Deconstructing the Landscape 2023

AI4Research

Vetenskapsrådet

Machine learning Calabi-Yau metrics

Magdalena Larfors, Uppsala University

Work in collaboration with
Andre Lukas, Fabian Ruehle, Robin Schneider, Yacoub Hendi, Moritz Wange
arxiv:2111.01436 & 2205.13408 and work in progress

…or
Simple neural nets learn tricky geometry

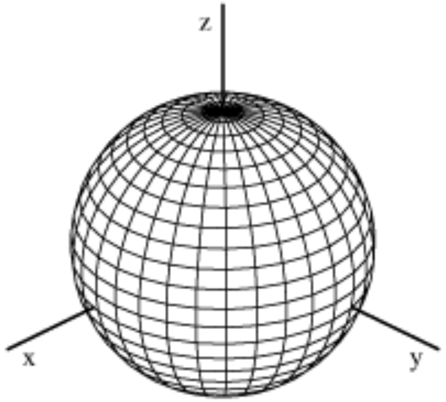
Motivation & problem set-up

Swampland, compactifications and Calabi-Yaus

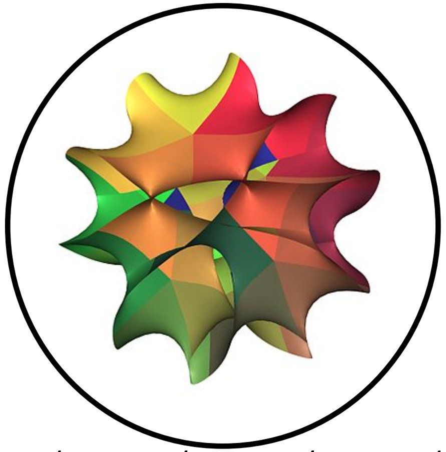
- String compactifications: test Swampland conjectures
- The topology and geometry of compact dimensions is key
  - e.g. Swampland Distance Conjecture
- Calabi-Yau manifolds are popular:
  - Give SUSY Minkowski vacua (with moduli), …
  - Admit Ricci-flat metric (but non-construtctive proof)
  - Many examples, topology well understood.
- *Ricci-flat CY metric* has info on geodesics, curvature, masses,…
  Can we compute it in examples?

Calabi-Yau manifolds: algebraic construction

- Build non-trivial spaces from from simple ambient spaces



$x^2 + y^2 + z^2 = 1$ in $\mathbb{R}^3$



$Z_0^5 + Z_1^5 + Z_2^5 + Z_3^5 + Z_4^5 = 0$ in $\mathbb{P}^4$

- Many examples collected in databases:
  CICY *Candelas et al:88*, hypersurfaces in toric spaces *Kreuzer-Skarke:00*, …

CY manifolds and Ricci flat metrics

Calabi:54, Yau:78

- Let X be an n-dimensional compact, complex, Kähler manifold with vanishing first Chern class.
  Then in any Kähler class $[J]$, X admits a unique Ricci flat metric g_{CY}.

- There is *no analytical expression* for g_{CY}.
- Impose Ricci-flatness: solve non-linear PDE for metric. This is hard.

Ricci-flat CY metrics

Calabi:54, Yau:78

- Let X be an n-dimensional compact, complex, Kähler manifold with vanishing first Chern class.
  Then in any Kähler class $[J]$, X admits a unique Ricci flat metric g_{CY}.
- There is *no analytical expression* for g_{CY}.

But on CY spaces, we know more! Kähler form J_{CY} satisfies

- $J_{CY} = J + \partial\bar{\partial}\phi$ — same Kähler class; ϕ is a function
- $J_{CY} \wedge J_{CY} \wedge J_{CY} = \kappa\, \Omega \wedge \bar{\Omega}$ — Monge-Ampere equation (κ constant)
  2nd order PDE for ϕ

Ricci-flat CY metrics

Calabi:54, Yau:78

- Let X be an n-dimensional compact, complex, Kähler manifold with vanishing first Chern class.
  Then in any Kähler class $[J]$, X admits a unique Ricci flat metric g_{CY}.
- There is *no analytical expression* for g_{CY}.

We can compute these in examples!

Kähler form J_{CY} satisfies

- $J_{CY} = J + \partial\bar{\partial}\phi$ same Kähler class; ϕ is a function
- $J_{CY} \wedge J_{CY} \wedge J_{CY} = \kappa\, \Omega \wedge \bar{\Omega}$ Monge-Ampere equation (κ constant)
  2nd order PDE for ϕ

Setting up the problem:

Find Ricci flat CY metric $g_{CY} \iff$ find J_{CY} that solves MA equation

$$J_{CY} \wedge J_{CY} \wedge J_{CY} = \kappa\, \Omega \wedge \bar{\Omega}$$

where κ is some complex constant.

Numerical method: Sample large set of random points on CY.

- Compute Ω and a reference J at all points
- Solve MA eq. numerically for J_{CY} (or ϕ)
- Check solution (on new points): Does MA eq hold? Is Ricci tensor 0?

Numerical CY metrics – a longstanding quest

- Donaldson algorithm

  Donaldson:05, Douglas-et.al:06, Douglas-et.al:08, Braun-et.al:08, Anderson-et.al:10, ...,

- Energy functionals

  Headrick–Nassar:13, Cui–Gray:20, Ashmore–Calmon–He–Ovrut:21, ...

- Machine learning

  Ashmore–He–Ovrut:19, Douglas–Lakshminarasimhan–Qi:20,
  Anderson–Gerdes–Gray–Krippendorf–Raghuram–Ruehle:20, Jejjala–Mayorga–Peña:20 ,
  Larfors-Lukas-Ruehle-Schneider:21, 22, Ashmore–Calmon–He–Ovrut:21,
  Berglund–Butbaia–Hübsch–Jejjala–Mayorga Peña–Mishra–Tan:22,
  Gerdes–Krippendorf:22...

Numerical CY metrics – a longstanding quest

- Donaldson algorithm

  Donaldson:05, Douglas-et.al:06, Douglas-et.al:08, Braun-et.al:08, Anderson-et.al:10, ...,

- Energy functionals

  Headrick–Nassar:13, Cui–Gray:20, Ashmore–Calmon–He–Ovrut:21, ...

- Machine learning

  Ashmore–He

  Anderson–G *orga–Peña:20 ,*

  Larfors-Luka

  Berglund–B

  Gerdes–Krip

https://github.com/yidiq7/MLGeometry

https://github.com/pythoncymetric/cymetric

https://github.com/ml4physics/cyjax

Numerical CY metrics – a longstanding quest

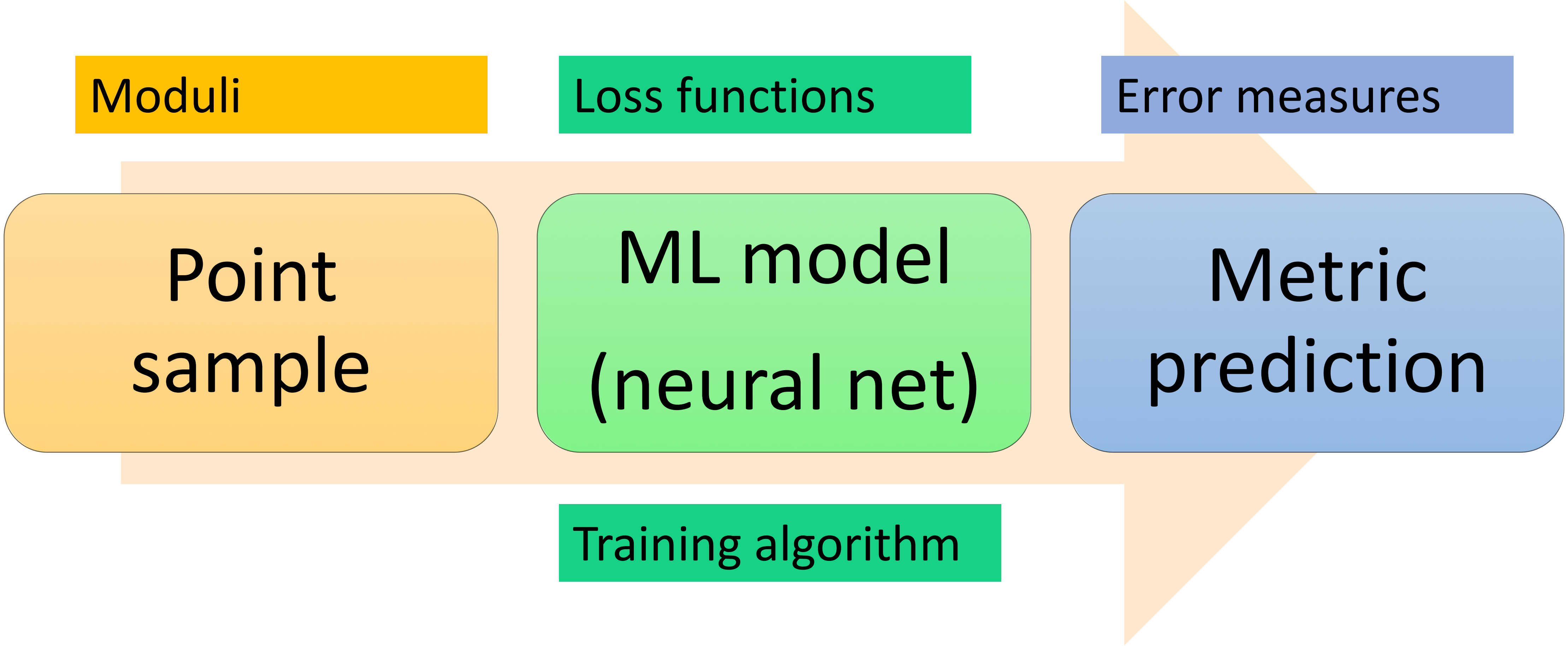
<u>Algebraic CY metrics</u>

- Expand K_{CY} in basis of hom. polynomials of degree k
- $K_k(z,\bar{z}) = \frac{1}{k}\sum \ln H_{a\bar{b}} p^a \bar{p}^{\bar{b}}$
- Solve for $H_{a\bar{b}}$ using
  - Donaldson algorithm
  - Minimize energy functional
  - Machine learning

<u>Machine Learning CY metrics</u>

- NNs are universal approximators
  Cybenko:89; Hornik:91; Leshno et.al:93; Pinkus:99
- ML model searches freely for CY metric
- Training objective: minimize loss
- Control evolution via NN architecture and loss functions

Machine Learning implementation



Example implementation: cymetric package

`cymetric`

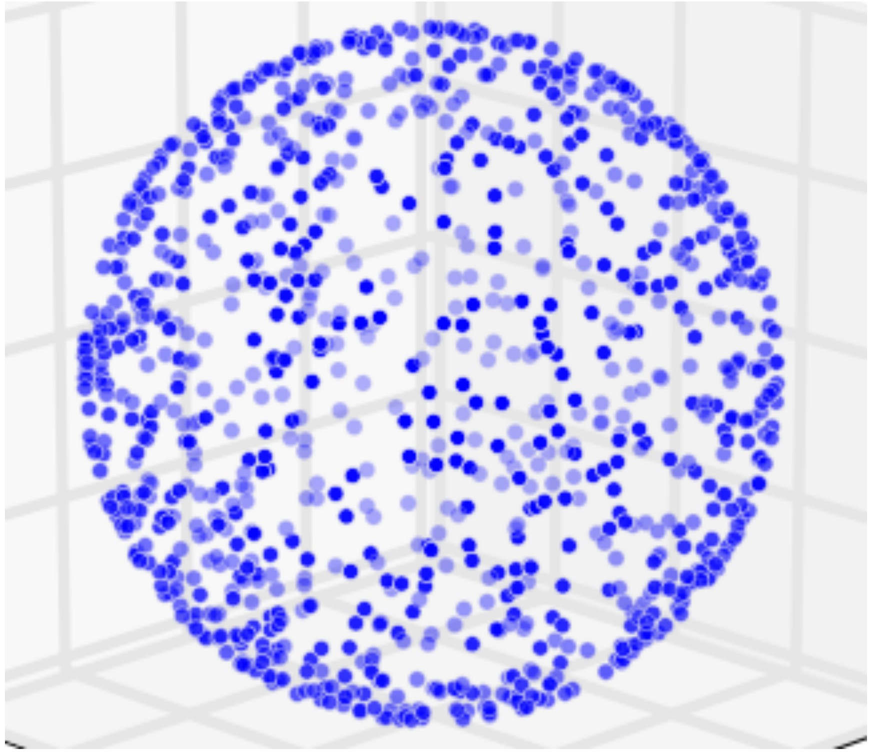
ML package `cymetric` written in Python and Mathematica (separately).

Decomposes into

1. point generators based on Shiffman–Zelditch theorem
   Uses NumPy, SciPy, SageMath and Mathematica.

2. custom neural networks give CY metric (at given point in moduli space)
   - 5 different models (metric Ansätze)
   - Implemented and optimized with TensorFlow/Keras

Point generators



What do we need? Error measures

After training, evaluate performance (on separate test set):

does the MA equation hold? is the metric Ricci flat?

Check via established benchmarks:

$$\sigma = \frac{1}{\mathrm{Vol}_{\mathsf{CY}}} \int_X \left| 1 - \kappa \, \frac{\Omega \wedge \overline{\Omega}}{(J_{\mathrm{pr}})^3} \right| \, , \; \mathcal{R} = \frac{1}{\mathrm{Vol}_{\mathsf{CY}}} \int_X |R_{\mathrm{pr}}| \, .$$

using Monte Carlo integration for any function f

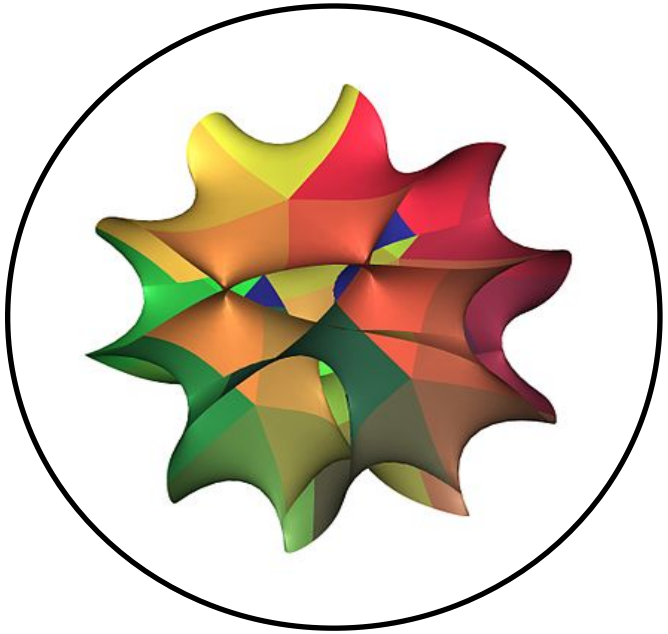
$$\int_X \mathrm{dVol}_{\mathrm{CY}} f = \int_X \frac{\mathrm{dVol}_{\mathrm{CY}}}{\mathrm{d}A} \mathrm{d}A \, f = \frac{1}{N} \sum_i w_i f|_{p_i} \quad \text{with} \quad w_i = \frac{\mathrm{dVol}_{\mathrm{CY}}}{\mathrm{d}A}|_{p_i}$$

Point generators

We need

- random set of points on CY
- sampled w.r.t. measure dA

..so we can compute integrals
(e.g to check accuracy)

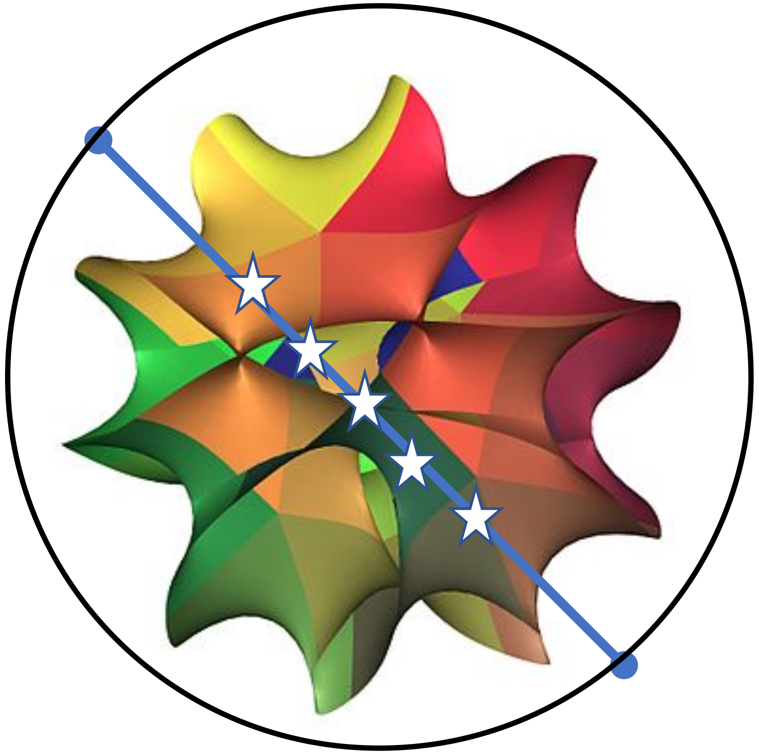


$$\int_X \mathrm{dVol}_{\mathrm{CY}} f = \int_X \frac{\mathrm{dVol}_{\mathrm{CY}}}{\mathrm{d}A} \mathrm{d}A\, f = \frac{1}{N} \sum_i w_i f|_{p_i} \quad \text{with} \quad w_i = \frac{\mathrm{dVol}_{\mathrm{CY}}}{\mathrm{d}A}|_{p_i}$$

Point generators

Quintic $X: p = 0 \subset \mathbb{P}^4$ *Douglas et. al: 06*

- Sample 2 points on $\mathbb{P}^4$; connect & intersect
- Repeat M times $\rightsquigarrow 5M$ random points on X
- Shiffman-Zelditch:
  points distributed w.r.t. FS measure on X



- Generalizations:
  CICY *Douglas et.al: 07*
  Kreuzer-Skarke *ML, Lukas, Ruehle, Schneider: 21,22*

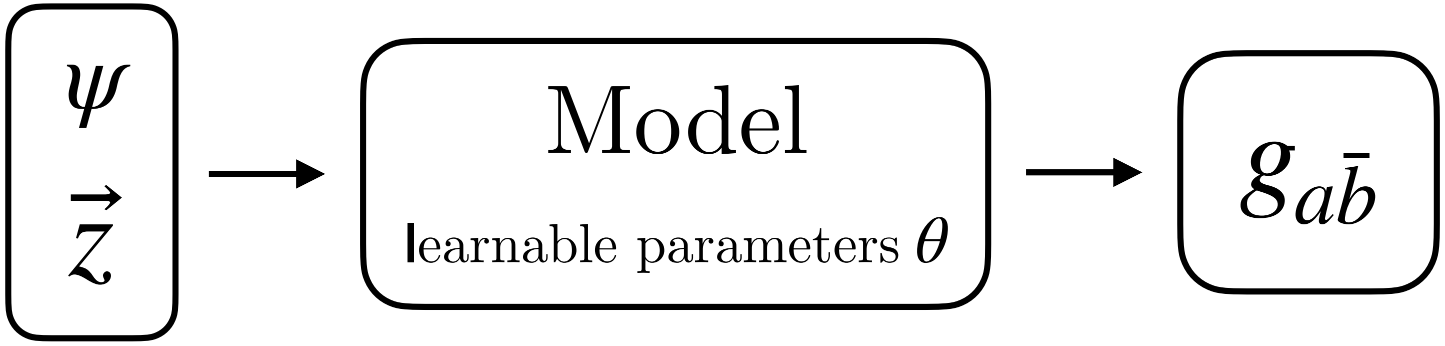
Point generators for KS CY manifolds

- Can we relate ambient toric variety A to projective spaces?
  Yes!
  Use sections of line bundle dual to Kähler cone divisors;
  recall nef divisors are base-point free
- So Shiffman–Zelditch applies and quintic algorithm generalizes.

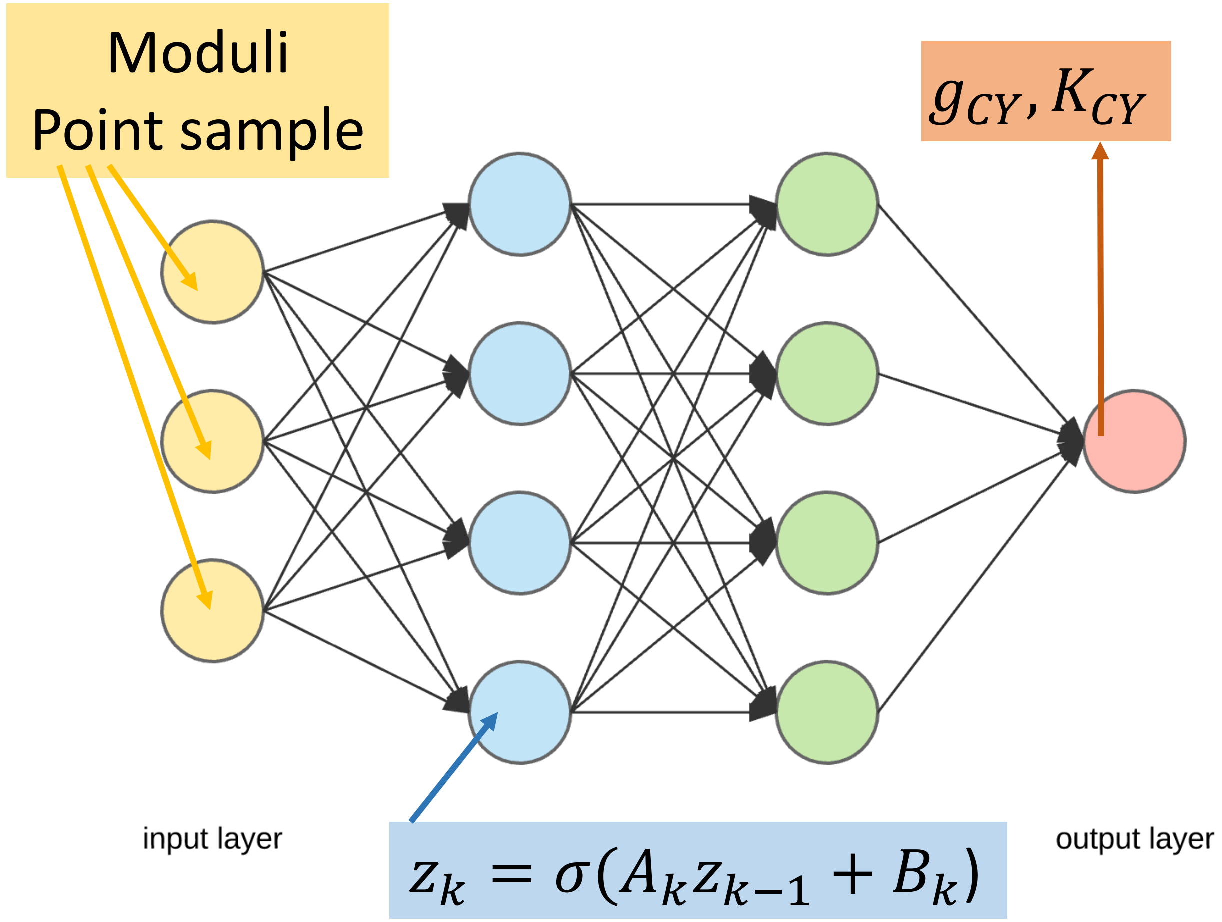
- Sections $s_j^{(\alpha)}$ of the toric Kähler cone generators $J_\alpha \sim$ coordinates of $\mathbb{P}^{r_\alpha}$
- Use Shiffman–Zelditch on $\mathbb{P}^{r_\alpha}$
- Express CY 3-fold as non-complete intersection in $\hat{A} \cong \bigotimes_{\alpha=1}^{h^{1,1}} \mathbb{P}^{r_\alpha}$
- Intersect $\rightsquigarrow$ sample of random points on CY distributed wrt FS measure.

Details: toric-point-gen

ML model architecture in cymetric



ML models: Set-up and training



Architectural choices

- What to predict?
- Encode constraints in NN or loss? (global, complex, Kähler…)

Then train

- Minimize loss functions
- Choose optimizer

And check performance

ML models in cymetric package

- 5 ML models
- Encode few/many constraints so need more/less loss functions

Model name	**Ansatz**
Free	$g_{\text{pr}} = g_{\text{NN}}$
Additive	$g_{\text{pr}} = g_{\text{FS}} + g_{\text{NN}}$
Multiplicative, elementwise	$g_{\text{pr}} = g_{\text{FS}} + g_{\text{FS}} \odot g_{\text{NN}}$
Multiplicative, matrix	$g_{\text{pr}} = g_{\text{FS}} \cdot (\mathbb{I} + g_{\text{NN}})$
ϕ-model	$g_{\text{pr}} = g_{\text{FS}} + \partial\bar{\partial}\phi$

Loss functions encode math constraints

- Train the network to get *unknown* Ricci-flat metric (in given Kähler class)
- Use semi-supervised learning
  1. Encode mathematical constraints as custom loss functions
  2. Train network (adapt layer weights) to minimize loss functions
- E.g. satisfy Monge-Ampere eq $\rightsquigarrow$ minimize Monge-Ampere loss

$$\mathcal{L}_{\mathsf{MA}} = \left|\left| 1 - \frac{1}{\kappa} \frac{\det g_{\mathsf{pr}}}{\Omega \wedge \bar{\Omega}} \right|\right|_n$$

- Depending on metric ansatz, need more or fewer loss functions.

Custom loss functions in cymetric package

- Monge-Ampere loss

$$\mathcal{L}_{\text{MA}} = \left|\left| 1 - \frac{1}{\kappa} \frac{\det g_{\text{pr}}}{\Omega \wedge \bar{\Omega}} \right|\right|_n$$

- Ricci loss (duplicates MA loss)
- Kähler loss (not needed for ϕ network)
- Transition loss
- Kähler class loss (needed when $h^{(1,1)} > 1$)

Error functions and accuracy measures

- After training, check that MA eq holds and Ricci tensor is zero

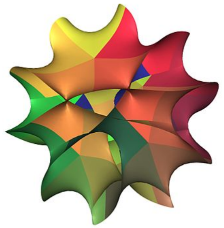
Check via established benchmarks:

$$\sigma = \frac{1}{\mathrm{Vol}_{\mathsf{CY}}} \int_X \left| 1 - \kappa \, \frac{\Omega \wedge \overline{\Omega}}{(J_{\mathsf{pr}})^3} \right| \, , \; \mathcal{R} = \frac{1}{\mathrm{Vol}_{\mathsf{CY}}} \int_X |R_{\mathsf{pr}}| \, .$$

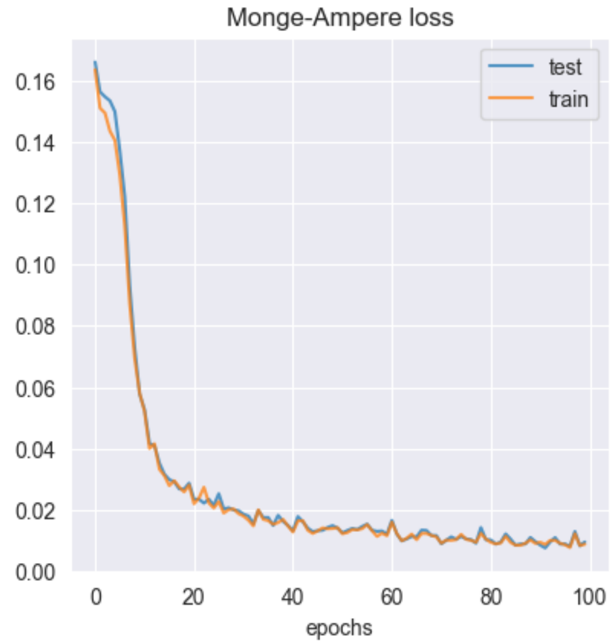
- For CY manifolds with more than one Kähler class, must also check that we keep this fixed:
- So we check that volume and line bundle slopes remain constant.

Experiments

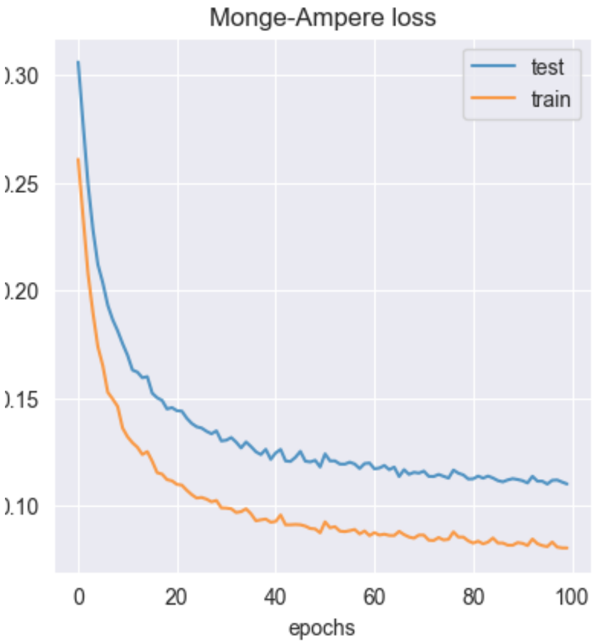
Fermat vs. generic quintic



Monge-Ampere loss

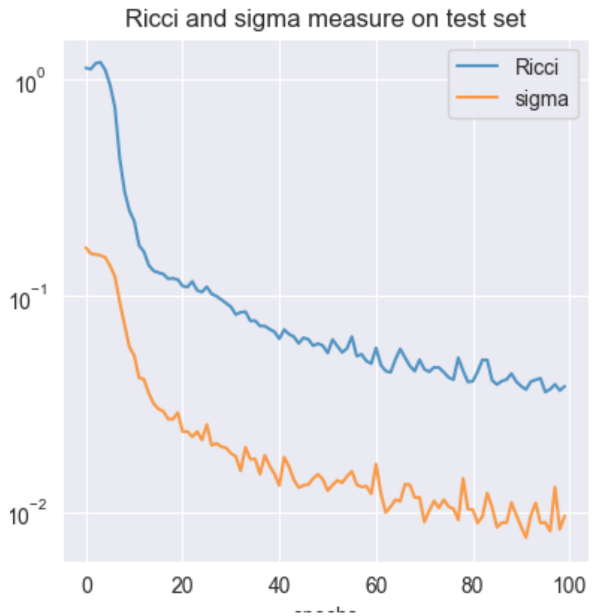


Fermat

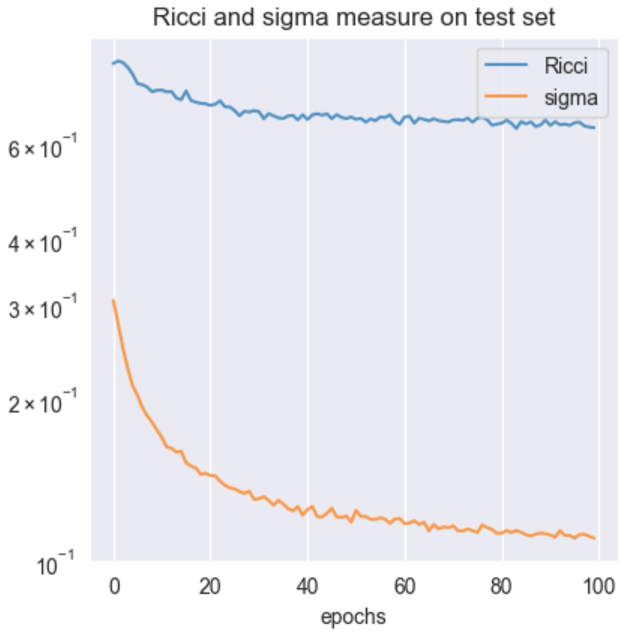


Generic

Error measures



Fermat



Generic

100000 points, ϕ model, 3 64-node layers, GELU, default loss, Adam, batch (64, 50000)

KS CY example

$h^{1,1} = 2, h^{2,1} = 80$ CY from the Kreuzer-Skarke list

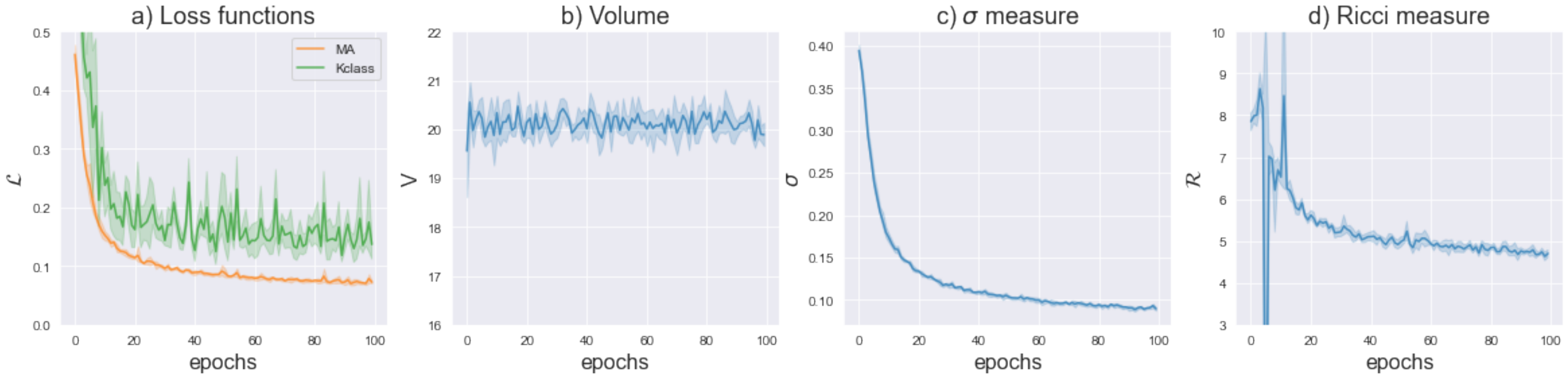
- Ambient space is $\mathbb{P}^1 \to \mathrm{A} \to \mathbb{P}^3$ w. toric coordinates $(x_0, \ldots, x_4)$

- CY hypersurface: $p(x_0, \ldots, x_4) = 0$ (80 terms; select randomly)
- 2 Kähler cone generators J_a ; $J = t_1 J_1 + t_2 J_2$
- Morphisms to $\mathbb{P}^1$ and $\mathbb{P}^5$ using $H^0(J_a)$

- Point generation ~ 1 hour (generic cpl structure moduli, $t_a = 1$).

KS CY example

- $h^{1,1} = 2$, $h^{2,1} = 80$ Kreuzer-Skarke



- Toric ϕ-model, default loss, 200 000 points
- NN width 256, depth 3, GELU, batch (128, 10000), SGD w. momentum

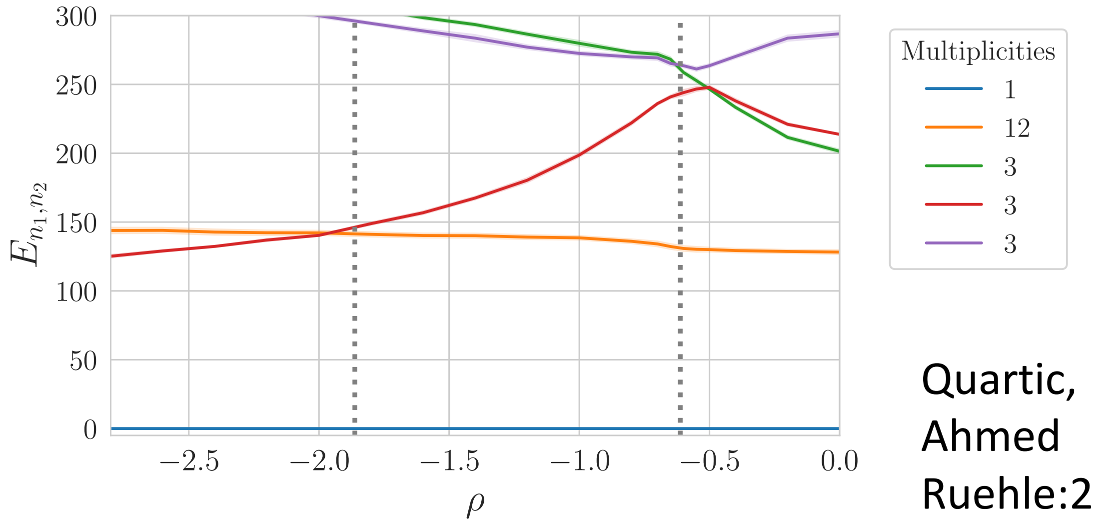
Further developments

Connecting to Swampland

Ashmore:20, Ashmore & Ruehle:21 Ahmed & Ruehle:23

- Compute *moduli-dependent* spectrum of Δ_{CY} *explicitly* in example CY:s

1. Compute the moduli space metric (using either analytic [20] or numeric [21] techniques)
2. Compute the geodesics and the geodesic distances in moduli space
3. Compute the CY metric along the moduli space geodesics
4. Compute the massive spectrum from the CY metric
5. Fit a function to the masses and compare with the prediction from the SDC



Quartic, Ahmed Ruehle:23

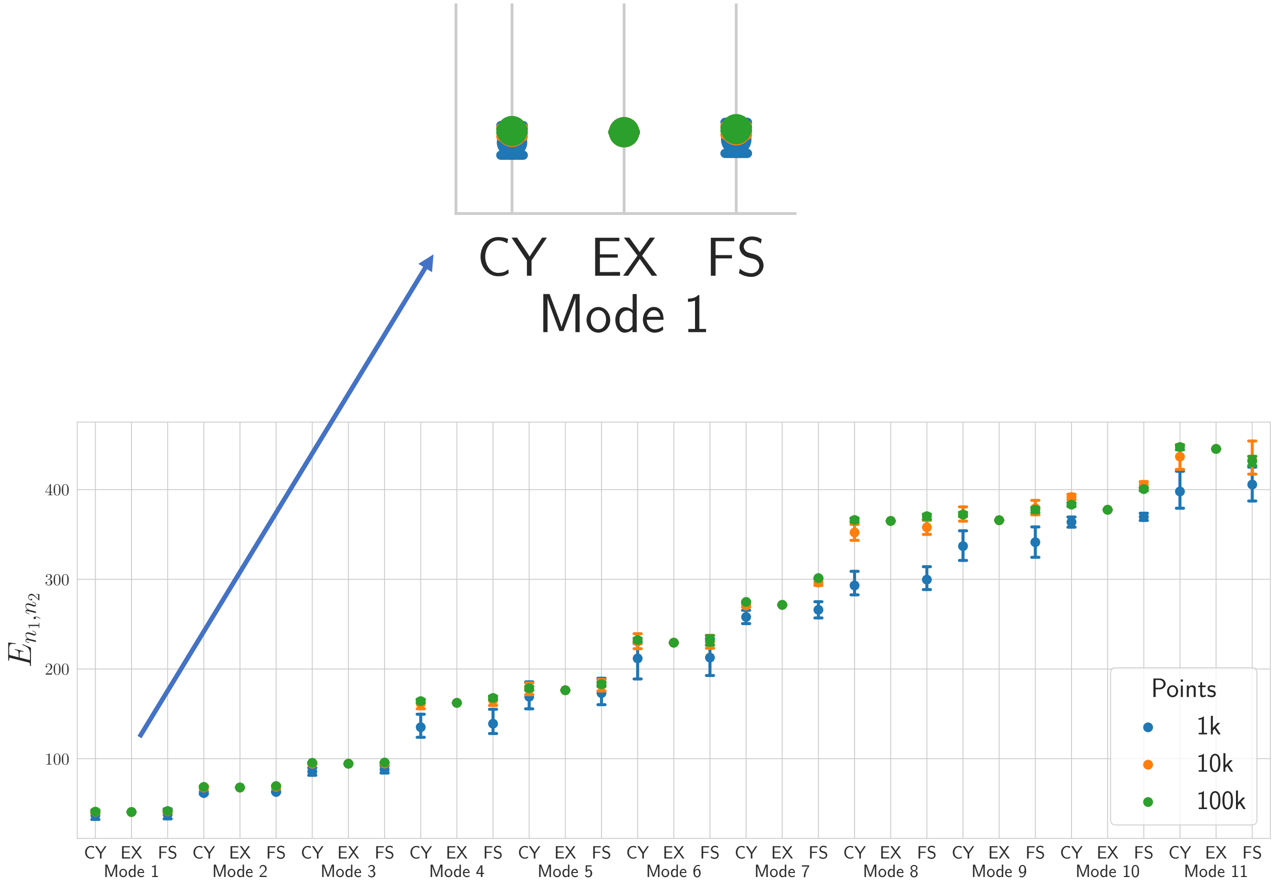
- Level crossing & number theory

Accuracy and benchmarks

Ahmed & Ruehle:23

Improve accuracy

- Larger point sample
- Wider/deeper NN
- Train longer



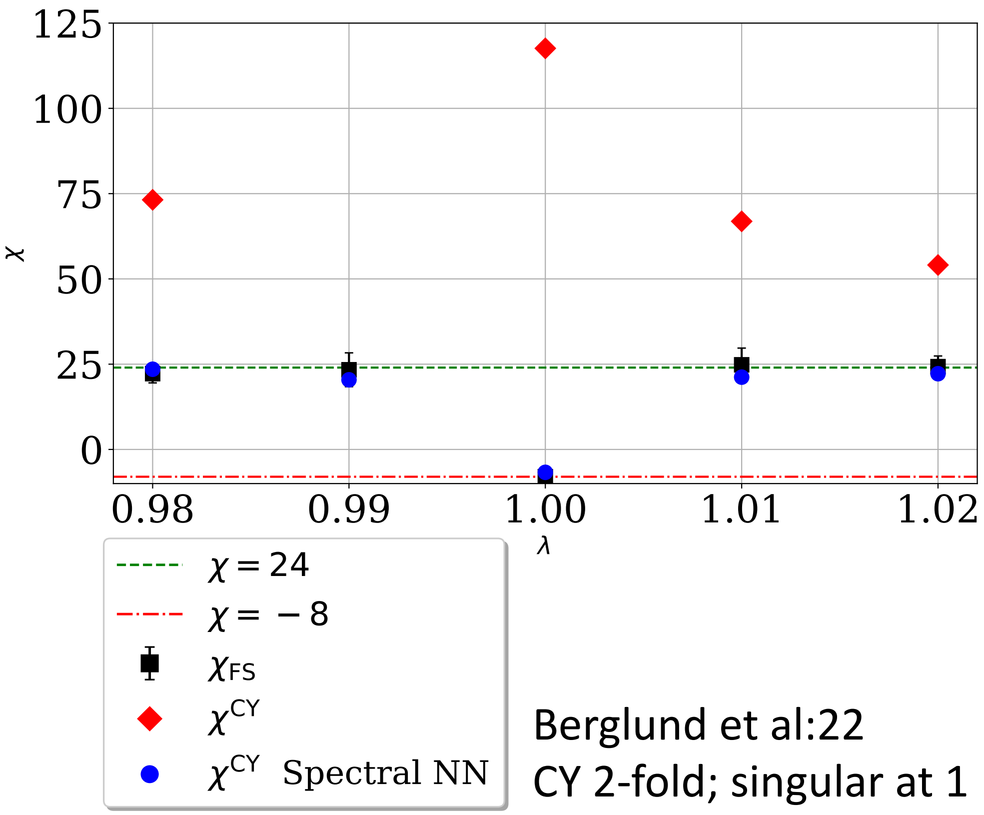
- Benchmark on cubic CY in $\mathbb{P}^2$ (a.k.a. the torus)
- Spectrum of Δ_{CY}

Accuracy, performance and architecture

- Is the control by loss functions accurate enough?
- Can performance be improved?

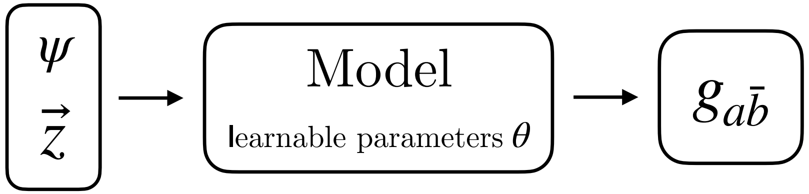
Change ML architecture

- Algebraic metric Ansatz
  Anderson et al : 20, Douglas et al : 20, Gerdes & Krippendorf:22, ...
- cymetric with spectral layer
  Berglund et al:22
- Metric Neural Tangent Kernel
  Halverson & Ruehle:23
- Symmetries and Geometric Deep Learning
  in progress



Berglund et al:22
CY 2-fold; singular at 1

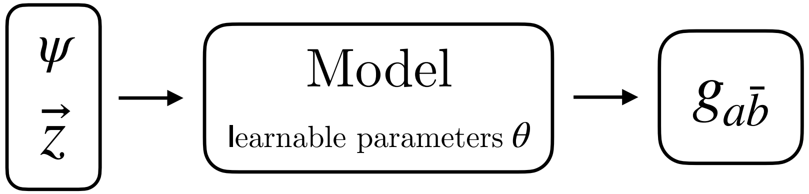
Conclusion and outlook



- Simple NNs can learn Ricci flat CY metrics
- Mathematical constraints: encoded in NN or in loss functions
- cymetric package: applies to all CICY and Kreuzer-Skarke list at given point in moduli space
- Architecture and accuracy, performance, generality,…

- Moduli-dependent CY metrics
- Applications in physics:
  massive modes, swampland conjectures, Yukawa couplings, wrapped branes, ...
- Go beyond CY: G2 metrics, G-structure manifolds, ...

Conclusion and outlook



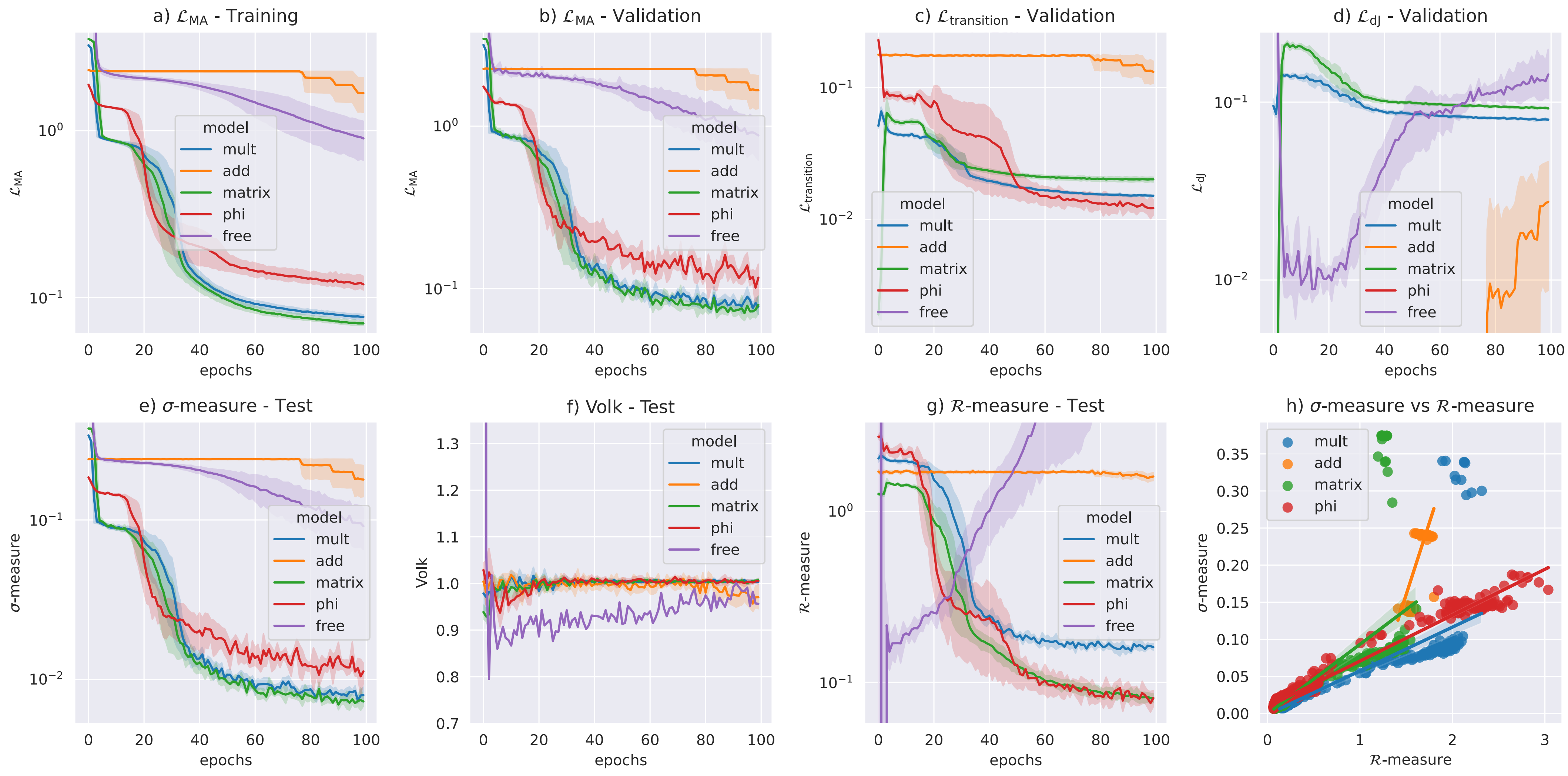
- Simple NNs can learn Ricci flat CY metrics
- Mathematical constraints: encoded in NN or in loss functions
- cymetric package: applies to all CICY and Kreuzer-Skarke list at given point in moduli space
- Architecture and accuracy, performance, generality,...

- Moduli-dependent CY metrics
- Applications in physics: massive modes, swampland conjectures, Yukawa couplings, wrapped branes, ...
- Go beyond CY: G2 metrics, G-structure manifolds, ...

Thank you for listening!

Additional slides

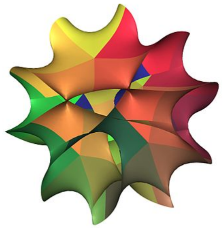
Different metric ansatze on Fermat quntic



198 000 points, 0.1 val split, 5 experiments/model.

NN width 64, depth 3, GELU, batch size 64, Adam optimizer. Ricci and K-class loss disabled.

Calabi-Yau spaces: details



- Complex: local coordinates $z_i, \bar{z}_{\bar{j}}$
  holomorphic top form $\Omega = dz_i \wedge dz_j \wedge dz_k$
- Kähler: metric determined by Kähler potential

$$g_{i\bar{j}} = \partial_i \partial_{\bar{j}} K, \quad g_{ij} = g_{\bar{i}\bar{j}} = 0$$

Kähler form $J = \frac{i}{2} \sum g_{i\bar{k}} dz^j \wedge d\bar{z}^{\bar{k}}$

- Come in families parametrized by *complex structure/Kähler moduli*

- Satisfy topological restriction ($c_1 = 0$)
  → admit *a unique Ricci-flat CY metric*

Traditional methods

- Approximate K_{CY} via *algebraic expansion* in polynomial basis

$$K_k(z, \bar{z}) = \frac{1}{k} \sum \ln H_{a\bar{b}} p^a \bar{p}^{\bar{b}}$$

- Hermitian matrix H to be computed

<u>Donaldson algorithm</u>

- H_k : fixed point of iteration scheme
- Slow convergence at given k
- Proven $K \rightarrow K_{CY}$ as $k \rightarrow \infty$

<u>Energy functional</u>

- H_k : minimum of functional encoding MA equation
- Fast convergence at given k

Traditional methods – scaling problem

- Approximate K_{CY} via *algebraic expansion* in polynomial basis

$$K_k(z,\bar{z}) = \frac{1}{k}\sum \ln H_{a\bar{b}} p^a \bar{p}^{\bar{b}}$$

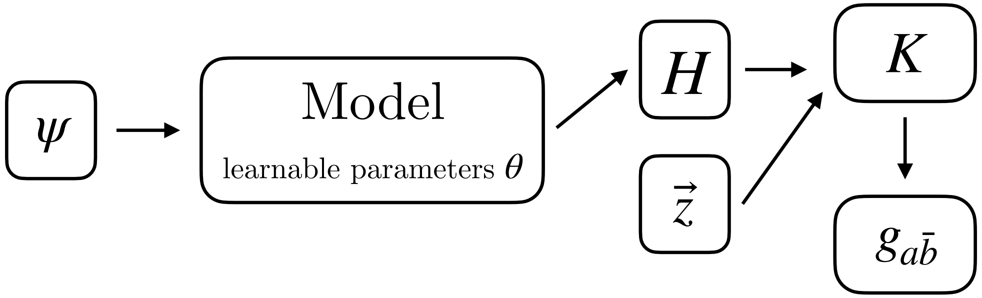
- Hermitian matrix H to be computed
- Problem: polynomial basis dim N_k grows with k, and $H \sim N_k^2$
  On quintic:

$$N_k = \binom{k+4}{k} - \binom{k-1}{k-5} = 5, 15, 35, 70, 125, 205, 315, \ldots$$

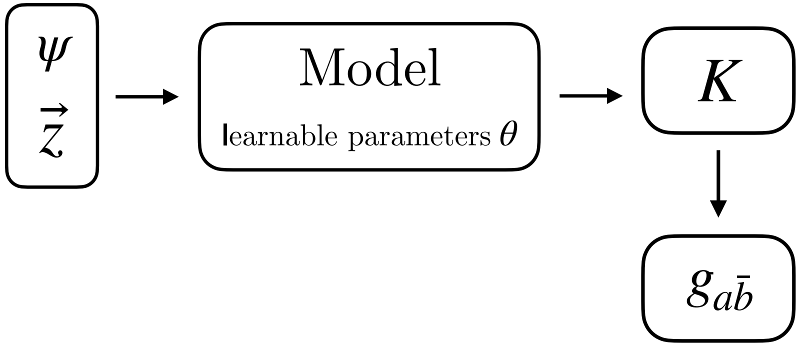
- Use discrete symmetries to cut down N_k. Restriction on moduli.

ML model architectures

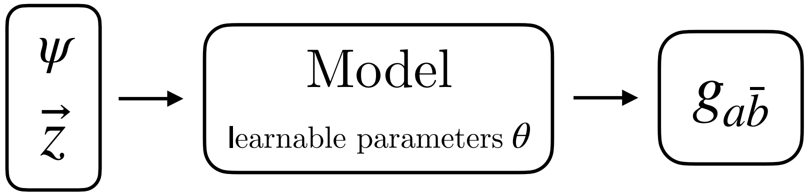
1. Learn Donaldson's H matrix
   Anderson et al 2012.04656, Gerdes et al 2211.12520



2. Learn Kähler potential
   Anderson et al 2012.04656, Douglas et al 2012.04797, Larfors et al 2111.01436 & 2205.13408 , Berglund et al 2211.09801



3. Learn metric
   Anderson et al 2012.04656, Jejjala et al 2012.15821, Larfors et al 2111.01436 & 2205.13408



Point Generators

Creating a point sample on KS CY 3-fold, part 1

Assume toric ambient space w. coordinates $x_i \sim D_i$ divisors

- Kähler cone generators $J_\alpha = \sum c^i_\alpha D_i$ dual to nef line bundle $\mathcal{O}(J_\alpha)$
- Sections of $\mathcal{O}(J_\alpha) \sim$ coordinates of $\mathbb{P}^{r_\alpha}$

$$\Phi_\alpha : \quad [x_0 : x_1 : \ldots] \quad \to \quad [s_0^{(\alpha)} : s_1^{(\alpha)} : \ldots : s_{r_\alpha}^{(\alpha)}]$$

- FS metrics on $\mathbb{P}^r \longrightarrow$ (non-FS) Kähler metric on $\mathcal{A}$.
- Build random sections

$$S = \sum_{j=0}^{r_\alpha} a_j^{(\alpha)} s_j^\alpha$$

  drawing $a_j^{(\alpha)}$ independently from Gaussian distribution

- **Theorem**[Shiffman and Zelditch]:
  Zeros of random sections of are distributed according to the FS measure.

Point Generators

Creating a point sample on KS CY 3-fold, part 2

Got map $\Phi_\alpha : \quad [x_0 : x_1 : \ldots] \quad \rightarrow \quad [s_0^{(\alpha)} : s_1^{(\alpha)} : \ldots : s_{r_\alpha}^{(\alpha)}]$

Know zeros of random sections $S = \sum_{j=0}^{r_\alpha} a_j^{(\alpha)} s_j^\alpha$ have good distribution.

- Express the CY 3-fold in terms of Kähler cone sections $s_j^{(\alpha)}$
  - Problem 1: too many sections! Problem 2: relations among sections!
- First find relations among sections ...
  - Groebner basis analysis using Singular (access via Sage)
  - Linear algebra routine (faster, requires generic points in section space)

$$\prod_I s_I^{f_I} = \prod_J s_J^{g_J} \Leftrightarrow \prod_I s_I^{h_I} = 1 \; , \; s_J = \prod_a x_a^{M_{a,J}} \implies \sum_I M_{a,I} h_I = \vec{0}_a$$

- ... then combine relations + hypersurface eq: CY 3-fold as non-complete intersection in $\hat{\mathcal{A}} \cong \bigotimes_{\alpha=1}^{h^{1,1}} \mathbb{P}^{r_\alpha}$.
- Intersect: random point sample on CY distributed wrt FS measure.

Back to slide 16

Custom loss functions in cymetric package

$$\mathcal{L}_{\text{MA}} = \left|\left|1 - \frac{1}{\kappa}\frac{\det g_{\text{pr}}}{\Omega \wedge \bar{\Omega}}\right|\right|_n ,$$

$$\mathcal{L}_{\text{dJ}} = \sum_{ijk} ||\text{Re}\ c_{ijk}||_n + ||\text{Im}\ c_{ijk}||_n , \quad \text{with } c_{ijk} = g_{i\bar{j},k} - g_{k\bar{j},i} \quad \text{and } g_{i\bar{j},k} = \partial_k g_{i\bar{j}}$$

$$\mathcal{L}_{\text{transition}} = \frac{1}{d}\sum_{(s,t)} \left|\left|g_{\text{pr}}^{(t)} - T_{(s,t)} \cdot g_{\text{pr}}^{(s)} \cdot T_{(s,t)}^{\dagger}\right|\right|_n , \quad T_{(s,t)} \text{ transition matrix}$$

$$\mathcal{L}_{\text{Ricci}} = ||R||_n = \left|\left|\partial\bar{\partial} \ln \det g_{\text{pr}}\right|\right|_n$$

$$\mathcal{L}_{\text{K-class}} = \frac{1}{h^{11}}\sum_{i=1}^{h^{11}} \left|\left|\mu_{J_{\text{FS}}}(\mathcal{L}_i) - \int_X (J_{\text{pr}})^{n-1}\mathcal{F}_i\right|\right|_n$$

Derivatives – TensorFlow gradient tapes on input

Integral – weighted sum; 2-step training loop

Multiple Kähler moduli: preserving the Kähler class

Loss function preserving the Kähler class

- Could define a loss function fixing curve, divisor and CY volumes (but have not; this requires sampling points on curves and divisors).
- Instead use that $\mathcal{O}_X(k)$ (line bundle over X with $c_1 = [k^\alpha J_\alpha]$) has slope

$$\mu_J := \int_X J \wedge J \wedge c_1(\mathcal{O}_X(k)) = -\frac{i}{2\pi}\int_X J \wedge J \wedge F = d_{\alpha\beta\gamma} t^\alpha t^\beta k^\gamma \ ,$$

The slope is topological, so agrees for metrics in the same Kähler class!

- Loss function:

$$\mathcal{L}_{\text{K-class}} = \frac{1}{h^{11}} \sum_{i=1}^{h^{11}} \left|\left| \mu_{J_{\text{FS}}}(L_i) - \int_X J_{\text{pr}} \wedge J_{\text{pr}} \wedge F_i \right|\right|_n$$

Multiple Kähler moduli: preserving the Kähler class

Loss function preserving the Kähler class

- Loss function: $\mathcal{L}_{\text{K-class}} = \frac{1}{h^{11}} \sum_{i=1}^{h^{11}} \left|\left| \mu_{J_{\text{FS}}}(L_i) - \int_X J_{\text{pr}} \wedge J_{\text{pr}} \wedge F_i \right|\right|_n$
- Integral requires many points $\rightsquigarrow$ 2-batch training loop.
- Cross-check after training: compare volume and line bundle slopes (from intersection numbers, FS and CY metric).