Exploration of string theory landscape

Yuta Hamada (KEK, SOKENDAI)

arXiv:26XX.XXXXX w/ M. Yamazaki (Tokyo), Y. Uematsu (KEK)

2026/01/21 The 2nd ``AI+HEP in East Asia” workshop

String Landscape

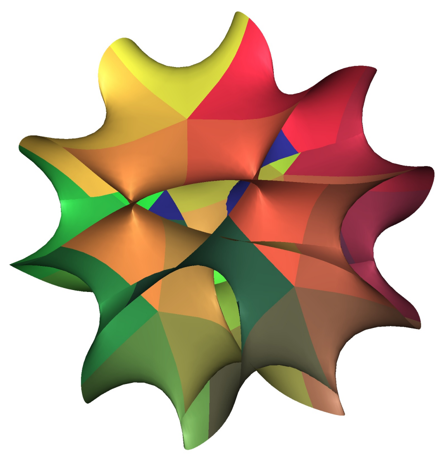
String theory is unique,
but admit many solutions.

Compatification

1+9 dimension — string theory

6d space X,

flux parameter $\vec{m} \in \mathbb{Z}^{\bullet}$.



Compactification

1+3 dimension — field theory

Depending on choice of X and $\vec{m}$, a lot of vacua are obtained though the number is finite [YH, Montero, Vafa, Valenzuela '21].

Choice of flux

There are cycles, where we can insert ``magnetic flux”.

Freedom to choose flux

Quantization number.

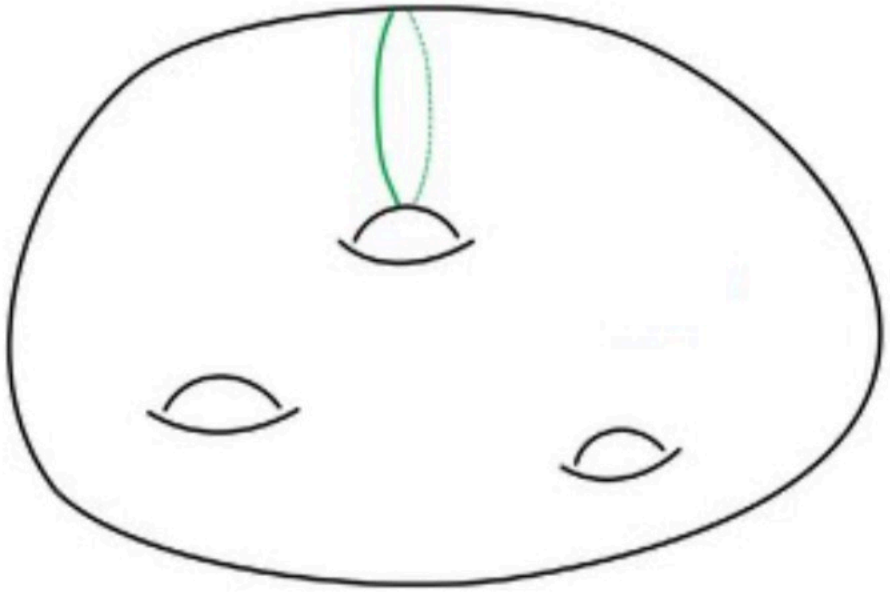
E.g. 10 choice of flux,

and #(cycle)=100,

then we have

$\mathcal{O}(10^{100})$ possibilities

[Douglas ‘03].



(Choice of CYs) $\times$ (Choice of fluxes) $=$ (Choice of models) .

Phenomenologically relevant vacua are extremely rare.

Most vacua have negative cosmological constant, or runaway solution.

So far, we do not find any solution close to our universe.

Complexity

Computational complexity of string vacua search was discussed in 2006.

Computational complexity of the landscape I

Frederik Denef[1,2] and Michael R. Douglas[1,3]

Complexity

Computational complexity of string vacua search was discussed in 2006.

Computational complexity of the landscape I

Frederik Denef[1,2] and Michael R. Douglas[1,3]

In this work, we observe that the class of problems which must be solved in identifying candidate vacua in the string landscape are NP-hard.

Complexity

Computational complexity of string vacua search was discussed in 2006.

Computational complexity of the landscape I

Frederik Denef[1,2] and Michael R. Douglas[1,3]

In this work, we observe that the class of problems which must be solved in identifying candidate vacua in the string landscape are NP-hard. The basic example is the problem of finding a vacuum with cosmological constant of order 10^{-120} in natural units.

Complexity

Computational complexity of string vacua search was discussed in 2006.

Computational complexity of the landscape I

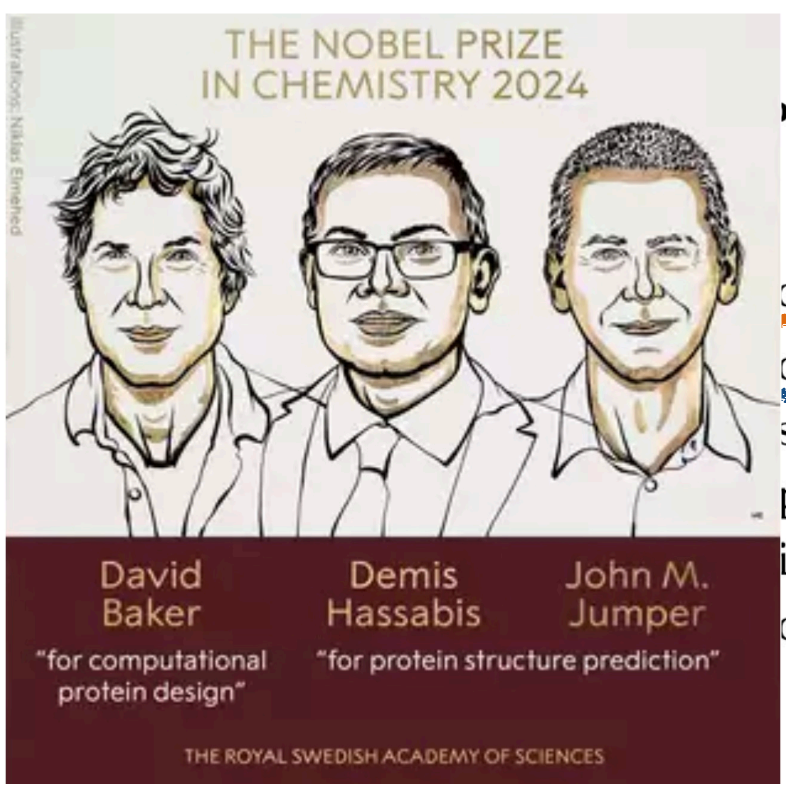
Frederik Denef[1,2] and Michael R. Douglas[1,3]

In this work, we observe that the class of problems which must be solved in identifying candidate vacua in the string landscape are NP-hard. The basic example is the problem of finding a vacuum with cosmological constant of order 10^{-120} in natural units. While in suitable cases it is not hard to argue statistically that such vacua are expected to exist, proving this by finding explicit examples may turn out to be intractable. Indeed the intractability of similar problems is well known in the contexts of the statistical mechanics of spin glasses, in protein folding, and in other fields in which landscapes of high dimension

Complexity

Computational complexity of string vacua search was discussed in 2006.

Computational complexity of the landscape I



uglas[1,3]

In this work, we observe that the class of problems which must be solved in identifying candidate vacua in the string landscape are NP-hard. The basic example is the problem of finding a vacuum with cosmological constant of order 10^{-120} in natural units. While in suitable cases it is not hard to argue statistically that such vacua are expected to exist, proving this by finding explicit examples may turn out to be intractable. Indeed the intractability of similar problems is well known in the contexts of the statistical mechanics of spin glasses, in protein folding, and in other fields in which landscapes of high dimension

Setup

6d space:

‶degree 18 hypersurface in $\mathbb{CP}^4_{1,1,1,6,9}$ with discrete symmetry $\mathbb{Z}_6 \times \mathbb{Z}_{18}$“

From 4d EFT point of view,

we just have potential V for 3 complex scalar fields, (τ, Z_1, Z_2),

and flux parameters $(m_1, m_2, m_3, n_1, n_2, n_3)$.

Potential V

$$V = V\left(\tau, Z_1, Z_2; m_1, m_2, m_3, n_1, n_2, n_3\right).$$

- Solution of string theory is (local) minimum of V.

- V is positive semidefinite.

- SUSY solution: $V = 0$, SUSY solution: $V > 0$.

- Many solution depending on integers $m_{1,2,3}$, $n_{1,2,3}$,

under condition $m_1^2 + m_2^2 + m_3^2 + n_1^2 + n_2^2 + n_3^2 \leq 276$.

Our work [YH, Uematsu, Yamazaki, to appear]:

1: Scan of $\mathcal{O}(10^8)$ vacua.

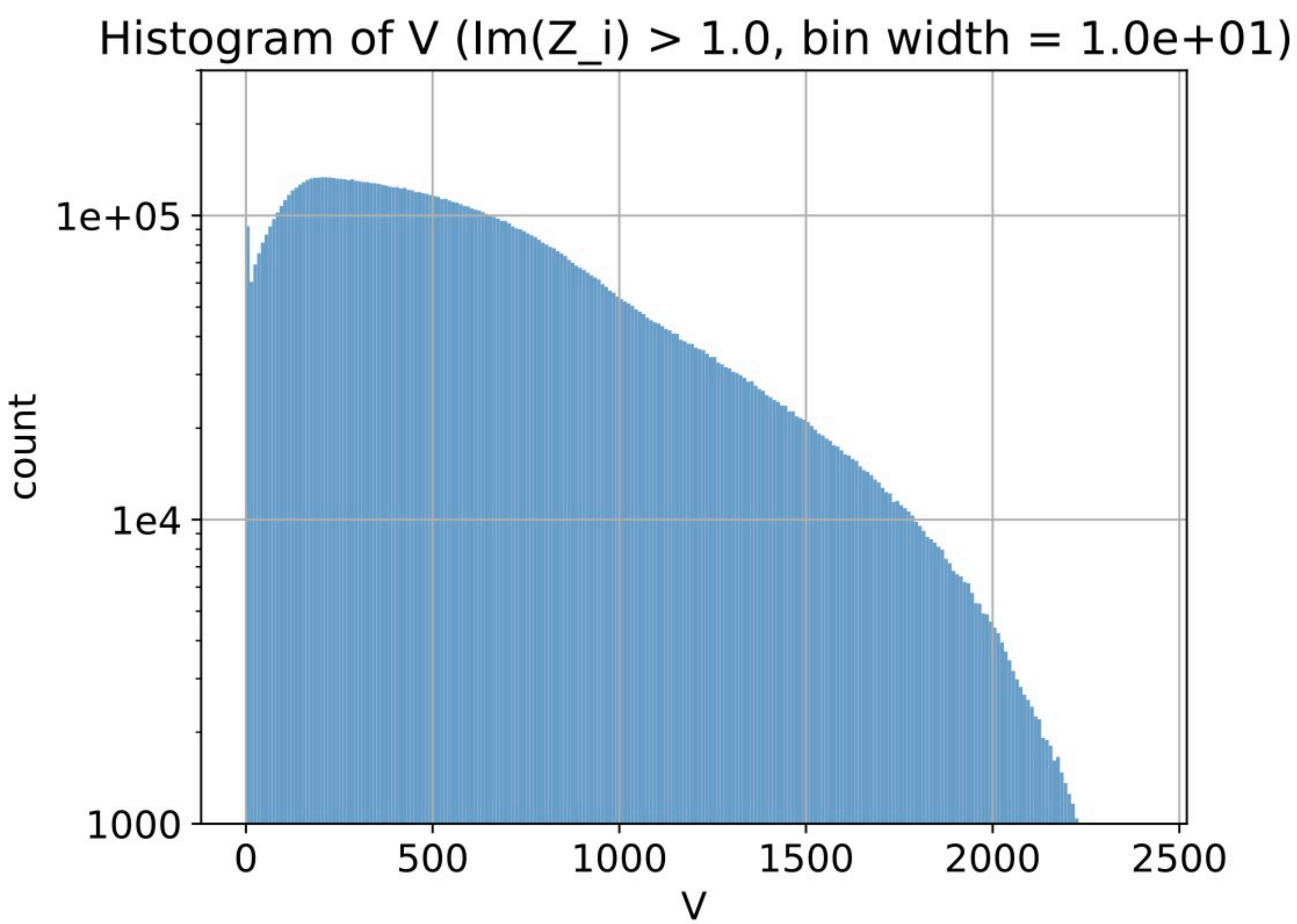
2: Test of effectiveness of machine-learning technique for regression tasks.

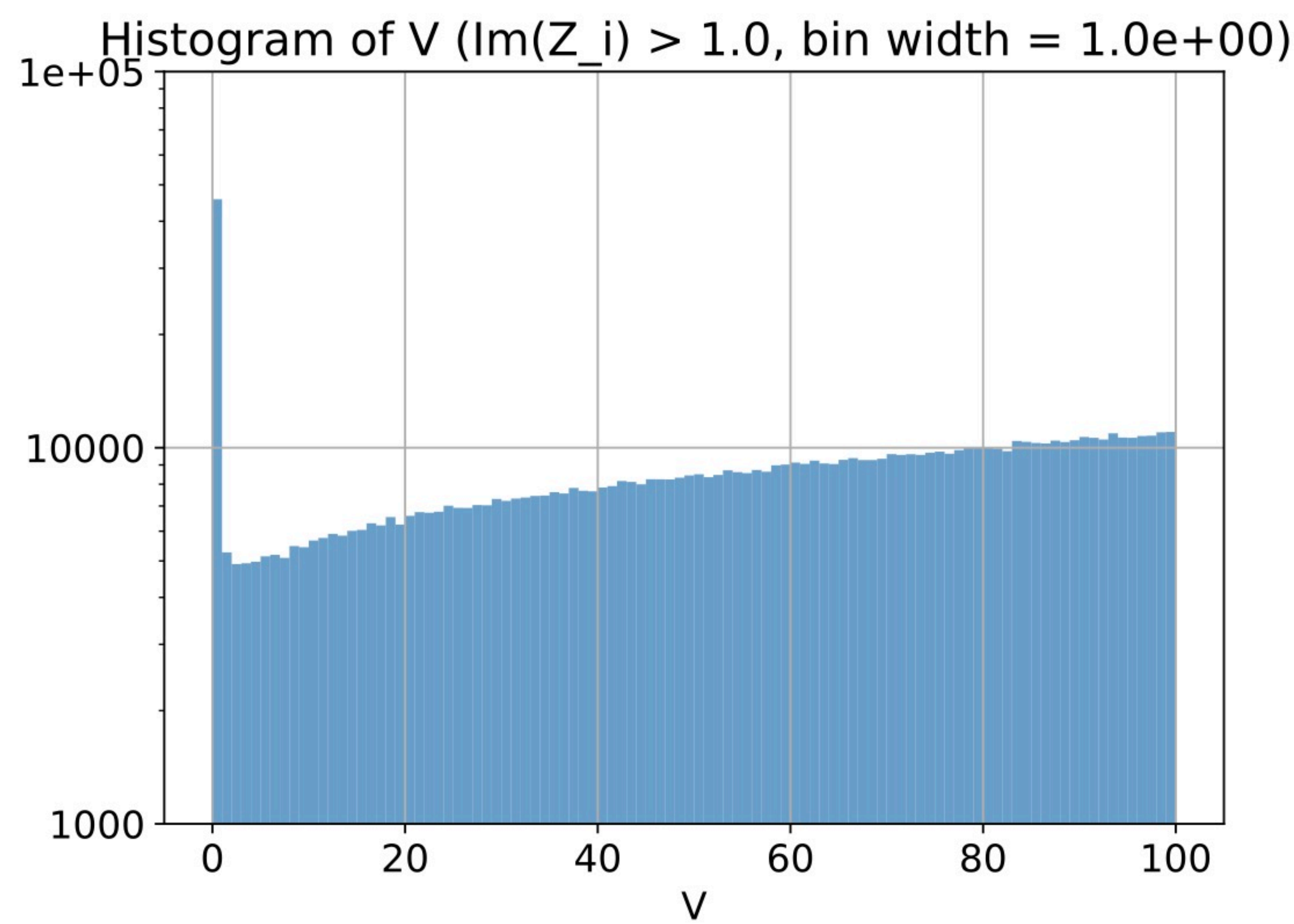
1: Scan of $\mathcal{O}(10^8)$ vacua.

1: We perform minimization by gradient descent method.

	#(non-SUSY vacua)	#(SUSY vacua)
[Martinez-Pedrera+ ’12]		500,865(including unphysical ones)
[Dubey+ ’23]		$\mathcal{O}(30,000)$
[Krippendorf+ '23]	33,619	
[Chauhan ’25]		179,445,394
This work	12,636,619	32,273

Potential Value



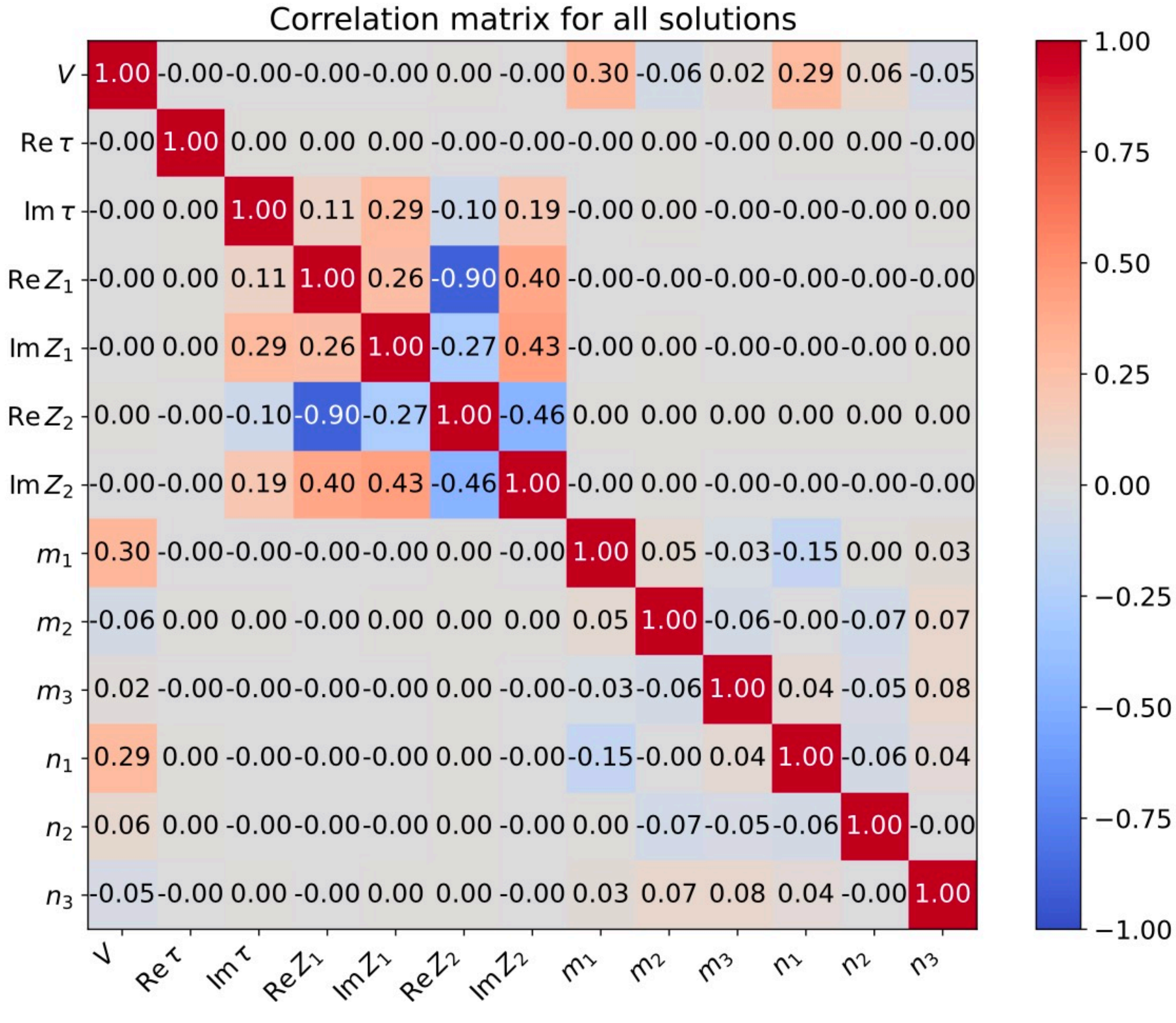


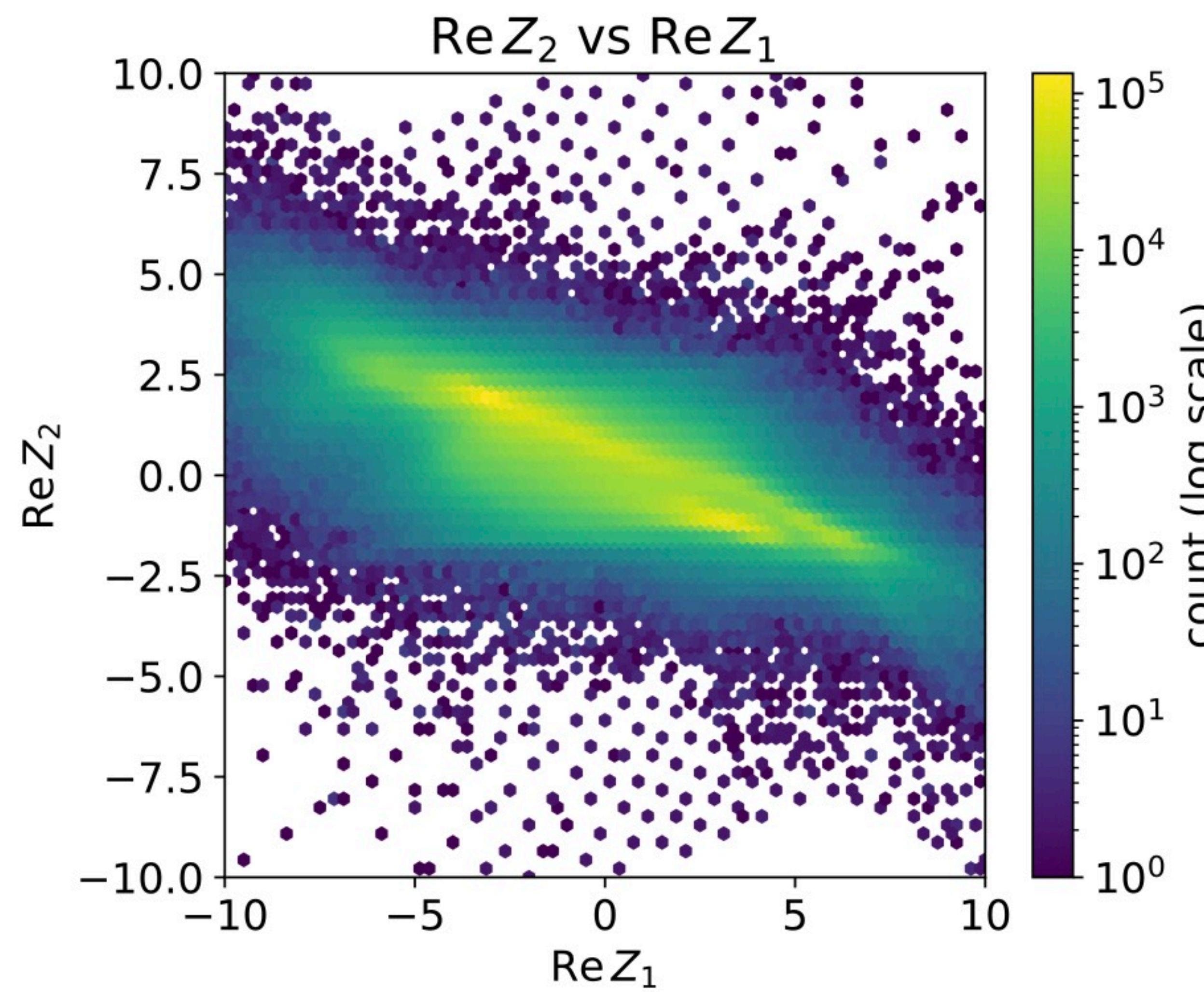
Starting from these, dS vacua could be realized

based on KKLT[Kachru+ '03]/LVS[Balasubramanian+ '05] ($V = 0$ case)

based on [Saltman, Silverstein '04] ($0 < V \ll 1$ case)

Correlations





2: Machine-learning regression task

Input features:

$X = \{V, \tau, Z_1, Z_2, m_i, n_i\}$.

$y \in \{V, \mathrm{Re}\,\tau, \mathrm{Im}\,\tau, \mathrm{Re}\,Z_1, \mathrm{Im}\,Z_1, \mathrm{Re}\,Z_2, \mathrm{Im}\,Z_2, m_i, n_i\}$

We remove y from X and train a regression model to predict y from the remaining features in X.

A non-linear map $f : X \backslash \{y\} \mapsto y$ is a target for learning.

Test of effectiveness

Data set is randomly split into training set (80%) + test set (20%).

We compute coefficient of determination R^2 to assess quality of fit.

true target values $\{y_i\}_{i=1}^n$ v.s. model predictions $\{\hat{y}_i\}_{i=1}^n$.

$$R^2 := 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2},$$

mean value $\bar{y} = \frac{1}{n}\sum_{i=1}^n y_i$

Target	R^2_{train}	R^2_{test}
V	0.77	0.77
m_1	0.88	0.88
m_2	0.55	0.55
m_3	0.61	0.61
n_1	0.85	0.85
n_2	0.55	0.55
n_3	0.62	0.62

$R^2 = 1$: Perfect fit.

$R^2 < 0$: Worse than baseline ($\hat{y}_i = \bar{y}$ for all i).

Summary

Search of flux vacua in string theory.

- $\mathcal{O}(10^8)$ solutions are generated by gradient descent.
- Statistics of solutions are analyzed, and candidates for dS universe are identified.
- Machine-learning for regression tasks.