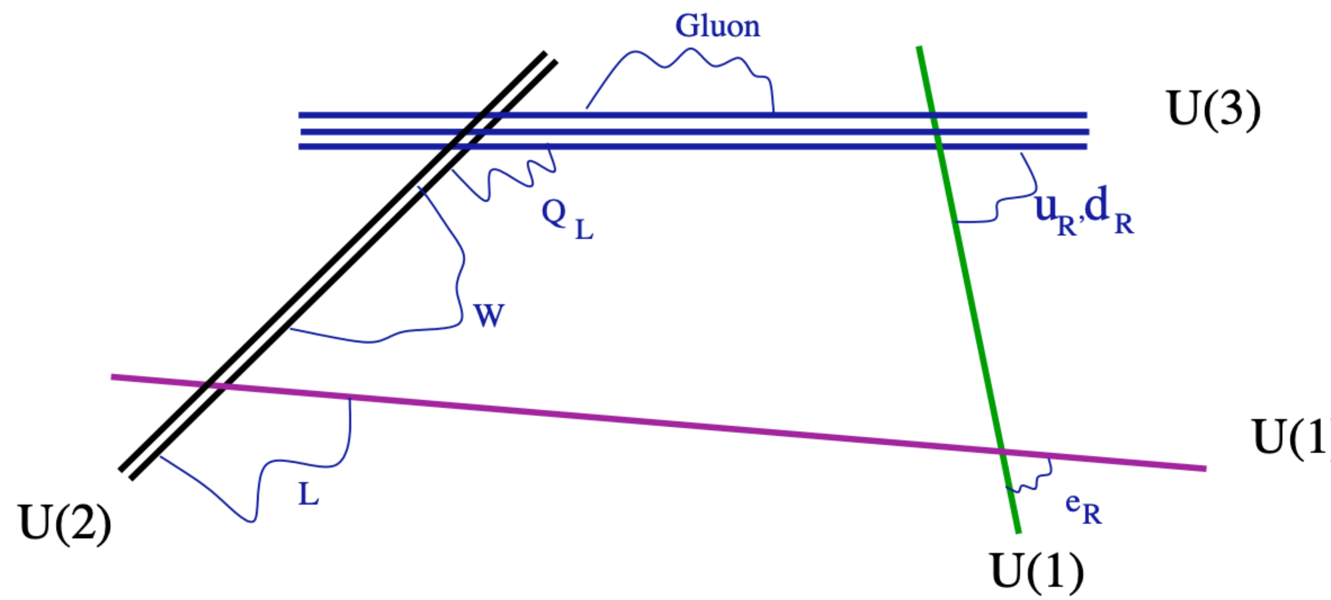


[AC, Shiu]

Intersecting brane landscape

six-torus T^6





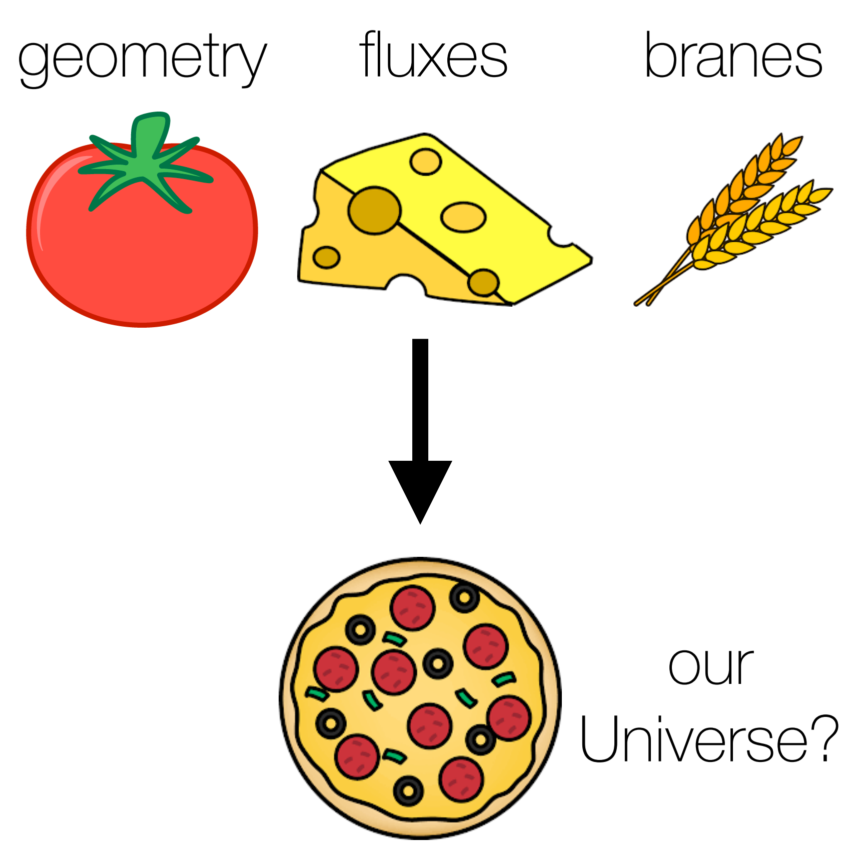
- Intersecting branes lead to non-Abelian gauge groups, chiral fermions.

- Simple toroidal model, count via dynamic programming $\Longrightarrow$ $\sim 215 \times 10^9$ distinct vacua [Loges, Shiu '22]

Cooking with strings

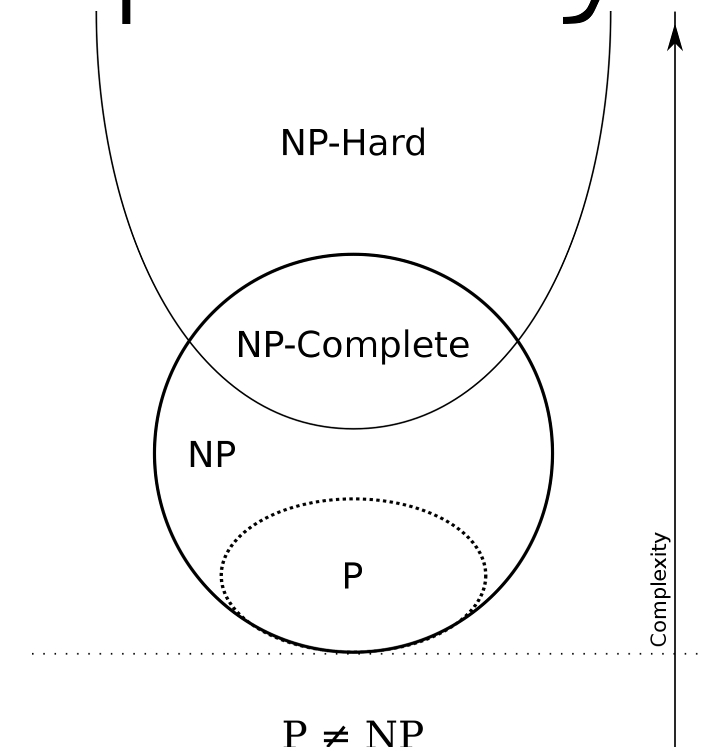
- Question: how to arrange “realistic” vacuum?

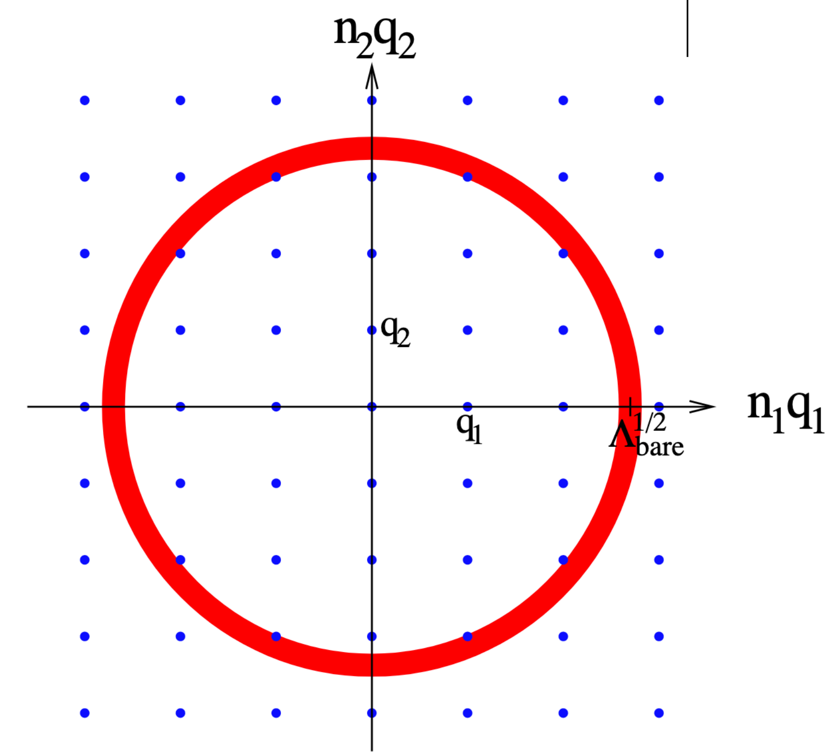
- Brute force search impossible: the string landscape is **large** and **complex**.



Computational Complexity

- In this regime, **computational complexity** (scaling of algorithms with input size) is important.
- Finding specific vacua is **NP-complete** (i.e. hard, probably exponentially!) in toy models [Denef,Douglas; Halverson,Rühle]
- Q: does string theory realize "worst-case" instances of these problems, or is there ***more structure***? Can we circumvent complexity?





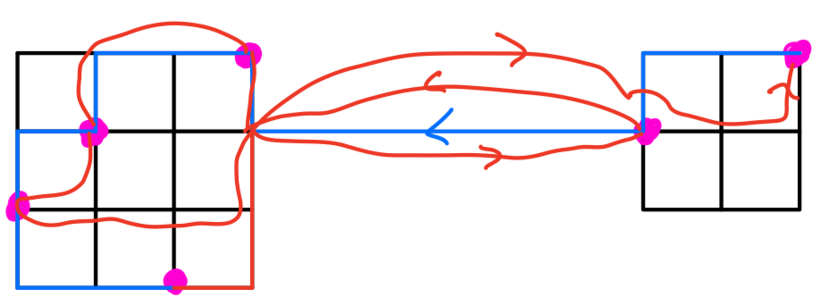
[Bousso,Polchinski]

Machine Learning for Combinatorial Optimization:
a Methodological Tour d'Horizon*

Yoshua Bengio[2,3], Andrea Lodi[1,3], and Antoine Prouvost[1,3]

- Computational complexity not always a death sentence:

Imagine a delivery company in Montreal that needs to solve TSPs. Every day, the customers may vary, but usually, many are downtown and few on top of the Mont Royal mountain. Furthermore, Montreal streets are laid on a grid, making the distances close to the ℓ_1 distance. How close? Not as much as Phoenix, but certainly more than Paris. The company does not care about solving all possible TSPs, but only *theirs*. Explicitly defining what makes a TSP a likely one for the company is tedious, does not scale, and it is not clear how it can be leveraged when explicitly writing an optimization algorithm. We would like to automatically specialize TSP algorithms for this company.



can we discover favorable structure in string data?

Vacua with Small Flux Superpotential

Mehmet Demirtas,[*] Manki Kim,[†] Liam McAllister,[‡] and Jakob Moritz[§]
Department of Physics, Cornell University, Ithaca, NY 14853, USA
(Dated: February 4, 2020)

We describe a method for finding flux vacua of type IIB string theory in which the Gukov-Vafa-Witten superpotential is exponentially small. We present an example with $W_0 \approx 2 \times 10^{-8}$ on an orientifold of a Calabi-Yau hypersurface with $(h^{1,1}, h^{2,1}) = (2, 272)$, at large complex structure and weak string coupling.

"fluxes"

$$W_0 := \sqrt{\frac{2}{\pi}}\left\langle e^{\mathcal{K}/2}\int_X G\wedge\Omega\right\rangle .$$

"moduli"

Vacua with $|W_0| \ll 1$ are rare elements in a large landscape. It is therefore impractical to exhibit vacua with $|W_0| \ll 1$ by enumerating general vacua on a massive scale and filtering out the desired ones. Instead one should pursue algorithms that preferentially find fluxes that lead to vacua with small $|W_0|$.

Introduce trick: solve *approximate* EOM, imposing $|\tilde{W}_0| = 0$ in "large complex structure limit"
Then include small corrections for actual EOM, leading to $0 < |W_0| \ll 1$.

String theory has special structure that can be exploited!

Representation and Optimization

- When should a local search be feasible? Some intuition via **fitness-distance correlation (FDC)**:

$$FDC = \frac{1}{N}\sum_{i}^{N} \frac{(f_i - \bar{f})(d_i - \bar{d}_i)}{\sigma_f \sigma_d}$$

f_i: fitness
d_i: distance to optimum

- Fitness is easy to max/minimize if f_i and d_i are anti/correlated, or $|FDC| \sim 1$.

- Note that FDC depends on encoding, or **representation**.

  - Connection to **dualities** (see [Betzler,Krippendorf]). For nearest-neighbor Ising model:

$$FDC_{\text{neighbor}} \sim -0.3$$
$$FDC_{\text{domain}} = -1 \longrightarrow \text{easier to minimize energy via local operations}$$

Easily Searched Encodings for Number Partitioning

Wheeler Ruml
ruml@das.harvard.edu
Harvard University, Cambridge, Massachusetts

et al.

Abstract

Can stochastic search algorithms outperform existing deterministic heuristics for the NP-hard problem NUMBER PARTITIONING if given a sufficient, but practically realizable amount of time? In a thorough empirical investigation using a straightforward implementation of one such algorithm, simulated annealing, Johnson et al. (1991) concluded tentatively that the answer is "no."

In this paper we show that the answer can be "yes" if attention is devoted to the issue of problem representation (encoding). We present results from empirical tests of several encodings of NUMBER PARTITIONING with problem instances consisting of multiple-precision integers drawn from a uniform probability distribution. With these instances and with an appropriate choice of representation, stochastic and deterministic searches can—routinely and in a practical amount of time—find solutions several orders of magnitude better than those constructed by the best heuristic known (Karmarkar and Karp, 1982), which does not employ searching.

The choice of encoding is found to be more important than the choice of search technique in determining search efficacy. Three alternative explanations for the relative performance of the encodings are tested experimentally. The best encodings tested are found to contain a high proportion of good solutions; moreover, in those encodings, the solutions are organized into a single "bumpy funnel" centered at a known position in the search space. This is likely to be the only relevant structure in the search space because a blind search performs as well as any other search technique tested when the search space is restricted to the funnel tip.

We also show how analogous representations might be designed in a principled manner for other difficult combinatorial optimization problems by applying the principles of parameterized arbitration, parameterized constraint, and parameterized greediness.

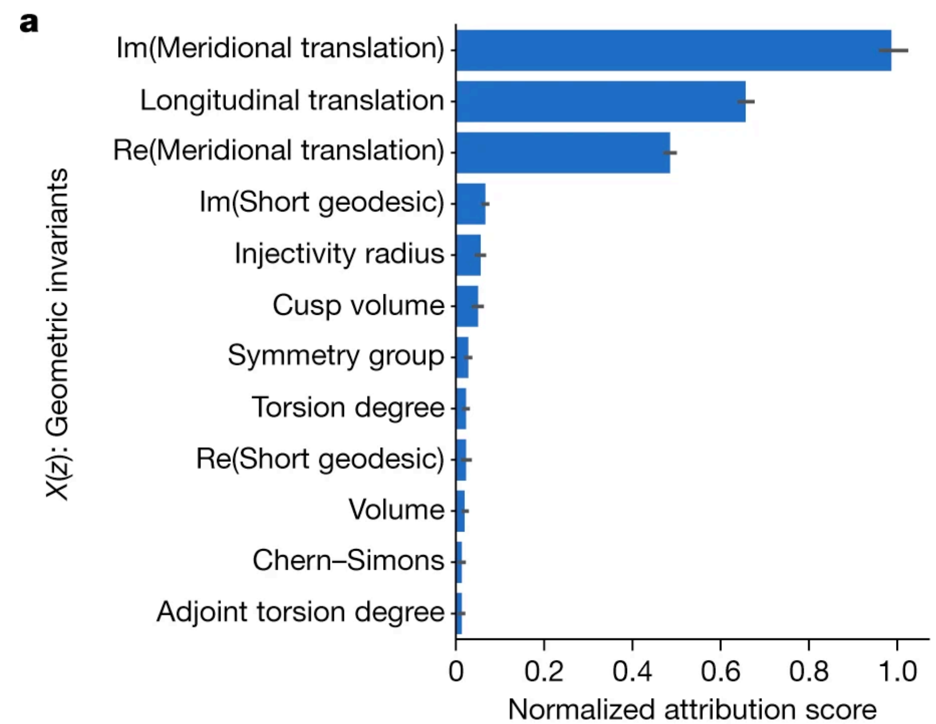
Pure Data

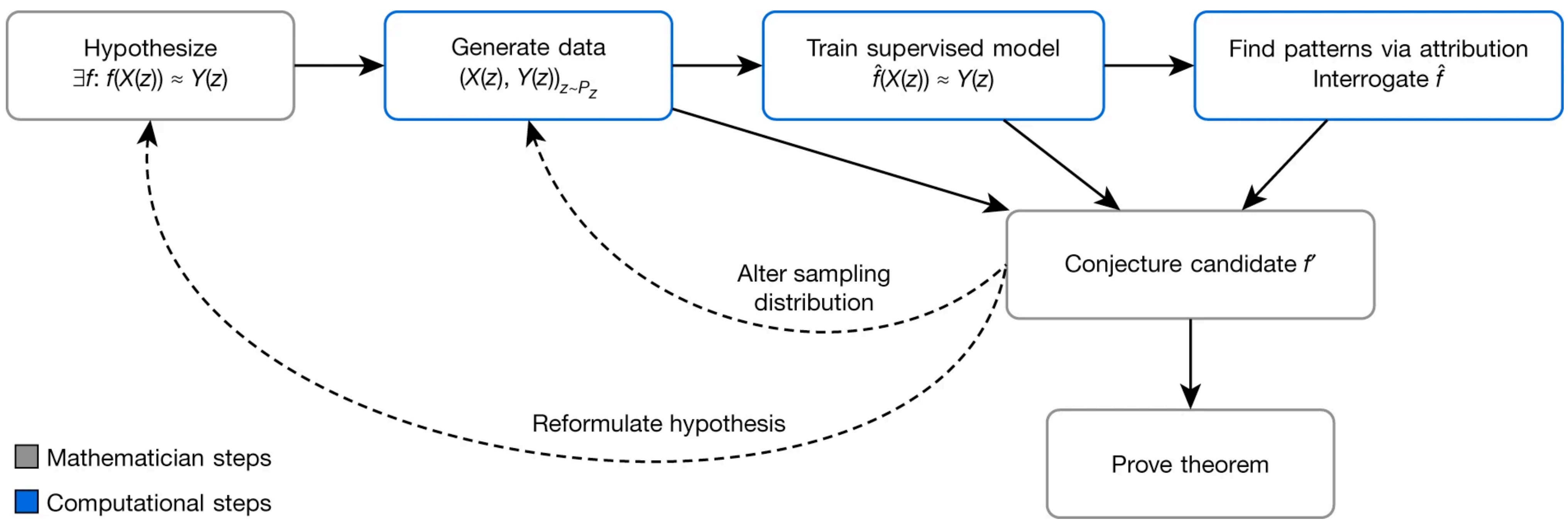
nature > articles > article

Article | Open Access | Published: 01 December 2021

Advancing mathematics by guiding human intuition with AI

Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, Marc Lackenby, Geordie Williamson, Demis Hassabis & Pushmeet Kohli





Pure Data

PySR: High-Performance Symbolic Regression in Python

Symbolic formula as tree, optimize via genetic algorithm

Application to gravitational waves: Wong and Cranmer

Automated discovery of interpretable gravitational-wave population models

Complexity	Score	Loss	Expression for Rate$[\mathrm{yr}^{-1}](M = M/M_\odot)$
16	0.126	0.212	$3.72\mathrm{Cond}\left(\mathrm{M} - 9.27, 0.9^{\mathrm{M}}\right) + 3.72\mathrm{Gauss}(0.52\mathrm{M} - 4.85)$
25	0.501	0.0352	$3.89\left(\mathrm{Gauss}(0.19\mathrm{M} - 6.54) + 0.45\right)\mathrm{Cond}\left(\mathrm{M} - 9.12, 0.91^{\mathrm{M}}\right) + 3.89\mathrm{Gauss}(0.5\mathrm{M} - 4.8)$
48	0.00771	0.0108	$\left(\mathrm{Cond}\left(\mathrm{M} - 9.42, 0.62 \cdot 0.9^{\mathrm{M}}\right) + 1.44\mathrm{Gauss}(0.51\mathrm{M} - 4.88)\right) \times (5.11\mathrm{Gauss}(0.06\mathrm{M} - 4.67) + 7.82\mathrm{Gauss}(0.17\mathrm{M} - 5.86) + 3.26)$

Table 1. Expressions obtained through symbolic regression with *PySR*. In the search we perform, there are 30 equations with different complexities. We select three representative equations from the Pareto front by setting three successive complexity ranges, and selecting the highest scoring expression in each range.

what can we learn about “noiseless” string data?

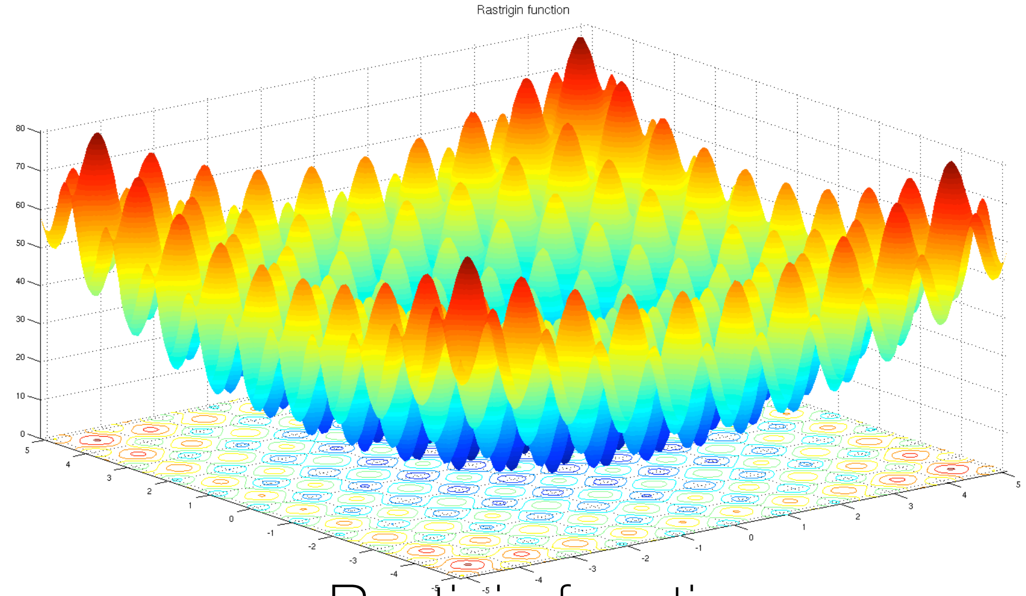
II. Applications

Optimization

Optimization

- Search problem: find $\boldsymbol{x}$ such that $f(x) = y_0$.
- In other words minimize $L(x) = d(f(x), y_0)$
- Lack of gradients (e.g. integer optimization), local minima make this difficult.

optimization in a landscape



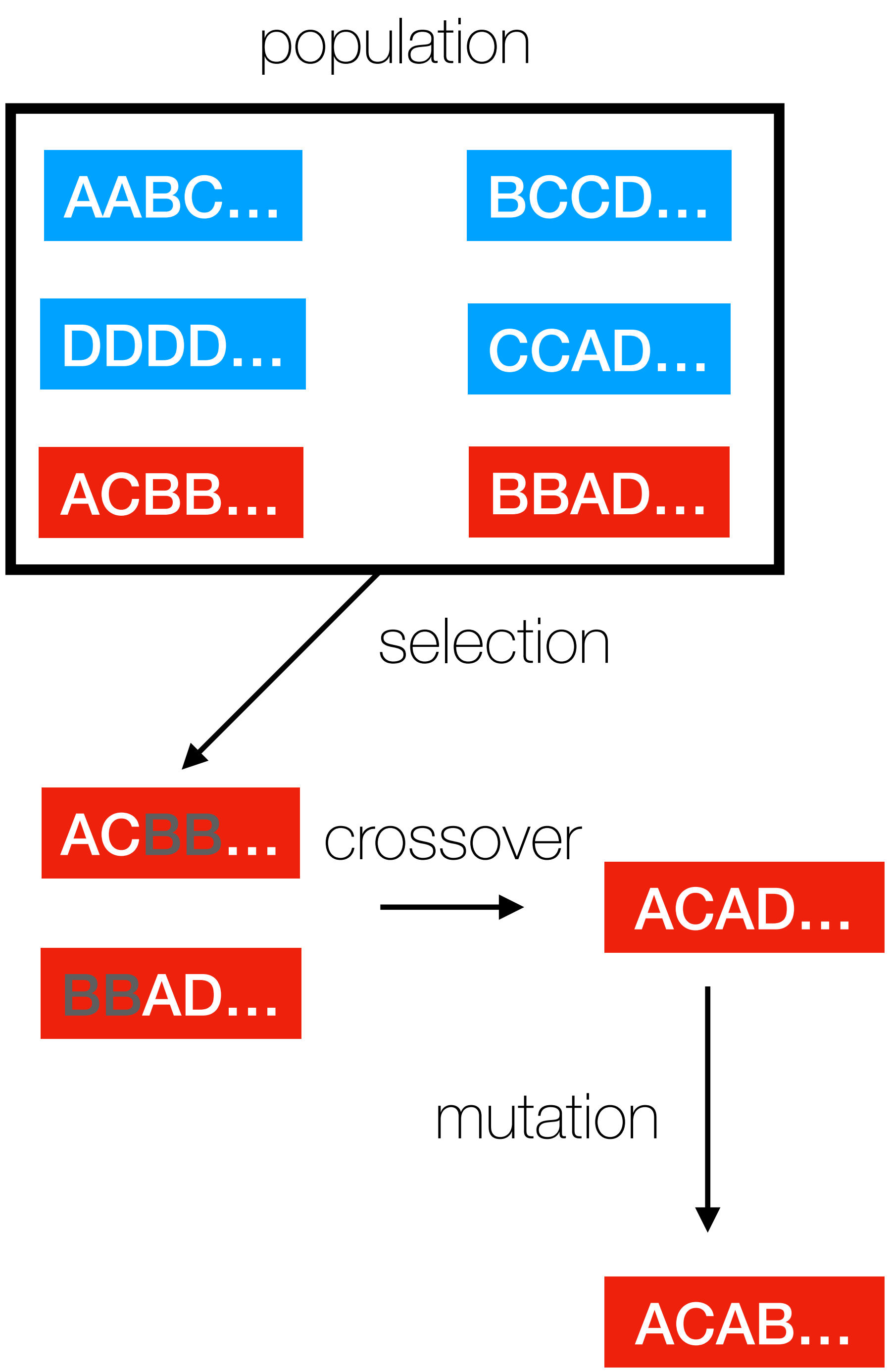
Rastigin function

Genetic Algorithms

- Genetic algorithm [Hollands]: model dynamics after **natural selection**.

    1. Generate a **population** of candidate solutions.
    2. Parents are **selected** according to their fitness.
    3. Parents **breed**: their genotypes are combined according to some predefined set of operators.
    4. Children **mutate** with some probability.

Repeat 2-4 with children replacing parents.



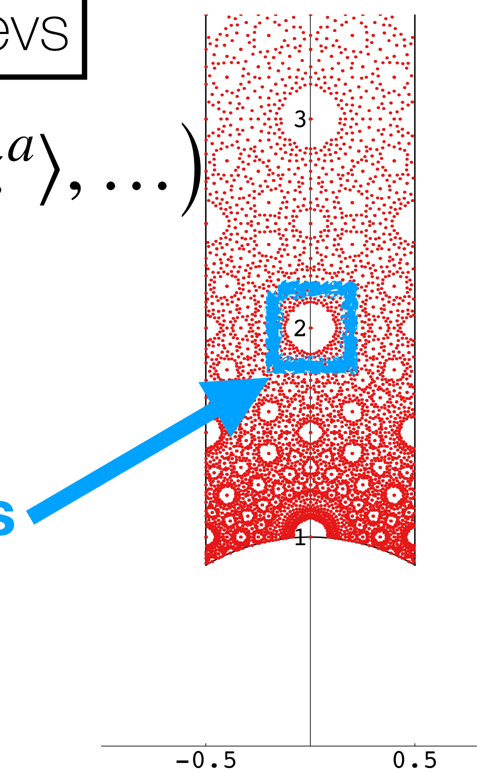
[Abel,Rizos],[Rühle],
[AC,Schachner,Shiu],
[AbdusSalam et al.]…

Optimizing Flux Vacua

- Type IIB flux vacua: **integer** fluxes stabilize complex structure moduli at specific values.
    - Goal: construct vacua with specific physical properties.
- Integers make life harder.

integers   moduli vevs

$$(F_3, H_3) \rightarrow_{DW=0} \big(\langle \phi \rangle, \langle z^a \rangle, \ldots \big)$$



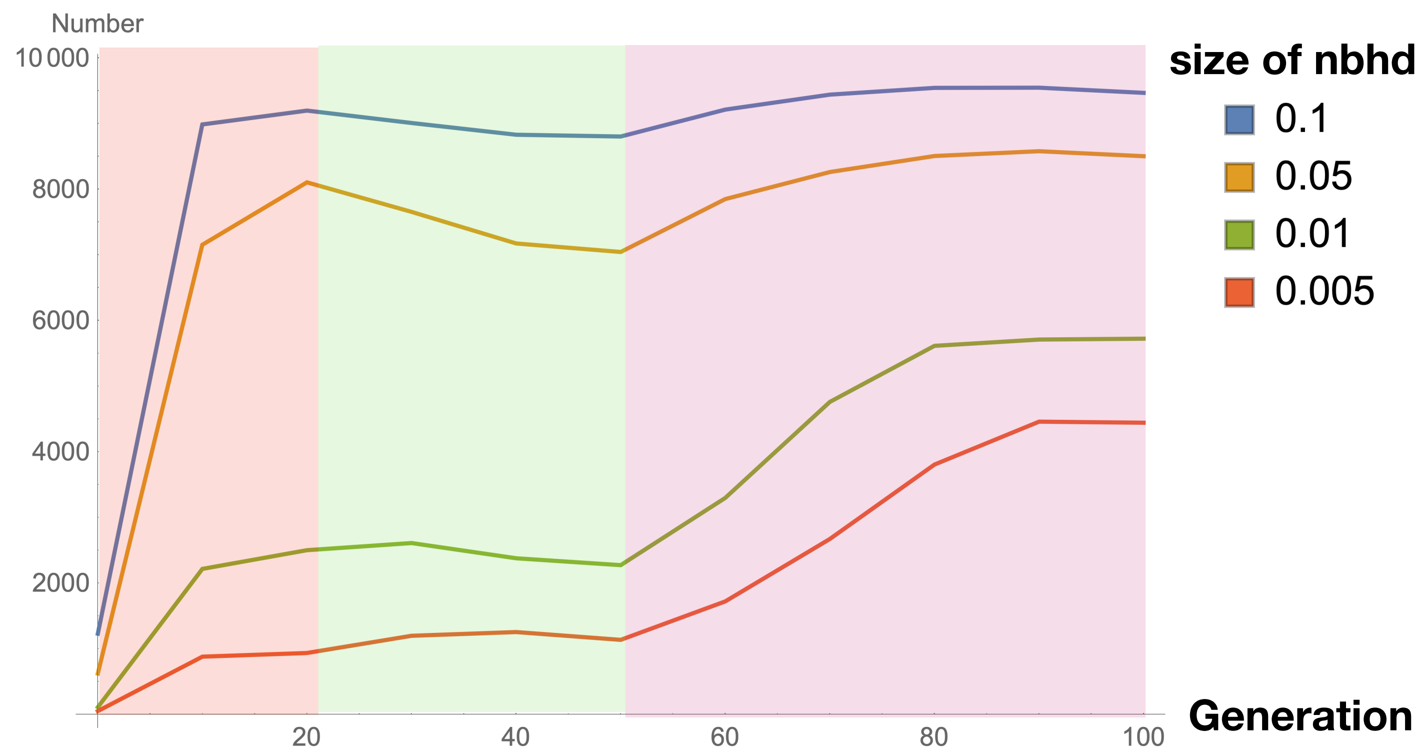
goal: find input that gives specific output

$$\boxed{\big(\langle \phi \rangle, \langle z^a \rangle, \ldots \big) \rightarrow_{DW=0} (F_3, H_3)}$$

our mission

[AC,Schachner,Shiu '19]

Breeding Flux Vacua



rapid evolution

two extrema compete

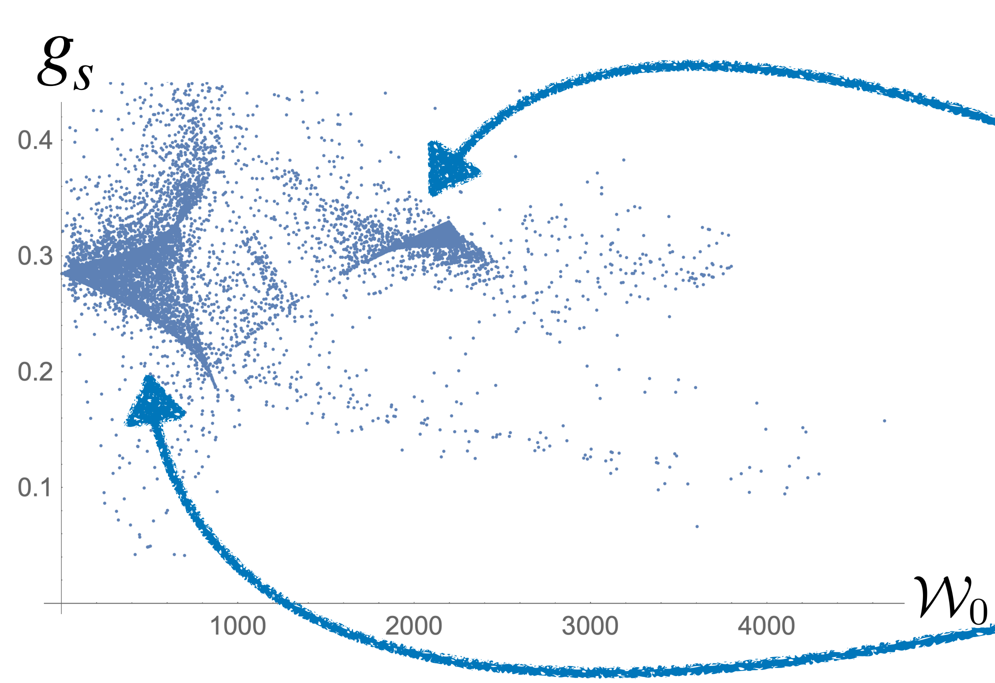
settle into optimum

symmetric $T^6 = (T^2)^3$

Task: search for $g_s^* = 0.3$

Population size

$p = 10000$



Two competing optima, cf. "schema"

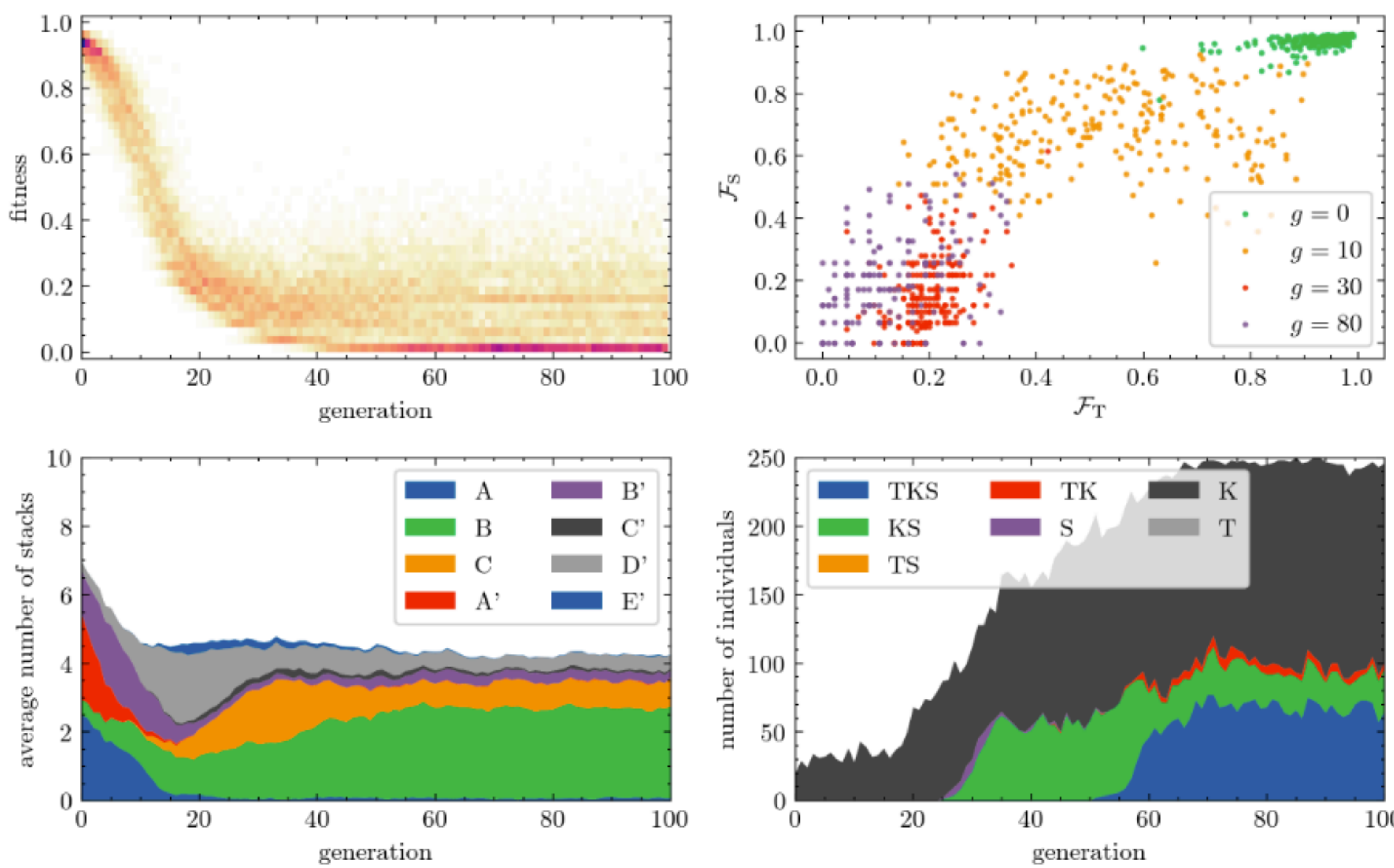
$\mathbf{N}_A = (3, -1, \ * \ , \ * \ , \ * \ , -8, \ * \ ,10)$

arbitrary

$\mathbf{N}_B = (\ * \ ,4, \ * \ , -7, 41 , -2, -33, \ * \ )$

Breeding Branes

- [Loges, Shiu '21]: **genetic algorithms** efficient at generating consistent models with MSSM gauge group! Preliminary study of landscape statistics.



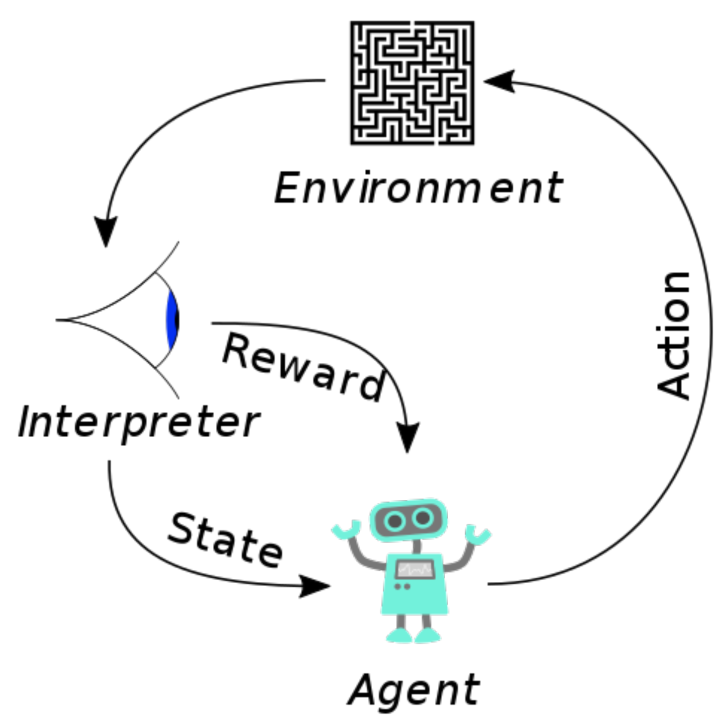
Reinforcement Learning

[book: Sutton&Barto]

- RL: **agent** interacts with **environment** and receives **rewards**. Actions determined by policy π. Policy a function of expected return given policy in state, expected return given action in state $Q(s, a)$.

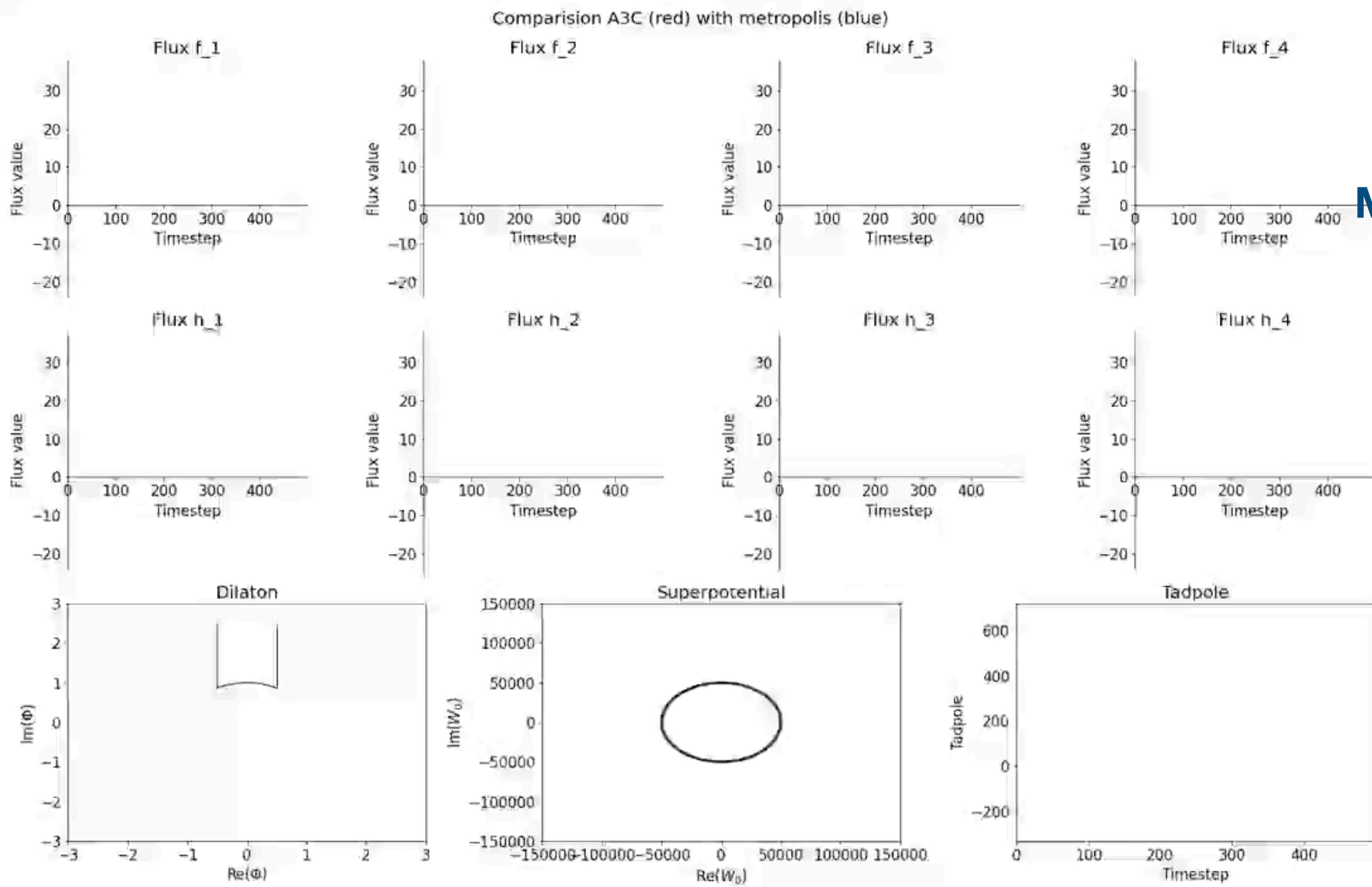
- Balance between **exploration** (discover new strategies) and **exploitation** (reward from known good strategy)

- **Deep** RL: use neural network to estimate e.g. Q.



RL for flux vacua

- **[Krippendorf, Kroepsch, Syvaeri ‘21]**: train RL agent to find flux vacua satisfying various criteria.



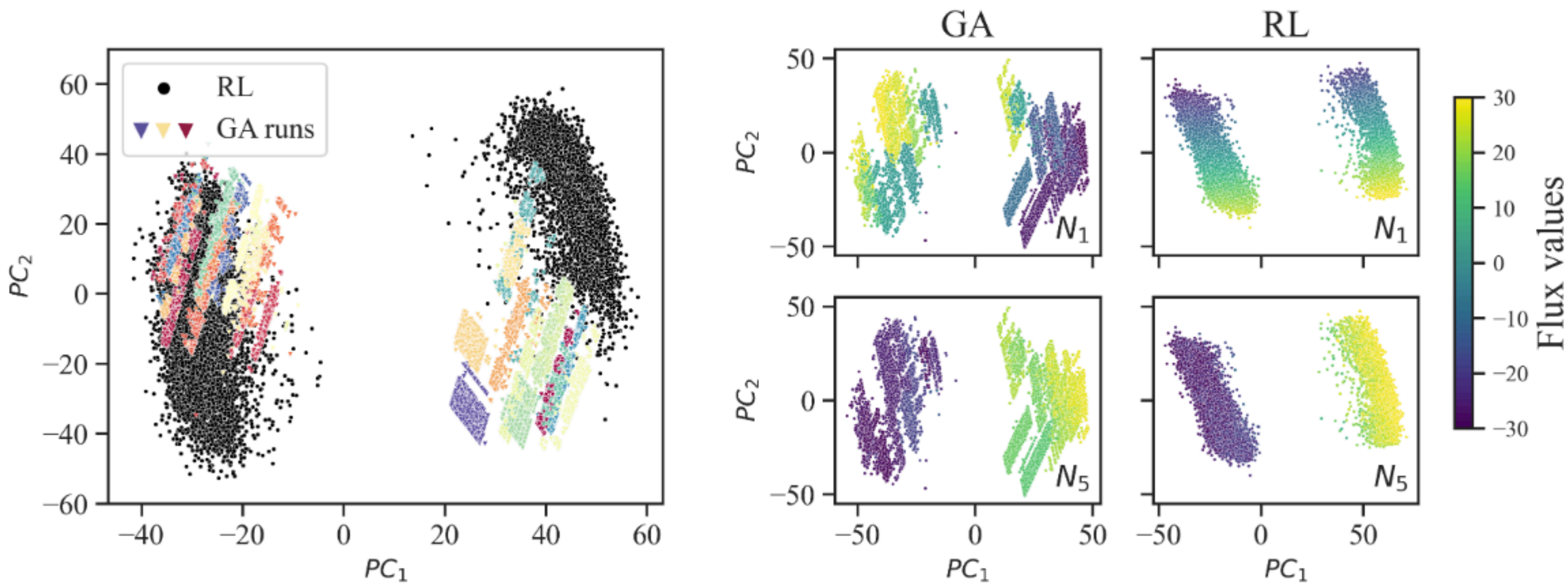
Metropolis

RL (A3C)

Identifying Symmetries

Towards Statistics

- Reduce sampling bias by combining data from GA and RL **[AC, Krippendorf, Schachner, Shiu @ NeurIPS ML4PS '21]**
- Identified unexpected $\mathbb{Z}_2$ symmetry relating near-optimal solutions!



structured learning → symbolic regression (→ expert inspection)

Discovering Symbolic Models from Deep Learning with Inductive Biases

Miles Cranmer[1] Alvaro Sanchez-Gonzalez[2] Peter Battaglia[2] Rui Xu[1]

Kyle Cranmer[3] David Spergel[4,1] Shirley Ho[4,3,1,5]

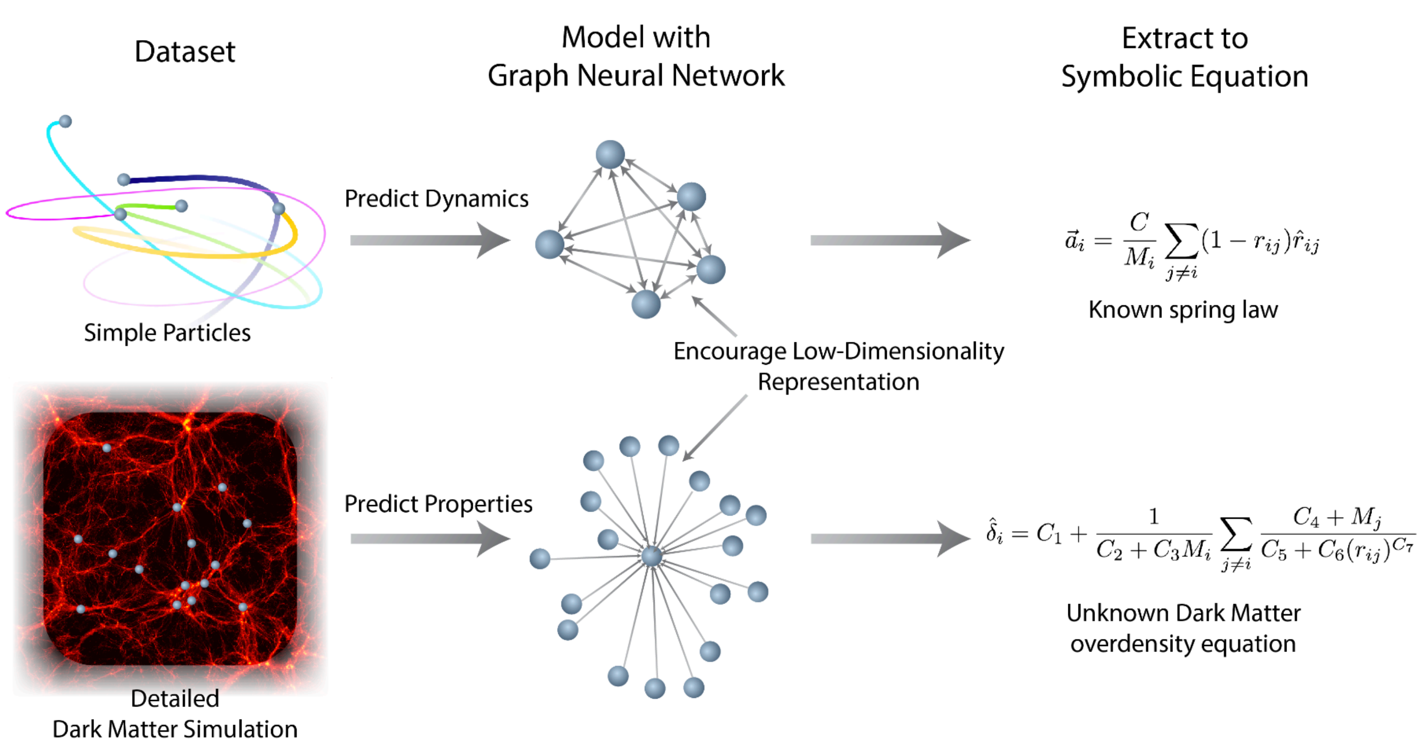
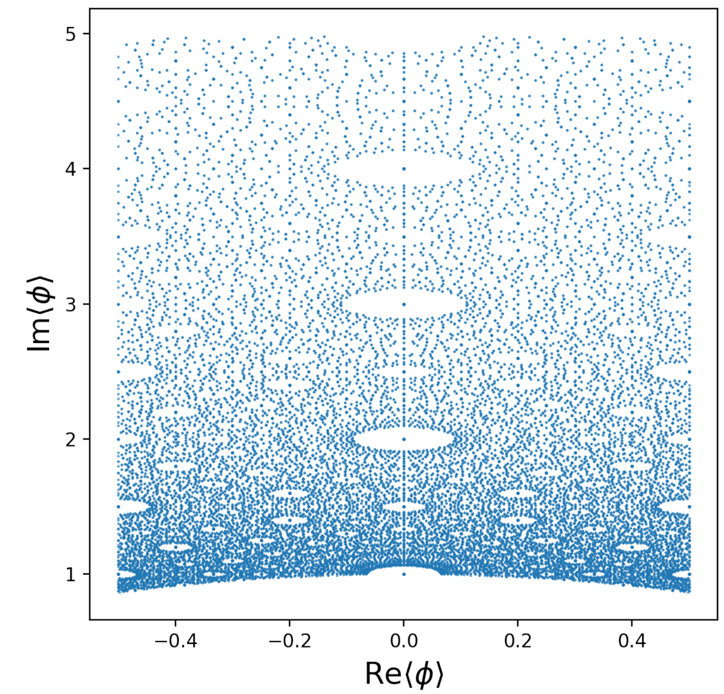


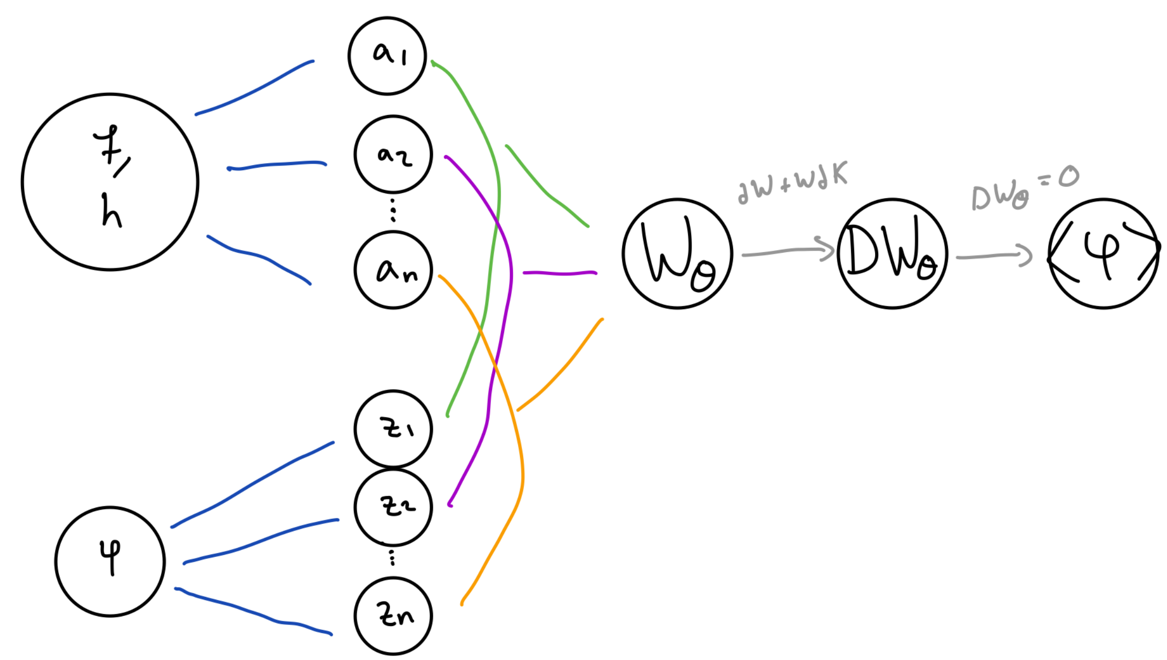
Figure 1: A cartoon depicting how we extract physical equations from a dataset.

Symmetries from symbolic flux vacua

[AC]

- Impose (super)symmetry, holomorphy for “FluxNet.” Root-finding is differentiable.
- Symbolic regression on individual components/ networks.
- Identify $SL(2,\mathbb{Z})$ by subsequent inspection.






```
PySRRegressor.equations = [
        pick     score                                           equation  \
   0          0.000000                                         0.14616045
   1          0.028663                                 (x0 * 0.04347339)
   2          0.030676                            ((x0 * x2) * 0.011989505)
   3          0.127443                       (((x1 + x2) * 0.03578272) * x3)
   4          0.464303                  (((x0 * x2) + (x1 * x3)) * 0.029397454)
   5   >>>>   0.602903  (((x0 * (x2 + 0.21638234)) + (x1 * x3)) * 0.05...
   6          0.012178  (((((x2 + 0.2579701) * x0) + (x1 * x3)) * 0.05...
   7          0.001296  ((((x0 * (x2 + 0.21638234)) + ((x1 * x3) * 0.9...
```

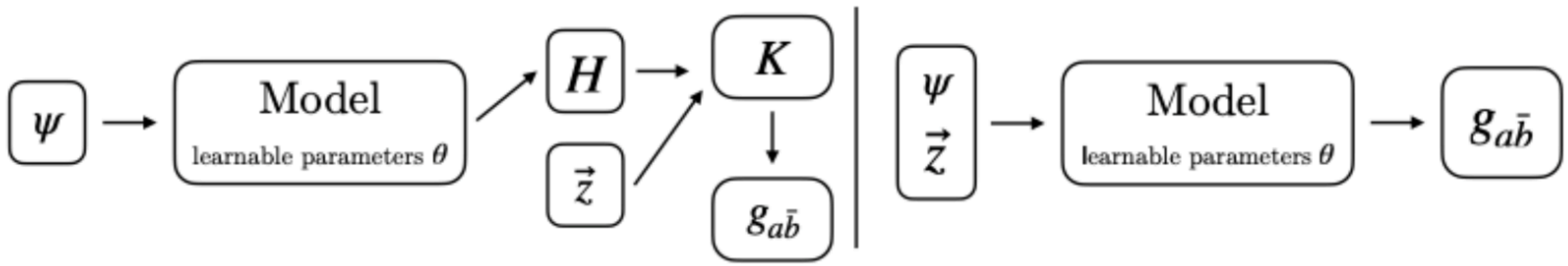
again we drop the small term, giving us $\approx 0.05(x_0x_2 + x_1x_3)$

together, this suggests that the overall result is

$$\sim \frac{x_0x_2 + x_1x_3}{x_2^2 + x_3^2},$$

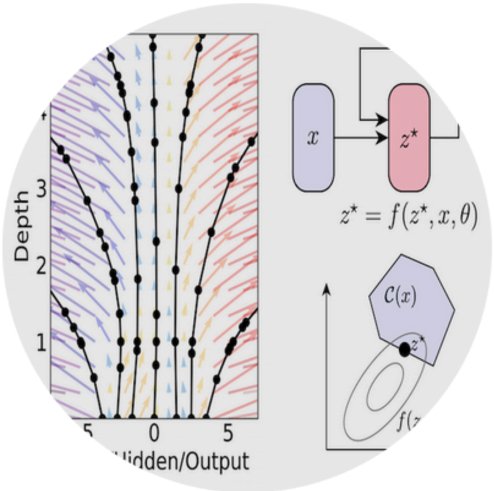
Moduli-dependent Calabi-Yau and $SU(3)$-structure metrics from Machine Learning

Lara B. Anderson[a,1], **Mathis Gerdes**[b,2], **James Gray**[a,3], **Sven Krippendorf**[b,4], **Nikhil Raghuram**[a,5], **Fabian Ruehle**[c,d,6]



- Explicit CY metrics hard to construct w/ conventional methods. They can tell us interesting physics (Yukawa couplings…)
- Q: where to put flexible ansatz Model_θ?
- Sven can tell us more details :-)

2020 NeurIPS tutorial on Deep Implicit Layers



Deep Implicit Layers - Neural ODEs, Deep Equilibirum Models, and Beyond

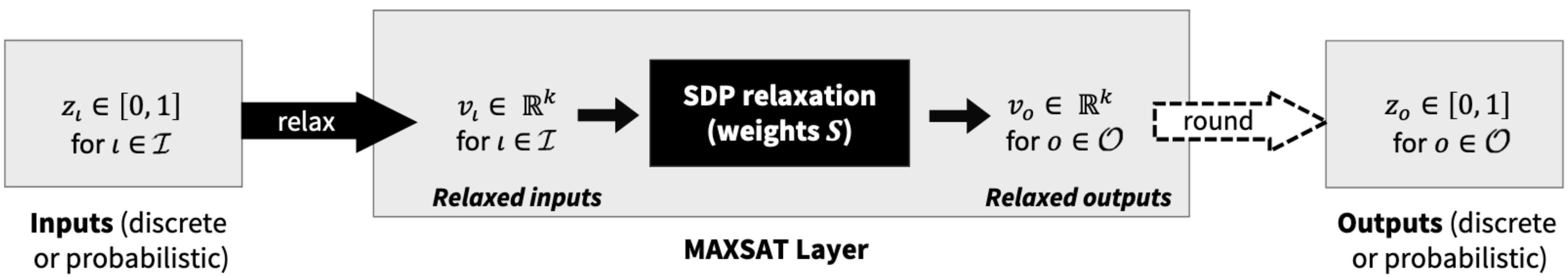
This web page is the companion website to our NeurIPS 2020 tutorial, created by Zico Kolter, David Duvenaud, and Matt Johnson. The page constain notes to accompany our tutorial (all created via Colab notebooks, which you can experiment with as you like), as well as links to our video presentation as slides. This web page will be under development until the official scheduled time of the tutorial (December 7, 1:30pm PT), and may undergo additional changes after that time.

e.g. root-finding is differentiable! When is this a useful inductive bias?

N.B. this also means that gradient descent is feasible for continuous generalizations of search problems in string theory!

SATNet: Bridging deep learning and logical reasoning using a differentiable satisfiability solver

Po-Wei Wang [1] Priya L. Donti [1 2] Bryan Wilder [3] Zico Kolter [1 4]



SATNet: Bridging deep learning and logical reasoning using a differentiable satisfiability solver

Model	Train	Test
ConvNet	72.6%	0.04%
ConvNetMask	91.4%	15.1%
SATNet (ours)	99.8%	**98.3%**

(a) Original Sudoku.

Model	Train	Test
ConvNet	0%	0%
ConvNetMask	0.01%	0%
SATNet (ours)	99.7%	**98.3%**

(b) Permuted Sudoku.

Model	Train	Test
ConvNet	0.31%	0%
ConvNetMask	89%	0.1%
SATNet (ours)	93.6%	**63.2%**

(c) Visual Sudoku. (Note: the theoretical "best" test accuracy for our architecture is 74.7%.)

Hard constraints like Sudoku also present in string theory (tadpole cancellation…)

III. Outlook

- The string landscape presents us with a “big data” problem — we stand to benefit from novel computational approaches.
- Gradient-free optimization via **genetic algorithms** and **reinforcement learning**: not only **identify “interesting recipes,”** but also explore **statistics near optimal states**
  - Interplay between optimization and representation
- String theory presents **novel data** for testing ML approaches and developing new computational insights.

Looking forward

- Identifying the proper ML methods (inductive biases etc.) for studying string-theoretic data remains an open question.
  - Incorporate modular groups like $SL(2,\mathbb{Z})$? Develop symbolic methods? Impose supersymmetry constraints? Implicit layers for constrained optimization?
  - What structures are lurking in the data? Can we come up with better organizing principles for the landscape and string data?