От символьной регрессии к физическому искусственному интеллекту

Андрей Устюжанин[1]

[1] НИУ ВШЭ

2021, НИИЯФ МГУ

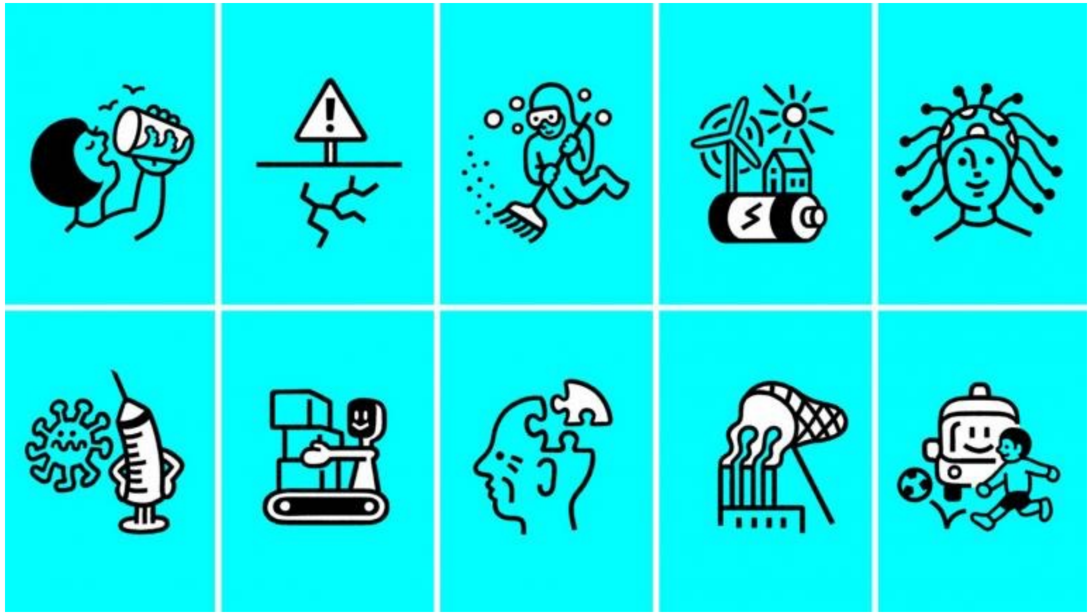
Shameless plug

- Development and application of Machine Learning methods for solving tough scientific challenges;
- Member of collaborations LHCb, SHiP, OPERA, NEWSdm, KIWI
- Research Project examples:
  - Storage/speed optimization for LHCb triggers;
  - Particle identification algorithms;
  - Optimization of detector devices;
  - Fast and meaningful physical process simulation.
- Co-organization of ML challenges: Flavours of Physics, TrackML
- 7 Summer schools on Machine Learning for High-Energy Physics
- *Open for interns, graduate students and post doc researchers!*

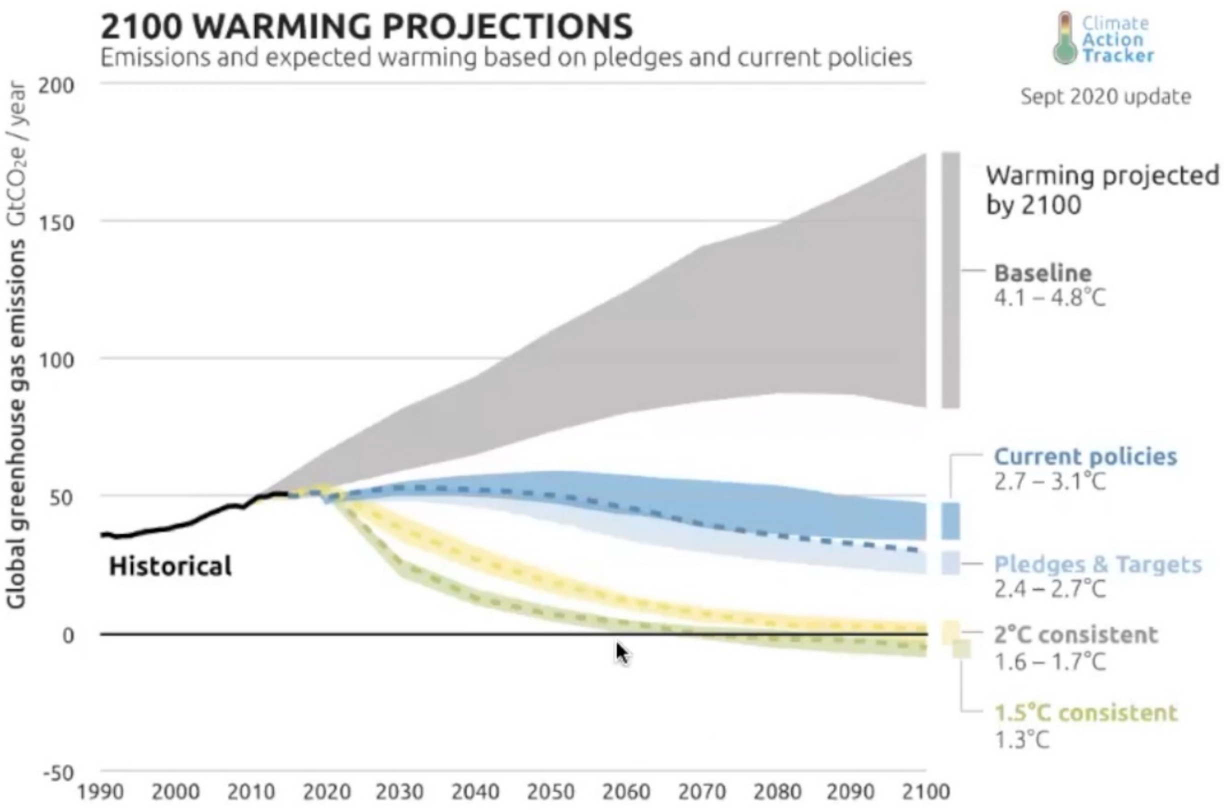
LAMBDA • HSE

hse_lambda

Motivation



- Growing demands for new:
  - Technologies (manufacturing processes), materials, drugs, …



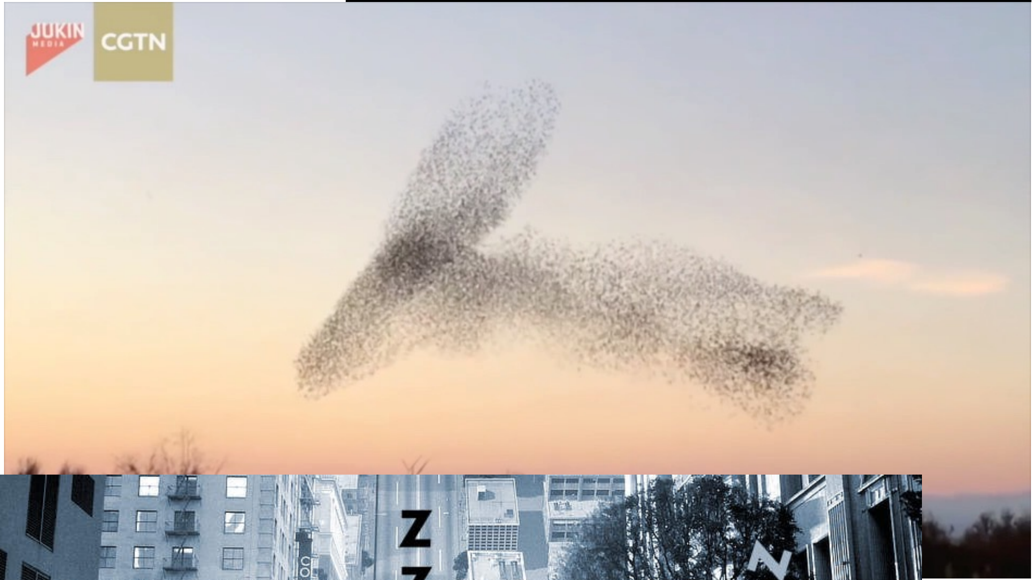
https://climate.mit.edu/posts/ten-big-global-challenges-technology-could-solve

Why AI + Physics?

- Physics has been the strong inspiration source for AI
  - Simulated annealing, Energy GANs, Langevin gradient descent, diffusion models, …
- Physicists deal with the Universe on the widest range of scales (from quarks to galaxies)
  - Approaches that easy to change the scale of the object under the study
  - Natural account for uncertainty in data and in models
- Has good tradition on compressing empirical facts/information into succinct principles
  - Powerful theoretical foundation + experimental verification tradition
- Rich mathematical language for natural phenomena description
- Strong real phenomena simulation experience, methods and tools

Why AI + Physics?

https://www.youtube.com/watch?v=0dskCpuxqtI



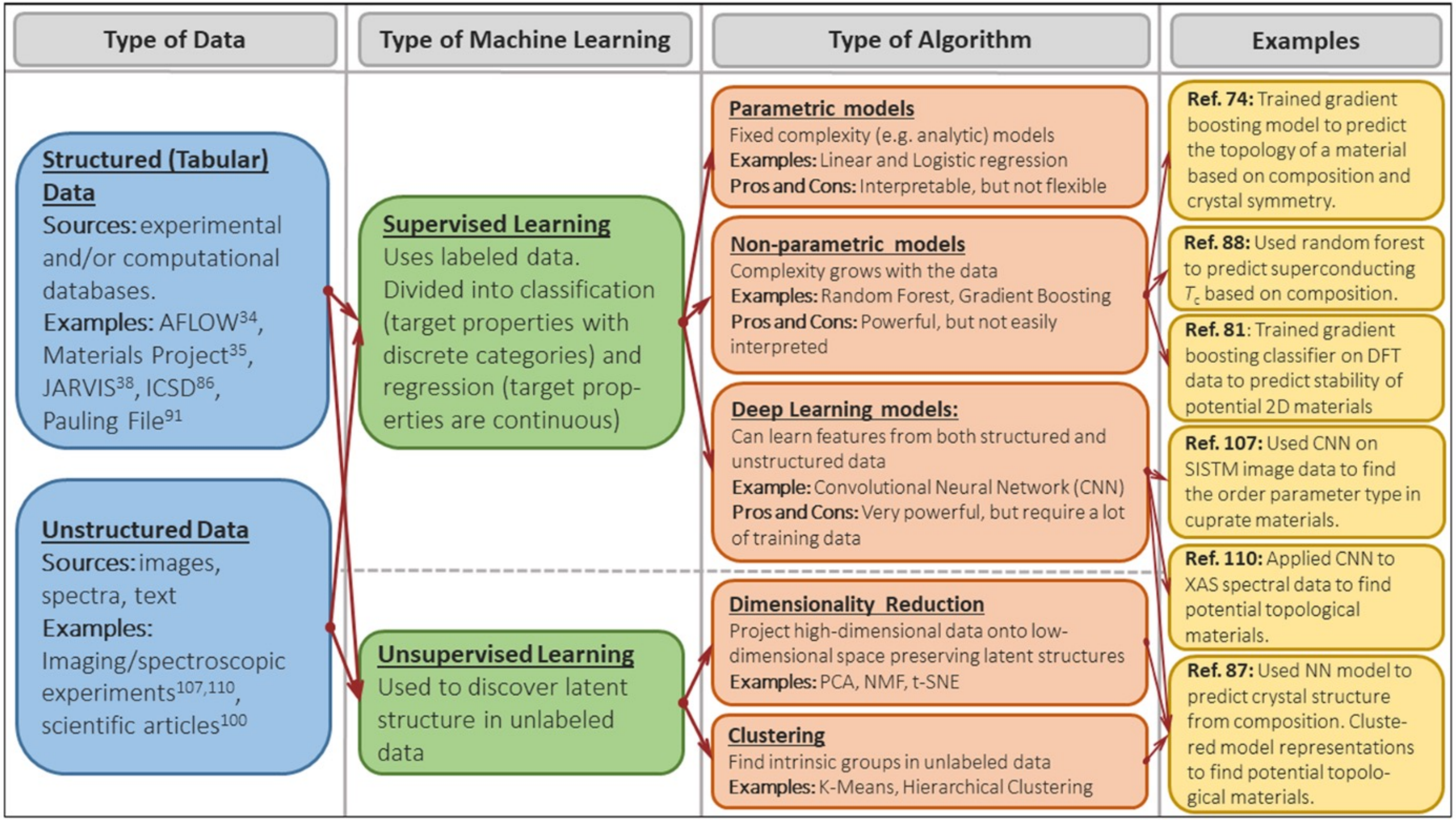
Challenges for describing complex systems

- Emergence
- Dark matter search
- Quantum vs classical gravity
- Self-driving research

atwentwrongwith.com/2020/07/23/what-went-wrong-with-inception-2010/

https://www.artstation.com/artwork/AqawDV

Commonly used ML tasks and algorithms



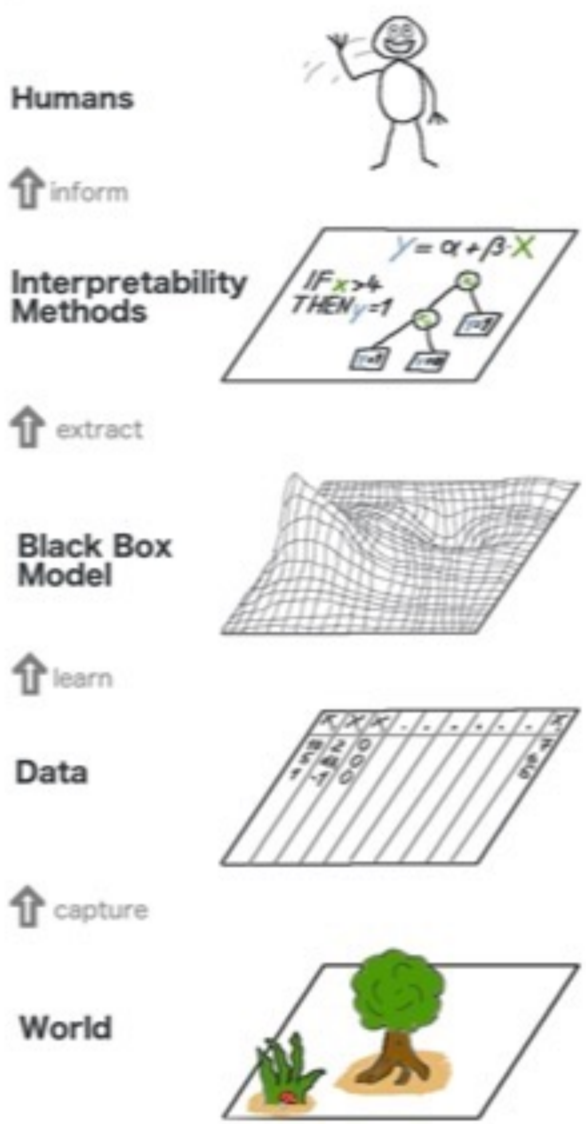
https://doi.org/10.1038/s43246-021-00209-z

Machine Learning (ML) challenges

- Incremental data handling
- Data representation
  - Account for noise
- Forward modelling (fast simulation)
  - Generative model, differentiable
  - Interpretability
  - Inductive bias
- Inference (inverse problem)
  - Physical

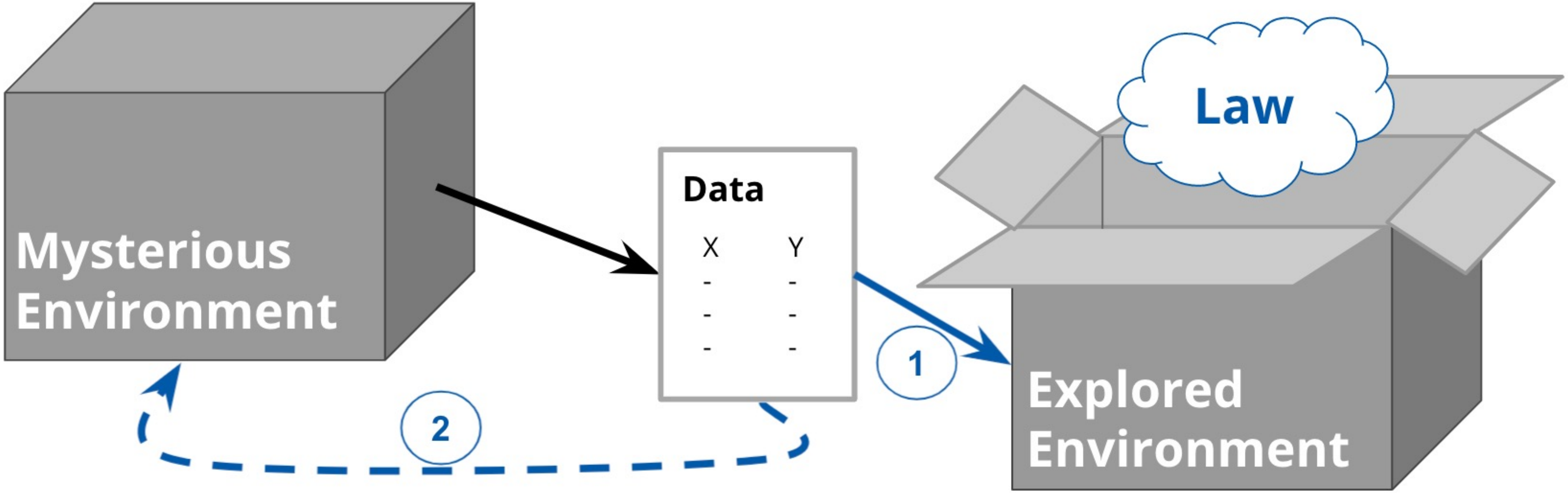
Model interpretation

- Model-agnostic
  - Example-based
  - Global: how features affect the prediction on average
  - Local: explain individual predictions
- Model-centric, e.g., for neural networks:
  - Learned Features: What features has the neural network learned?
  - Pixel Attribution (Saliency Maps): How did each pixel contribute to a particular prediction?
  - Which more abstract concepts has the neural network learned?
  - Adversarial Examples: How can we trick the neural network?
  - How influential was a training data point for a certain prediction?



https://christophm.github.io/interpretable-ml-book/agnostic.html

Symbolic regression problem statement



Laws: statements y=f(x), f - symbolic expression

1. Find the formula that best fits the given dataset
2. Add the most informative data to the dataset

Basic approach

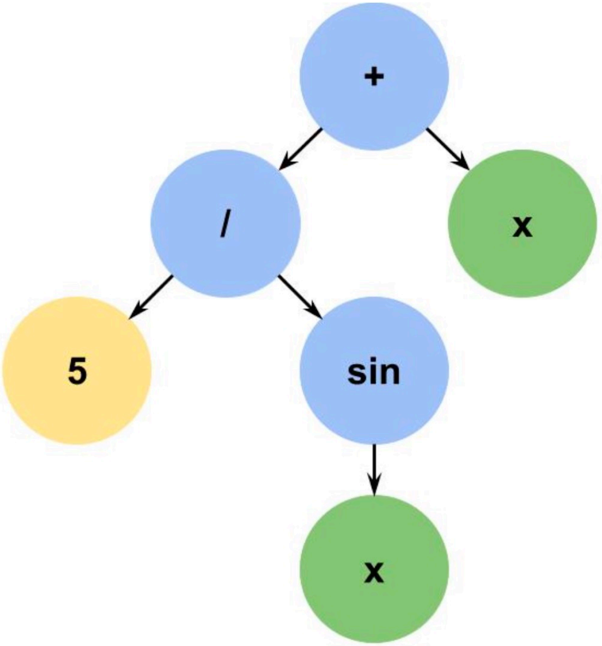
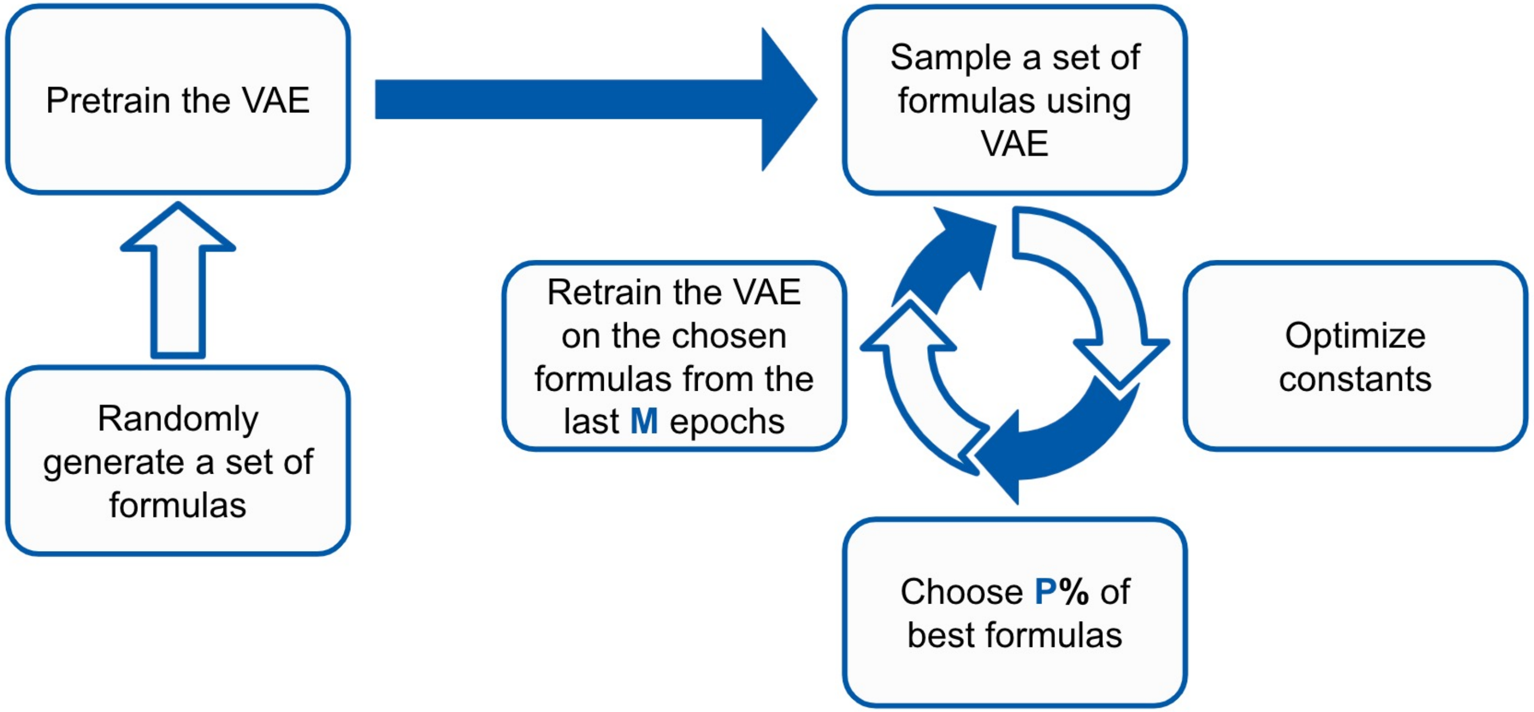
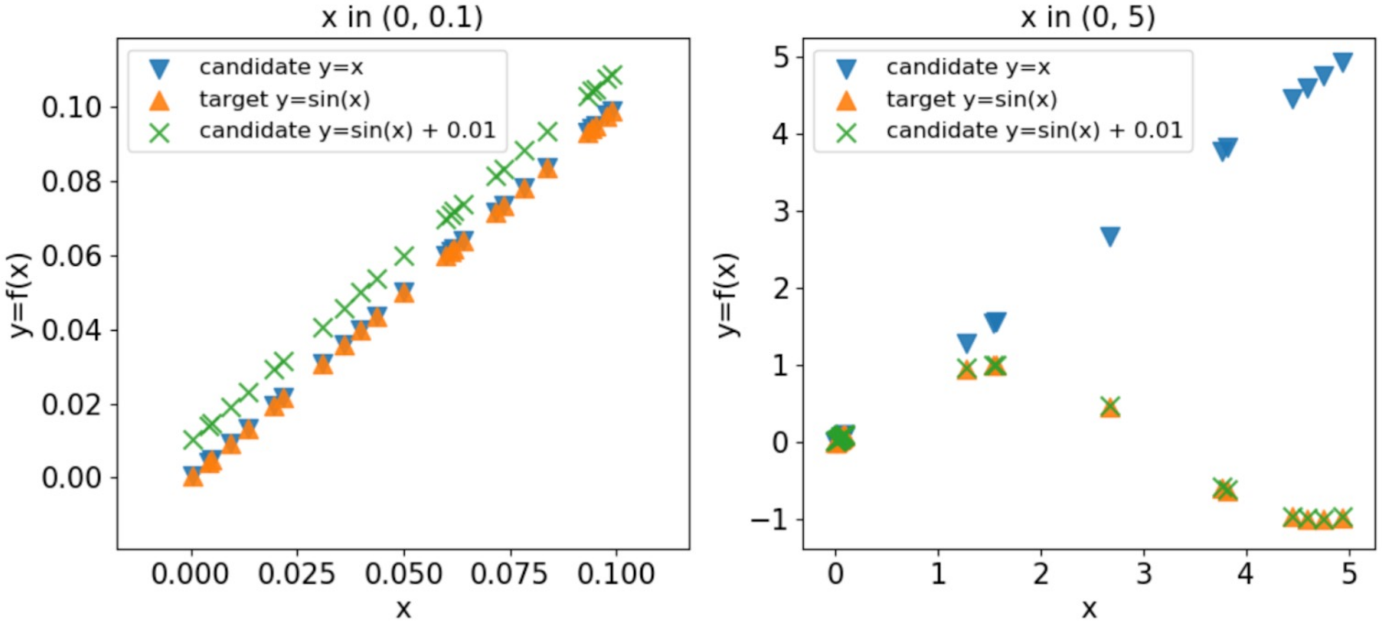


Figure 2: Formula tree for $y = \frac{5}{\sin x} + x$



Active Learning

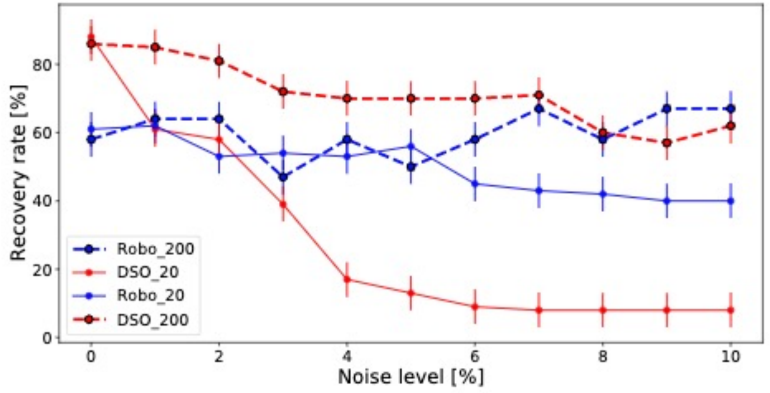
- Add points to the training sample based on evaluation of the best trained formula
  - Experiments are expensive
  - Faster discovery
  - Less data, less computational resources
- Possible approaches:
  - **Random sampling** - choose a new point randomly
  - **Full variance** - choose the point with max variance based on the whole retraining set
  - **Top-10 variance** - choose the point with max variance based on the top-10 generated formulas



Performance comparison

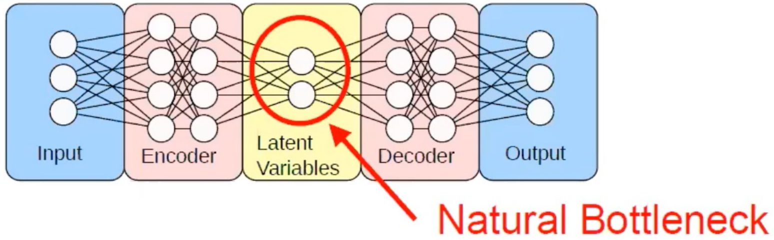
- Benchmarks: Ngyuen, Feynman
- Metrics
  - Recovery rate
  - Noise-resistance
  - Data-size dependence
- Notable players
  - Deep Symbolic Regression, Petersen et al. 2020
  - AI Feynman, Urdescu, Tegmark 2020

Name	Expression	Dataset I	Dataset II
Nguyen-1	$x_1^3+x_1^2+x_1$	$U(-1,1,20)$	$U(-1,1,200)$
Nguyen-2	$x_1^4+x_1^3+x_1^2+x_1$	$U(-1,1,20)$	$U(-1,1,200)$
Nguyen-3	$x_1^5+x_1^4+x_1^3+x_1^2+x_1$	$U(-1,1,20)$	$U(-1,1,200)$
Nguyen-4	$x_1^6+x_1^5+x_1^4+x_1^3+x_1^2+x_1$	$U(-1,1,20)$	$U(-1,1,200)$
Nguyen-5	$sin(x_1^2)cos(x_1)-1$	$U(-1,1,20)$	$U(-1,1,200)$
Nguyen-6	$sin(x_1)+sin(x_1+x_1^2)$	$U(-1,1,20)$	$U(-1,1,200)$
Nguyen-7	$log(x_1+1)+log(x_1^2+1)$	$U(0,2,20)$	$U(0,2,200)$
Nguyen-8	$sqrt(x_1)$	$U(0,4,20)$	$U(0,4,200)$
Nguyen-9	$sin(x_1)+sin(x_2^2)$	$U(0,1,20)$	$U(0,1,200)$
Nguyen-10	$2sin(x_1)cos(x_2)$	$U(0,1,20)$	$U(0,1,200)$
Nguyen-11	$x_1^{x_2}$	$U(0,1,20)$	$U(0,1,200)$
Nguyen-12	$x_1^4-x_1^3+0.5x_2^2-x_2$	$U(0,1,20)$	$U(0,1,200)$



Unsupervised approach

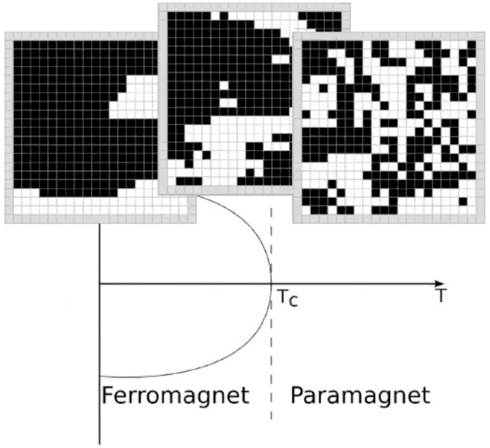
(Variational) Autoencoder
2d Ising Model



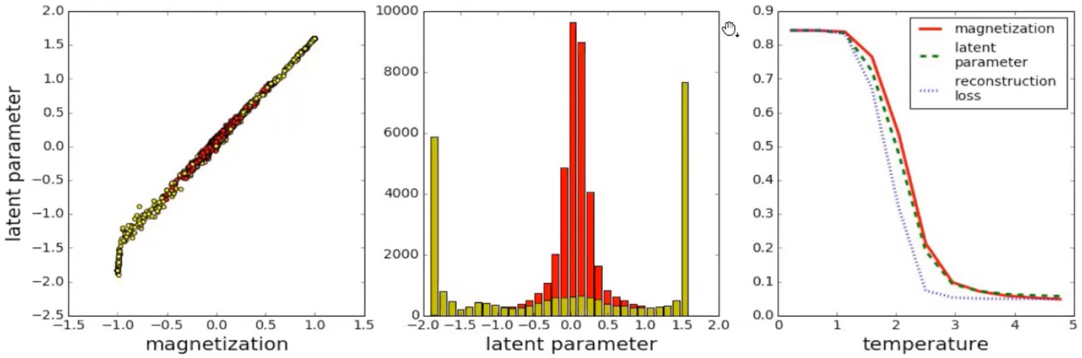
Objective: Minimize Reconstruction error

$$MSE = \frac{1}{N}\sum_{k}\left\|x_k - F(x_k)\right\|^2$$

- Data: Monte Carlo samples
- Train everywhere in phase diagram
- Labels: None



(Variational) Autoencoder
2d Ising Model



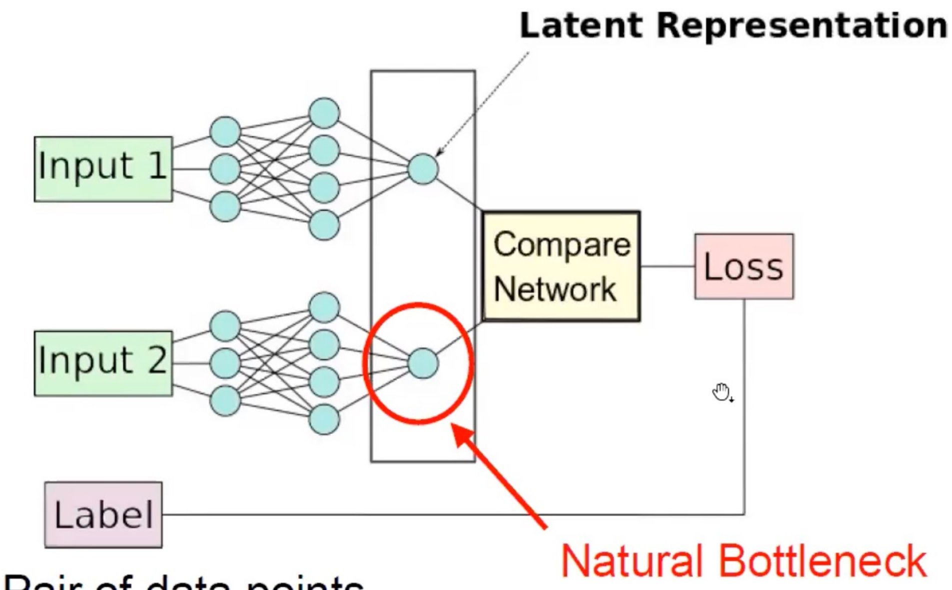
Ferromagnetic Ising model on the square lattice

Wetzel, PRE 2017

- Latent parameter corresponds to magnetization
- Identification of phases: Latent representations are clustered
- Location of phases: Magnetization, latent parameter and reconstruction loss show a steep change at the phase transition.

Similarity finding

Siamese Neural Networks



- Input : Pair of data points
- Label : same / different
- Network pair contains identical neural networks with shared weights

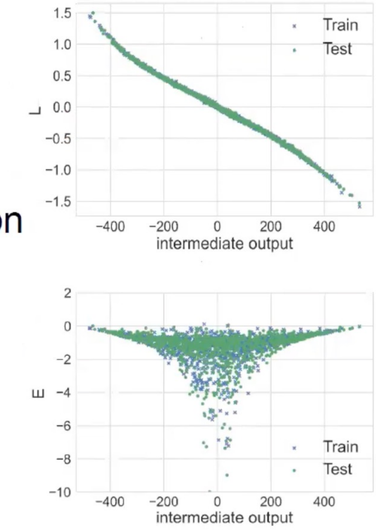
Wetzel, Melko, Scott, Panju, Ganesh, PRR 2020

Siamese Neural Networks
Particle in Gravitational Potential

Results:

Training accuracy : 98%

Test accuracy : 97%

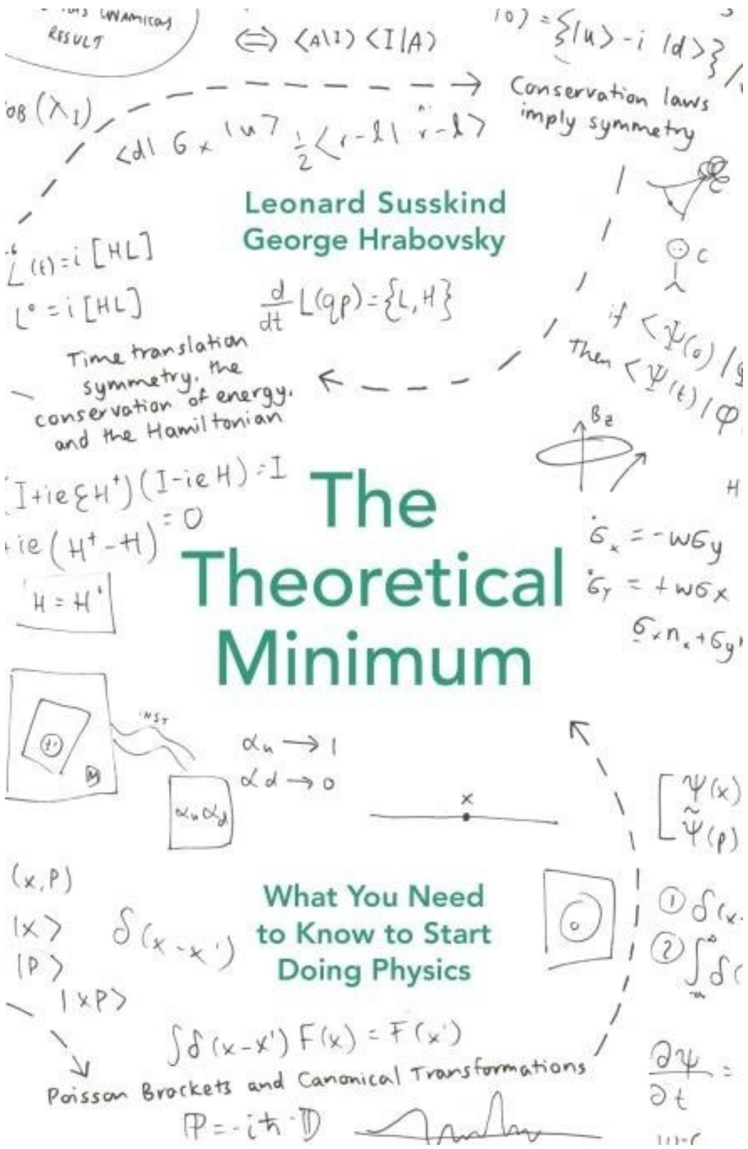


- Interpretation by polynomial regression on latent representation:

$$
\begin{aligned}
f(\mathbf{x}) \approx & -403.71 x v_y - 4.85 x - 0.58 x y \\
& - 0.17 x v_x - 0.02 v_y^2 - 0.01 v_x v_y \\
& + 0.00 v_y^2 + 0.01 v_y + 0.02 v_x \\
& + 0.45 x^2 + 0.66 y^2 + 0.74 \\
& + 0.99 y v_y + 1.24 y + 402.44 y v_x \\
\approx & -403 \underbrace{(x v_y - y v_x)}_{=L_z}
\end{aligned}
$$

- Network has learned the angular momentum to infer its prediction.

Looking to physics for inductive bias



- Inductive biases are needed to make learning tractable, even in data-rich domains.

- But the lesson seems to be that we should make the inductive bias as general as possible.
  See: Sutton 2019 - The Bitter Lesson

- Example:
  - Forces are summed at each receiver
  - Message are summed at each receiver
  - In a 2D simulation the forces are 2D
  - …
  - 2D messages => messages = forces?

Hamiltonian Systems

- Generalized coordinates **q** and conjugate momenta **p**:

$$p = \frac{\partial L}{\partial \dot{q}}.$$

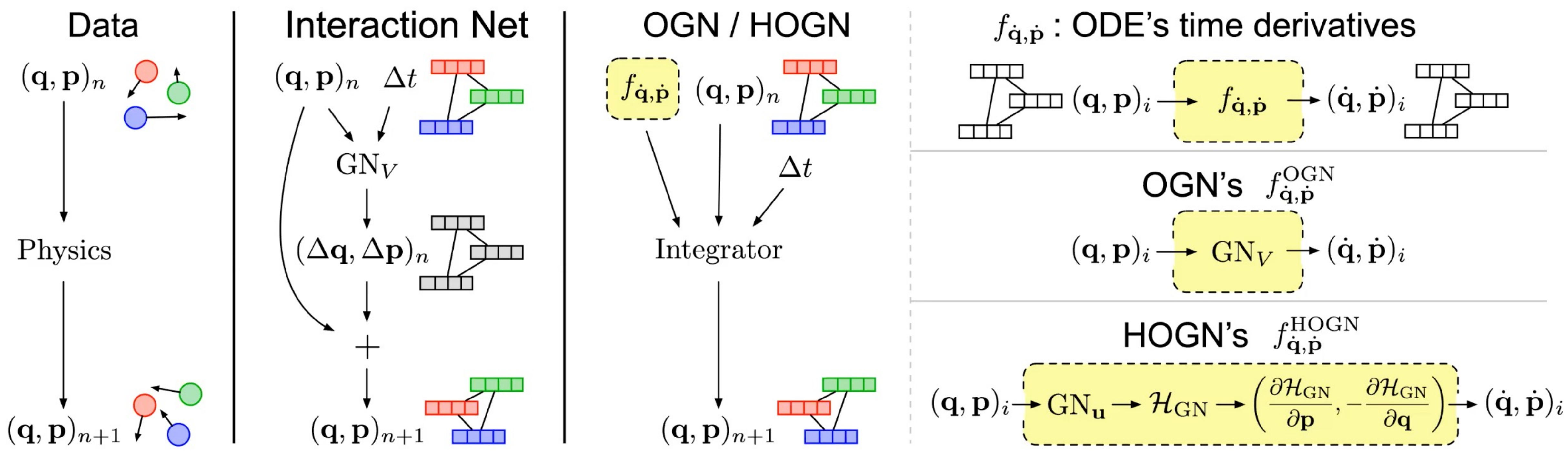
- Hamiltonian (usually corresponds to system total energy: T + V)

$$H = \sum_{i=1}^{n} \dot{q}_i \frac{\partial L}{\partial \dot{q}_i} - L$$

- Evolution equations

$$\frac{\partial \mathcal{H}}{\partial q^j} = -\dot{p}_j \quad , \quad \frac{\partial \mathcal{H}}{\partial p_j} = \dot{q}^j$$

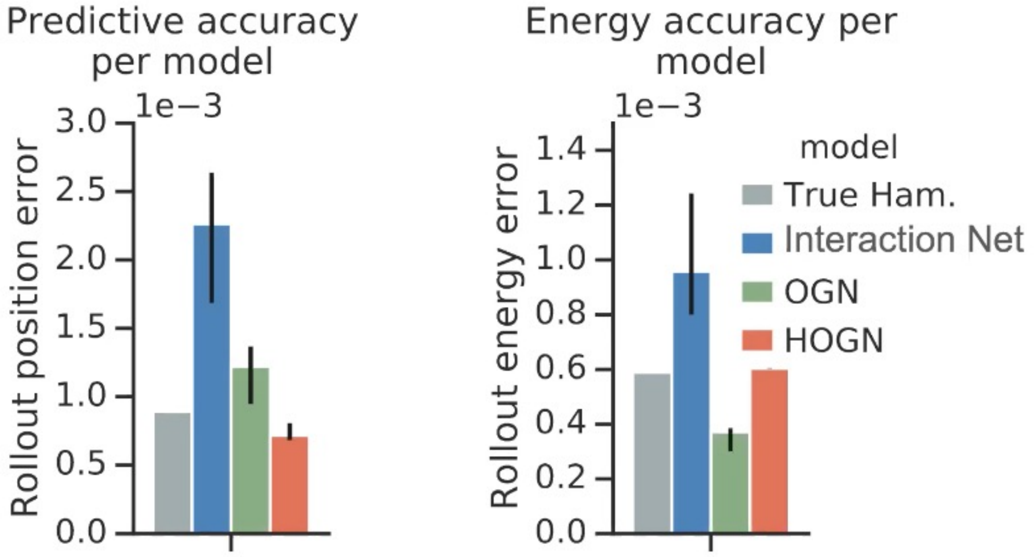
Hamiltonian ODE Graph Network



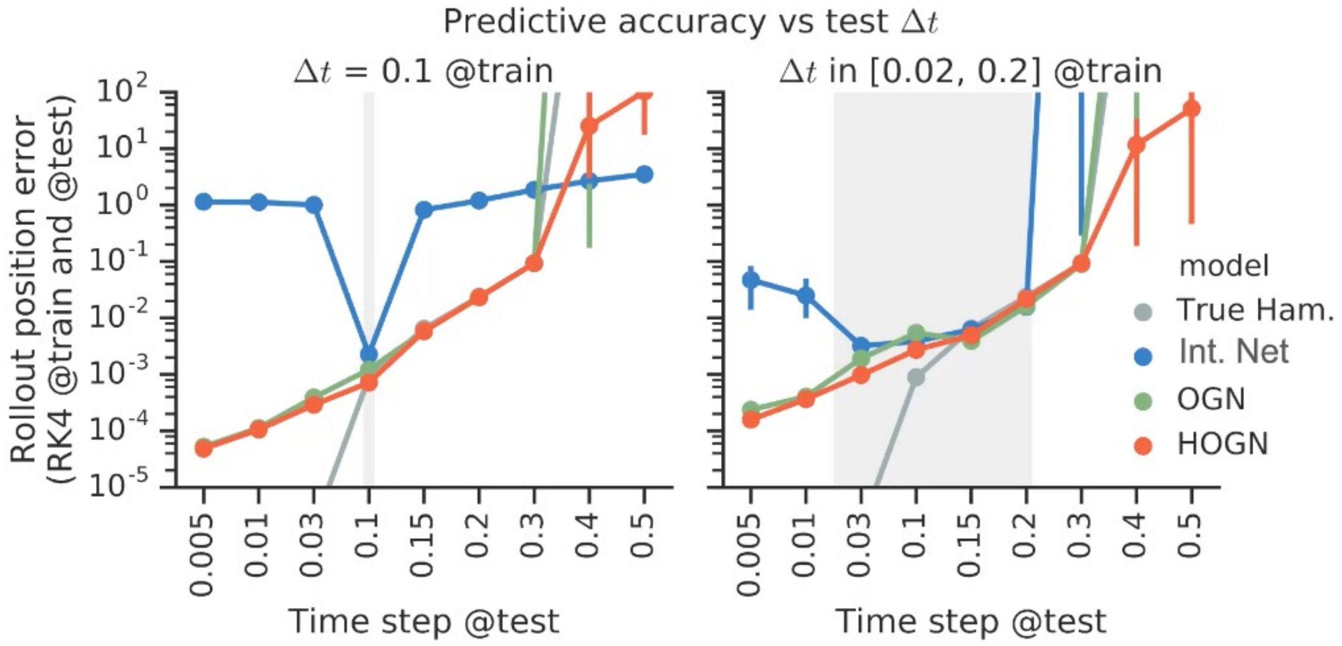
Sanchez-Gonzalez et al., 2019, arXiv/NeurIPS 2019 workshop

Hamiltonian ODE Graph Network

Performance



Generalization to untrained time steps



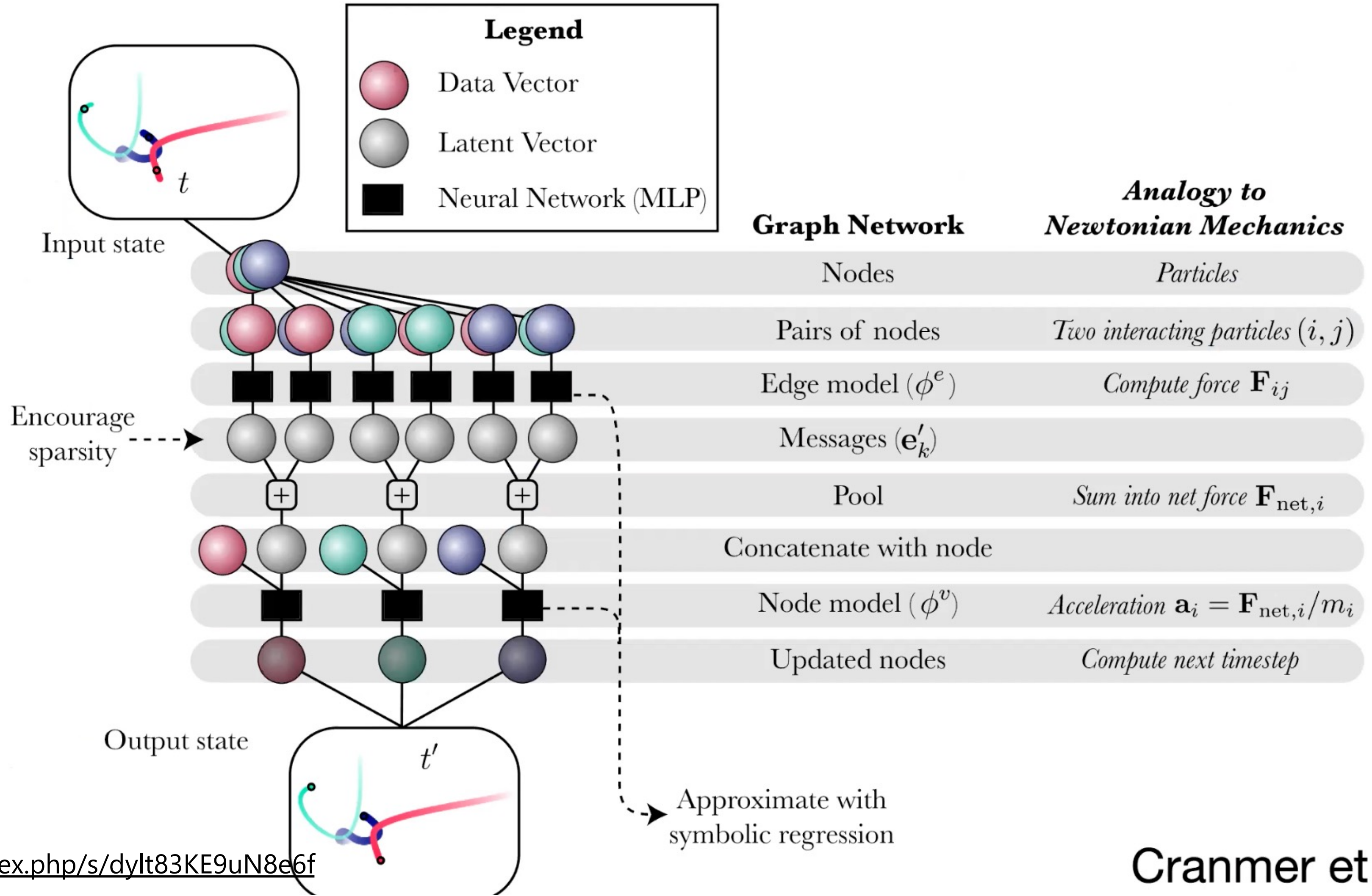
- OGN and HOGN used RK4 integrator (we also tested lower order RK integrators)
- We also tested symplectic integrators, and found HOGN has better energy accuracy/conservation

Sanchez-Gonzalez et al., 2019, arXiv/NeurIPS 2019 workshop

https://owncloud.gwdg.de/index.php/s/dylt83KE9uN8e6f

Discovering symbolic physics equations

Because the GN-based learned simulator is structured in a way that has correspondences to physical mechanics, we can interpret the functions and variables in physical terms



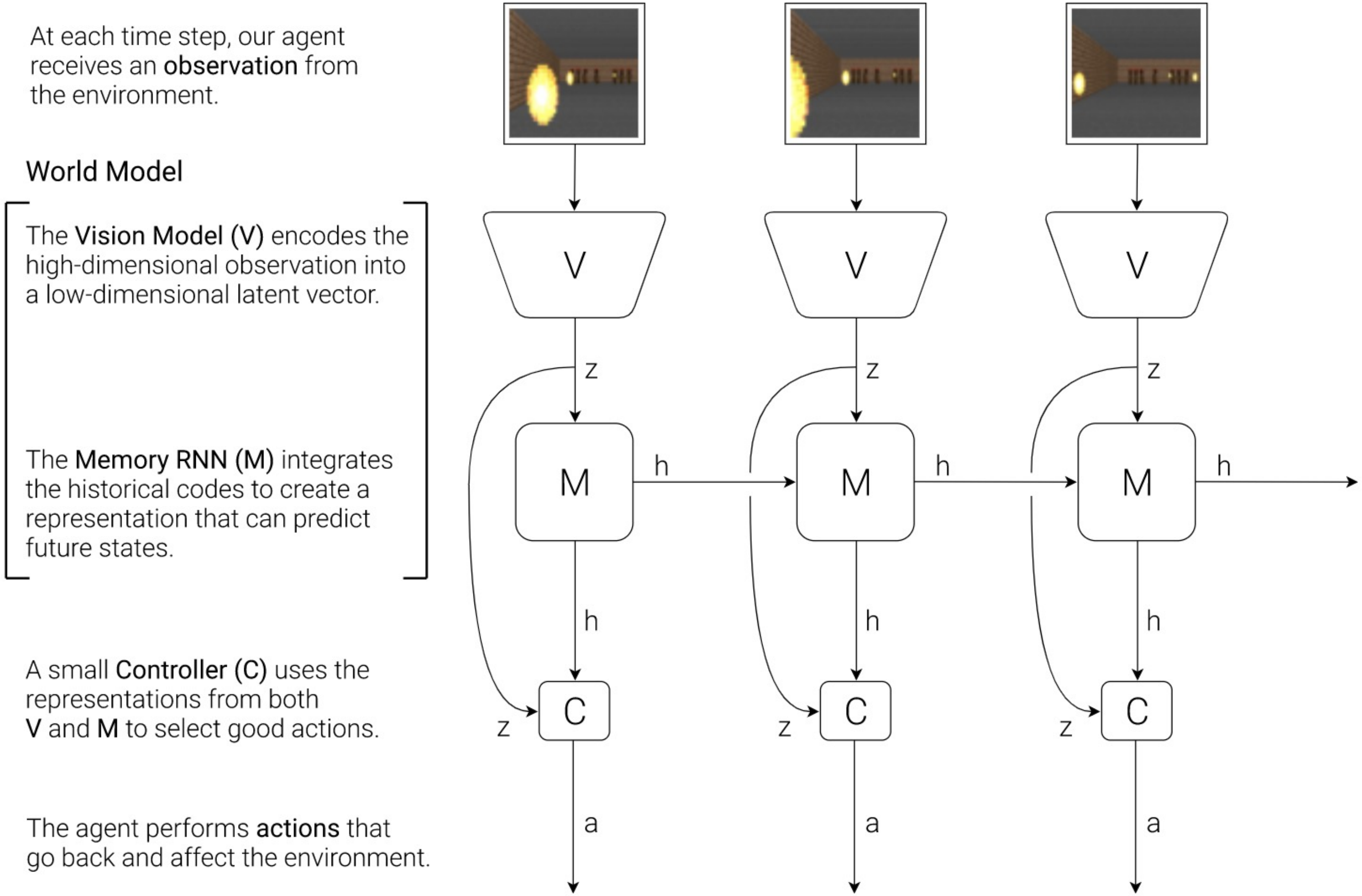
https://owncloud.gwdg.de/index.php/s/dyIt83KE9uN8e6f

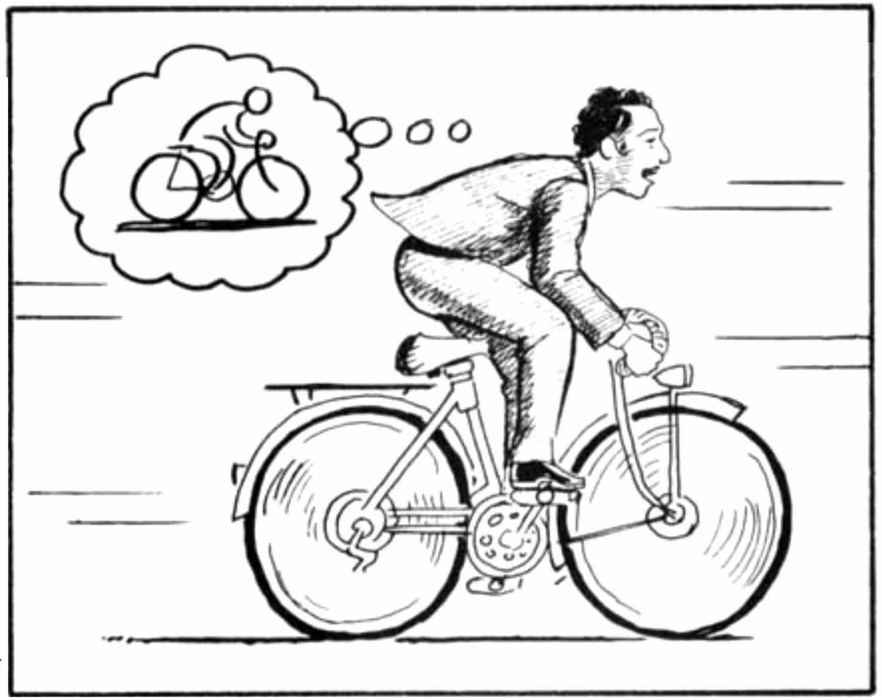
Cranmer et al., NeurIPS 2020

Simulation-based approach to discovery

- Reinforcement learning
- World-model
- Sleep-Wake cycle
- Simulation-based inference

World Model



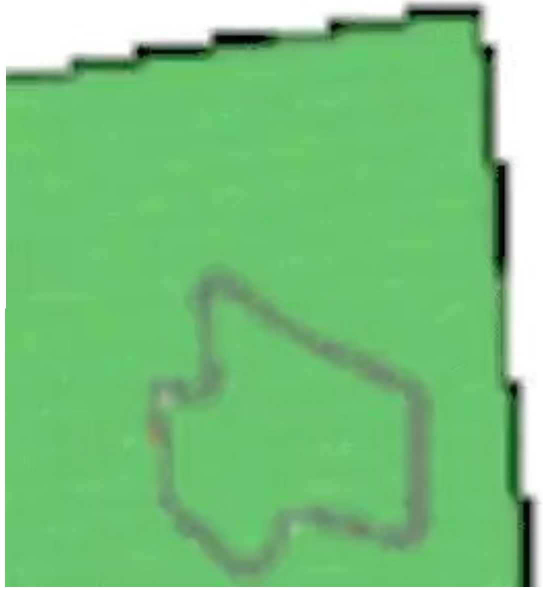


https://worldmodels.github.io

Our agent consists of three components that work closely together: Vision (V), Memory (M), and Controller (C).

World model procedure illustration

1. Collect 10,000 rollouts from a random policy.
2. Train VAE (V) to encode each frame into a latent vector $z \in \mathcal{R}^{32}$.
3. Train MDN-RNN (M) to model $P(z_{t+1} \mid a_t, z_t, h_t)$.
4. Evolve Controller (C) to maximize the expected cumulative reward of a rollout.

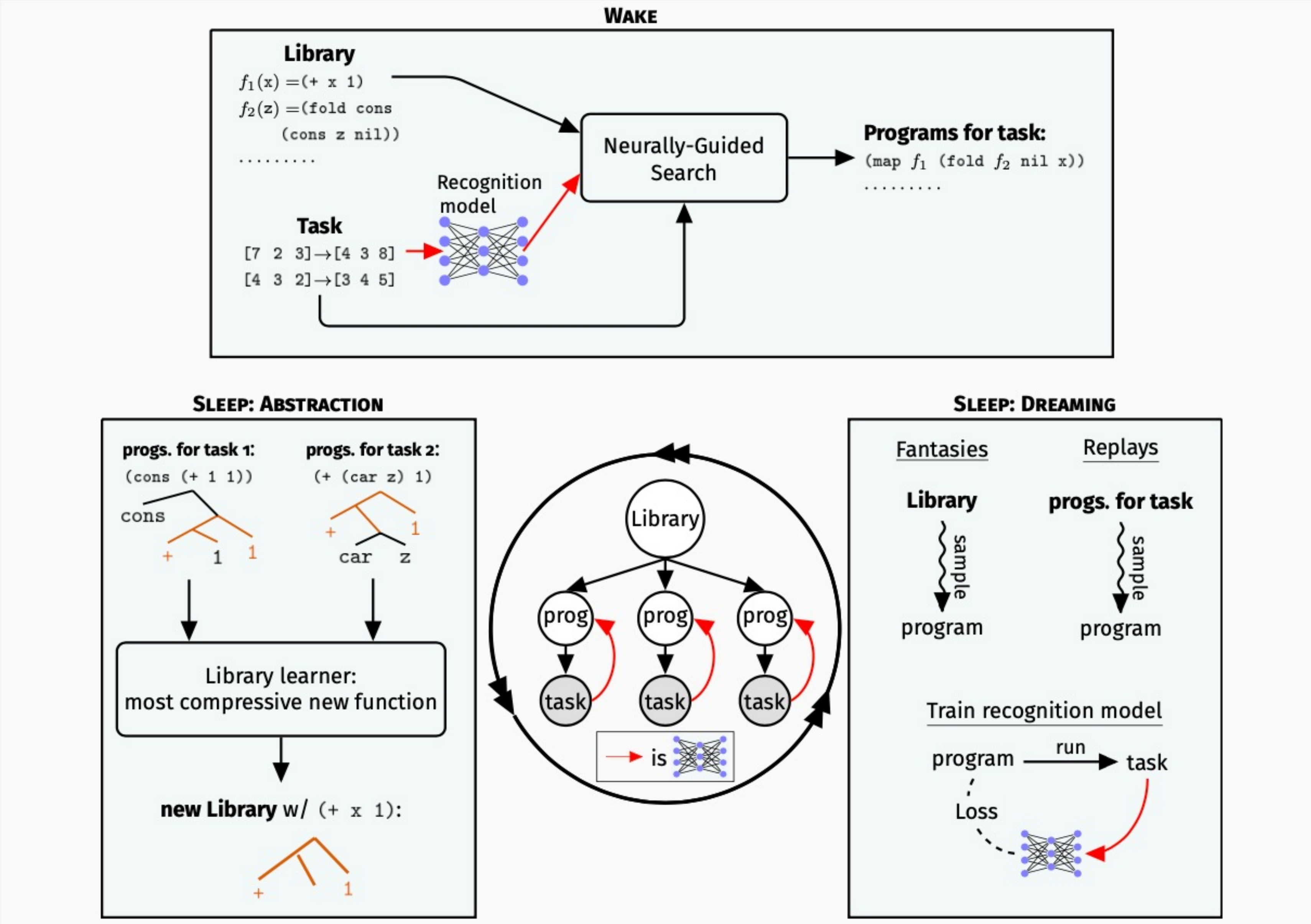


Actual observations from the environment.

What gets encoded into z_t.

https://worldmodels.github.io/assets/mp4/carracing_vae_compare.mp4

Dream Coder



https://dl.acm.org/doi/10.1145/3453483.3454080

DreamCoder Domains

List Processing

Sum List

[1 2 3] → 6

[4 6 8 1] → 17

Double

[1 2 3] → [2 4 6]

[4 5 1] → [8 10 2]

Text Editing

Abbreviate

Allen Newell → A.N.

Herb Simon → H.S.

Drop Last Three

shrdlu → shr

shakey → sha

Regexes

Phone numbers

(555) 867-5309

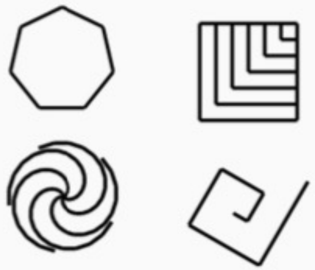
(650) 555-2368

Currency

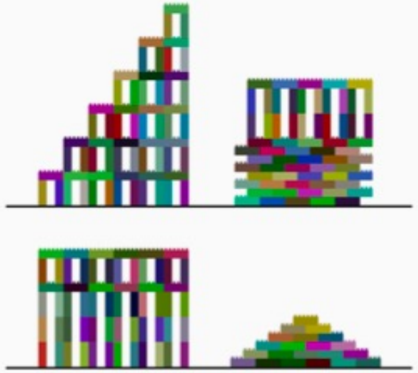
$100.25

$4.50

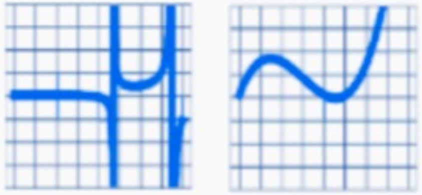
LOGO Graphics



Block Towers



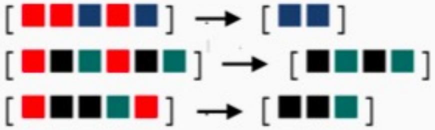
Symbolic Regression



$y = f(x)$

Recursive Programming

Filter Red



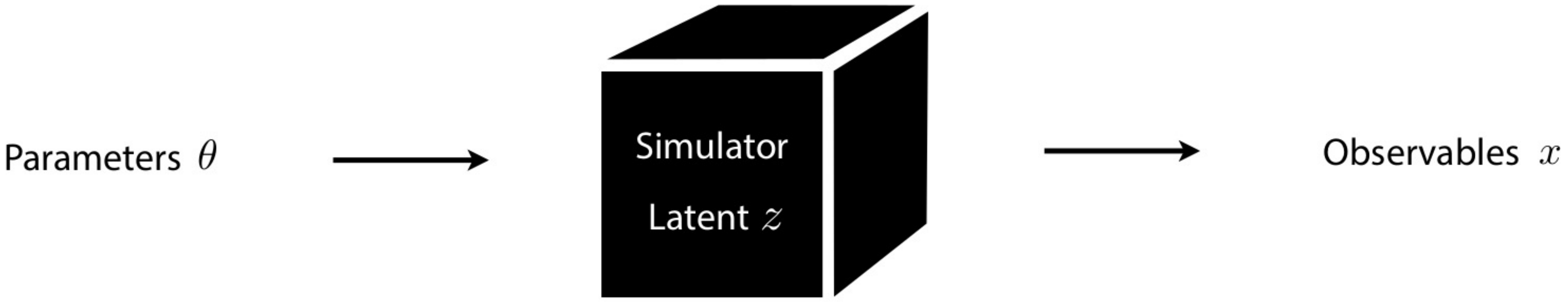
Physical Laws

$$\vec{a} = \frac{1}{m}\sum_i \vec{F}_i$$

$$\vec{F} \propto \frac{q_1 q_2}{|\vec{r}|^2}\hat{r}$$

Ellis, Wong, Nye, ..., Solar-Lezama, Tenenbaum. PLDI 2021.

Simulation-based inference



Prediction:

- Well-motivated mechanistic, causal model
- Simulator can generate samples $x \sim p(x|\theta)$

Inference:

- Interactions between low-level components lead to challenging inverse problems
- Likelihood $p(x|\theta) = \int \mathrm{d}z \; p(x, z|\theta)$ is intractable

Simulation-based inference

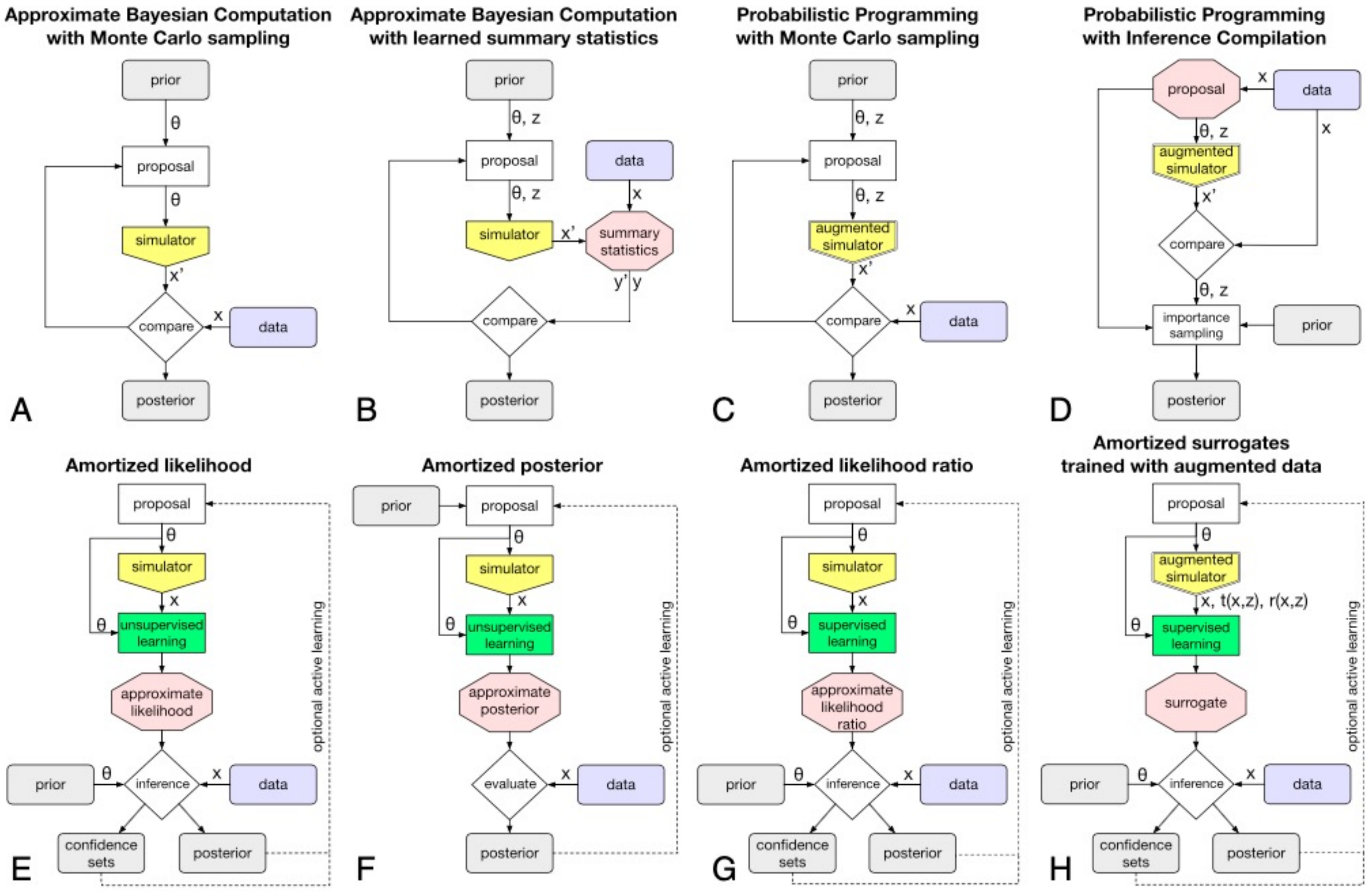


Fig. 1. *(A–H)* Overview of different approaches to simulation-based inference.

https://doi.org/10.1073/pnas.1912789117

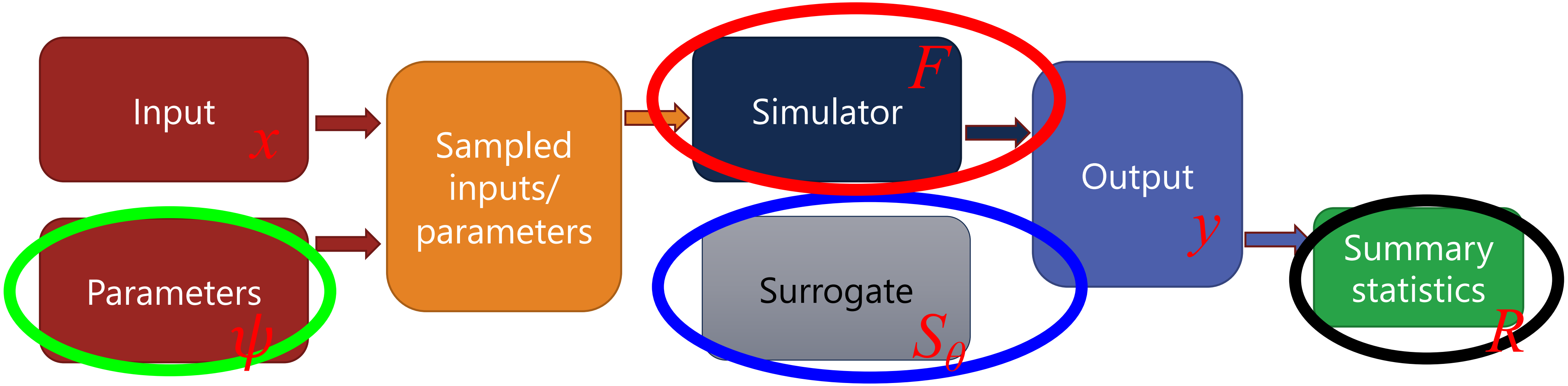
TL;DR:

Let's approximate a stochastic black-box with a local generative surrogate.

This allows computing gradients of the **objective** w.r.t. parameters of the black-box.

$$\mathbb{E}[\mathcal{R}(\boldsymbol{y})] = \int \mathcal{R}(\boldsymbol{y}) \boxed{p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\psi})} q(\boldsymbol{x}) d\boldsymbol{x} d\boldsymbol{y} \approx \frac{1}{N}\sum_{i=1}^{N} \mathcal{R}\left(\boxed{F(\boldsymbol{x}_i;\boldsymbol{\psi})}\right)$$

$\boldsymbol{y}_i = F(\boldsymbol{x}_i;\boldsymbol{\psi}) \sim p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\psi})$,
$x_i \sim q(\boldsymbol{x})$

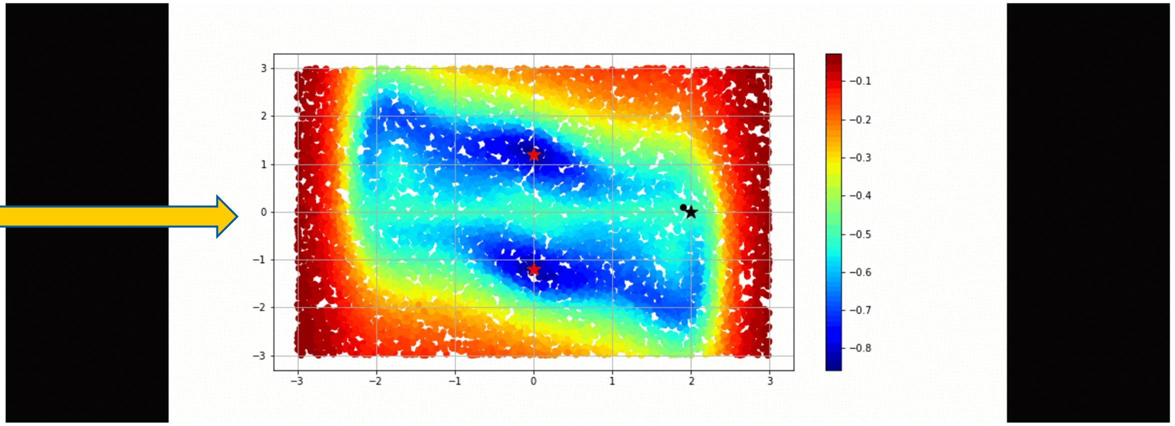


$$\nabla_{\psi}\mathbb{E}[\mathcal{R}(\boldsymbol{y})] \approx \frac{1}{N}\sum_{i=1}^{N} \nabla_{\psi} \mathcal{R}(S_\theta(\boldsymbol{z}_i, \boldsymbol{x}_i; \boldsymbol{\psi}))$$

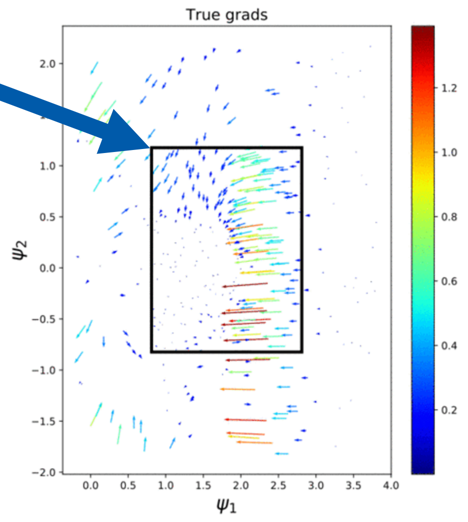
Shirobokov S., Belavin V., Kagan M, AU, Baydin A
https://arxiv.org/abs/2002.04632

Key point: training **local** generative surrogate

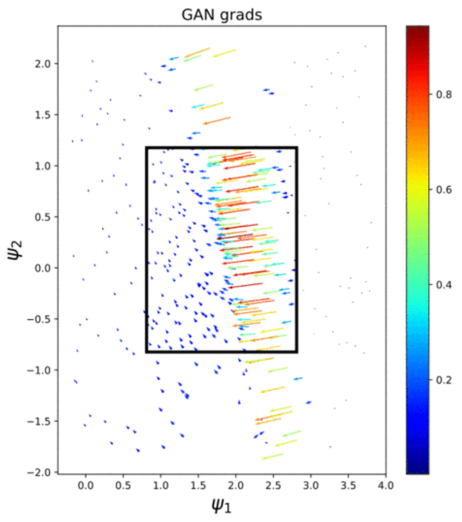
Optimization path



Area inside which the local surrogate was trained



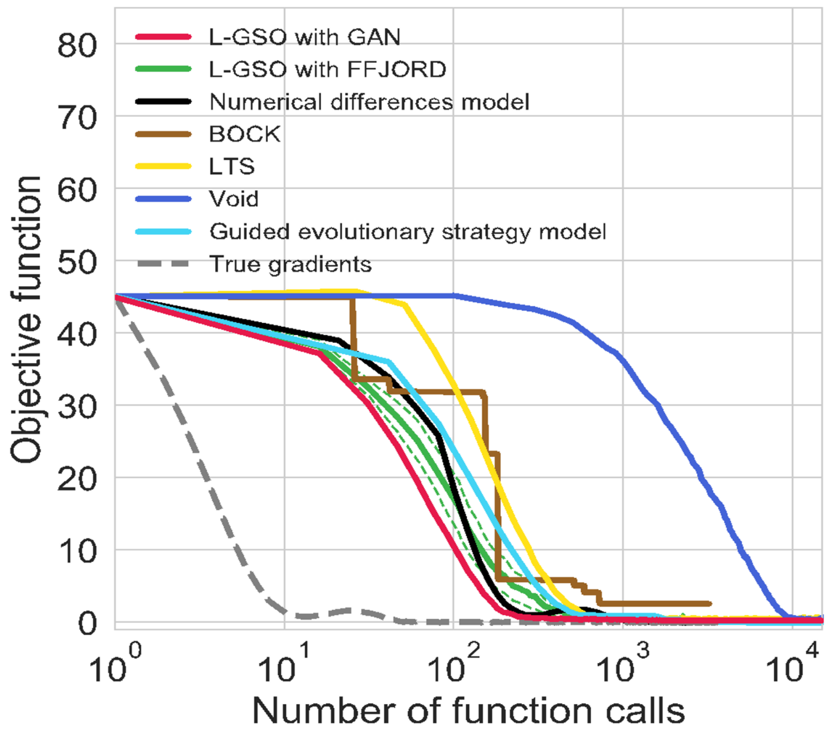
True gradients



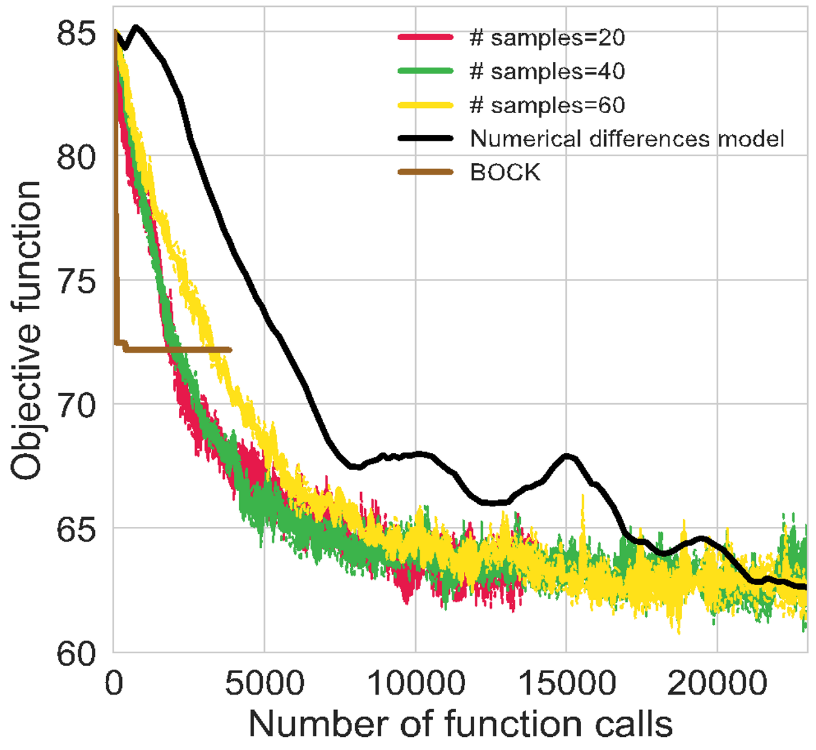
Surrogate gradients

- ✓ gradients of the non-linear surface are well estimated inside the local area.
- ✓ L-GSO outperforms **all** algorithms in a high-dimensional setting when parameters lie on a lower dimension manifold.

Results on high-dimensional problems with low-dimensional manifold



Nonlinear Three Hump problem, 40dim



Neural network weights optimization, 91dim

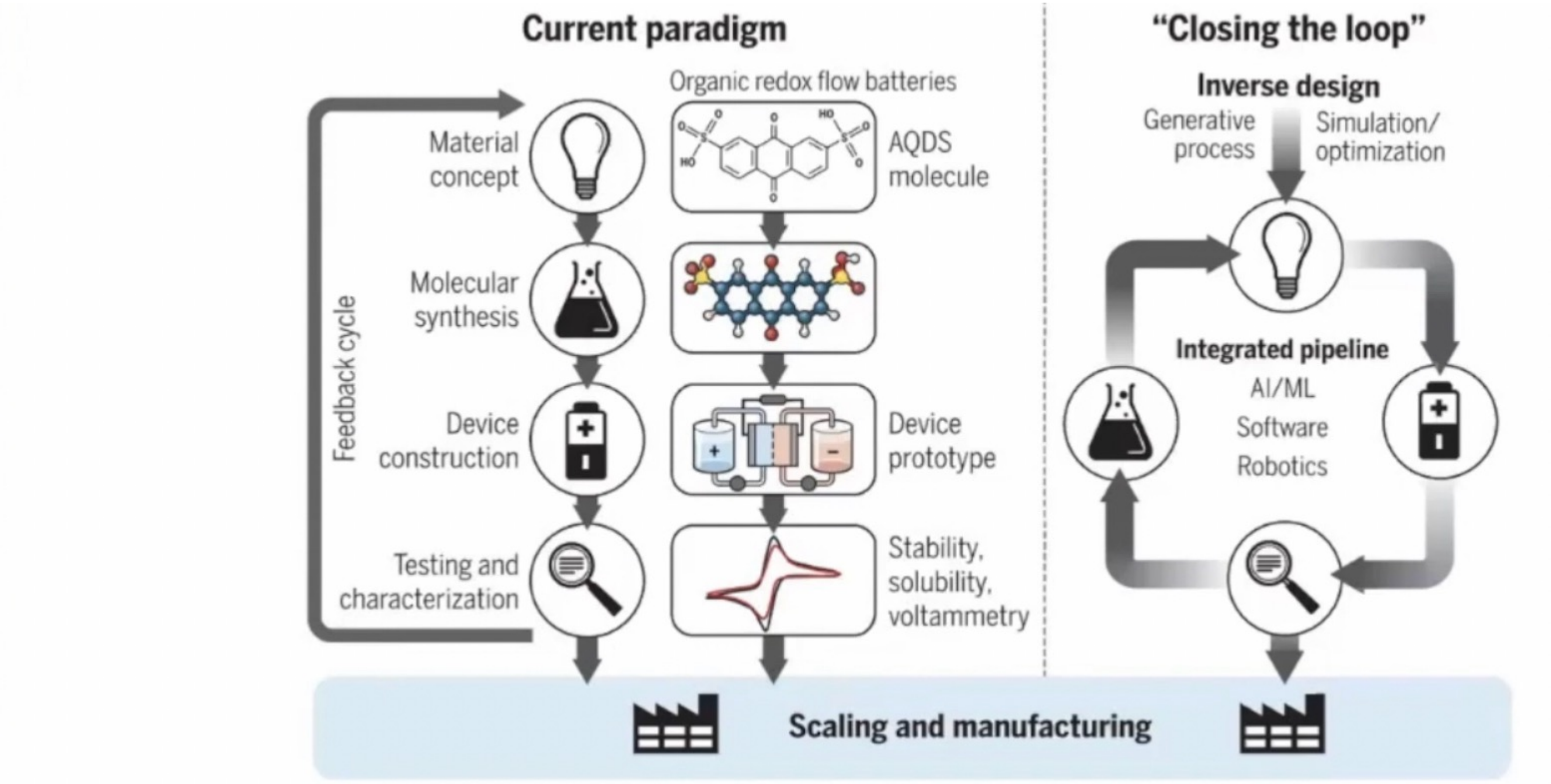
L-GSO is free from explicit variational distribution model

L-GSO outperforms **all** algorithms in a high-dimensional setting with lower dimension manifold.

1. Liu, Shuang, and Kamalika Chaudhuri. "The inductive bias of restricted f-gans." *arXiv preprint arXiv:1809.04542* (2018).

2. Uppal, Ananya, Shashank Singh, and Barnabás Póczos. "Nonparametric density estimation & convergence rates for gans under besov ipm losses." *Advances in Neural Information Processing Systems*. 2019.

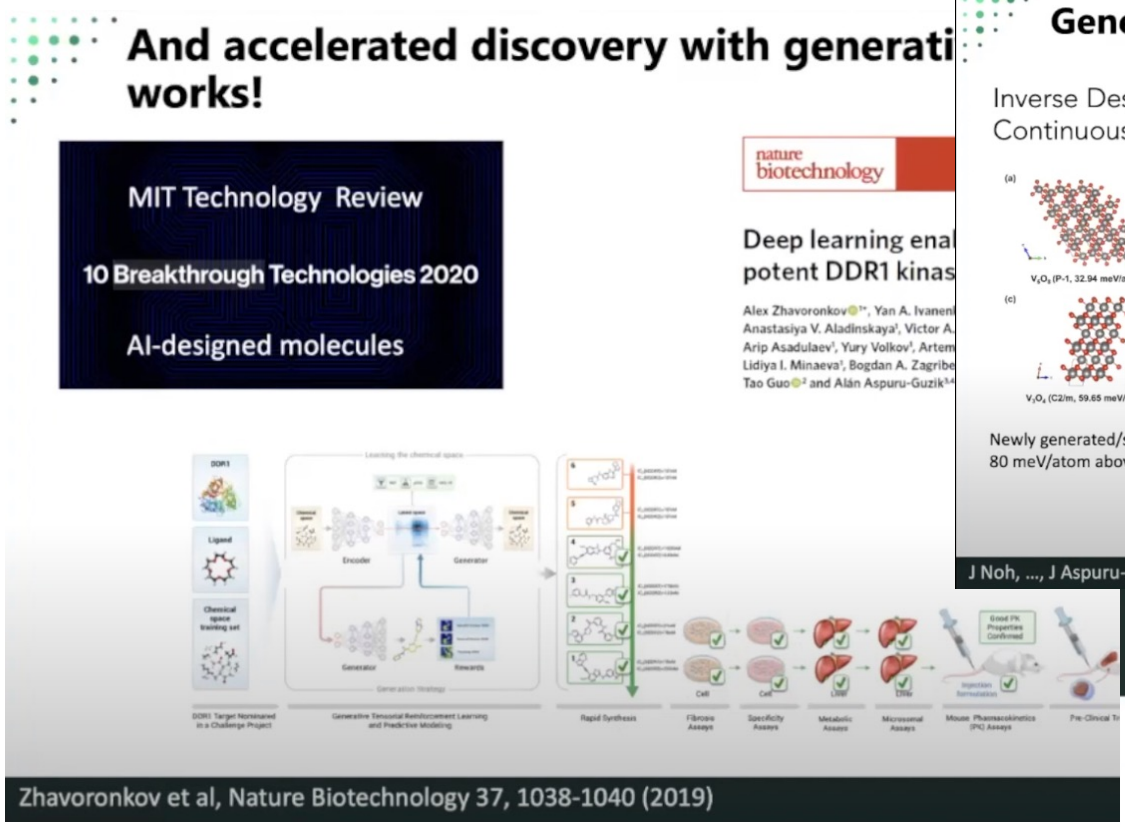
Closing the loop for material science

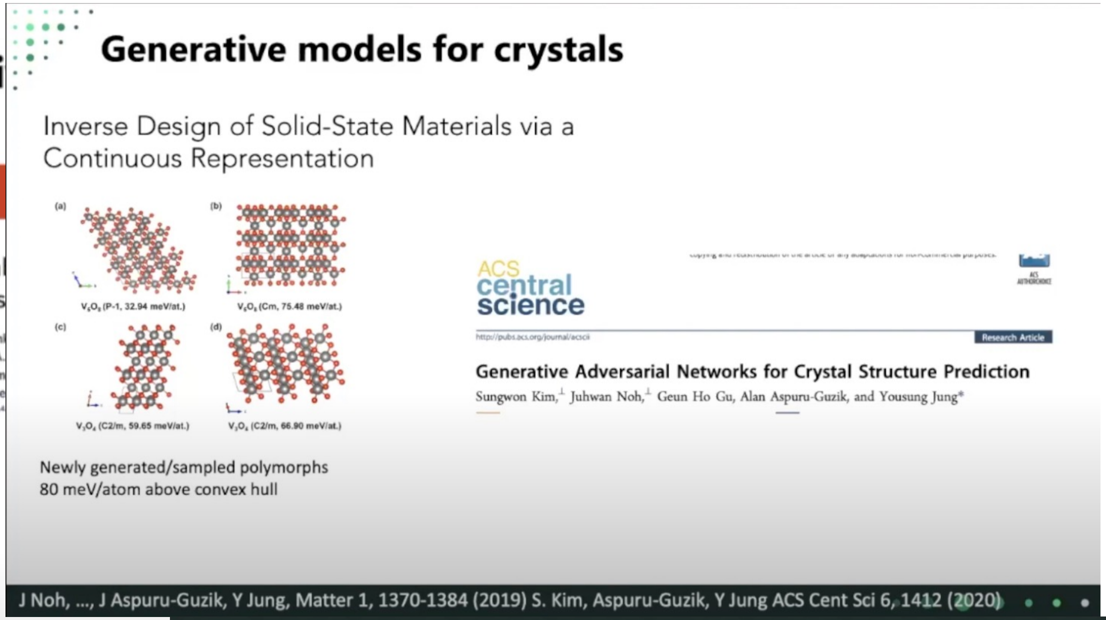


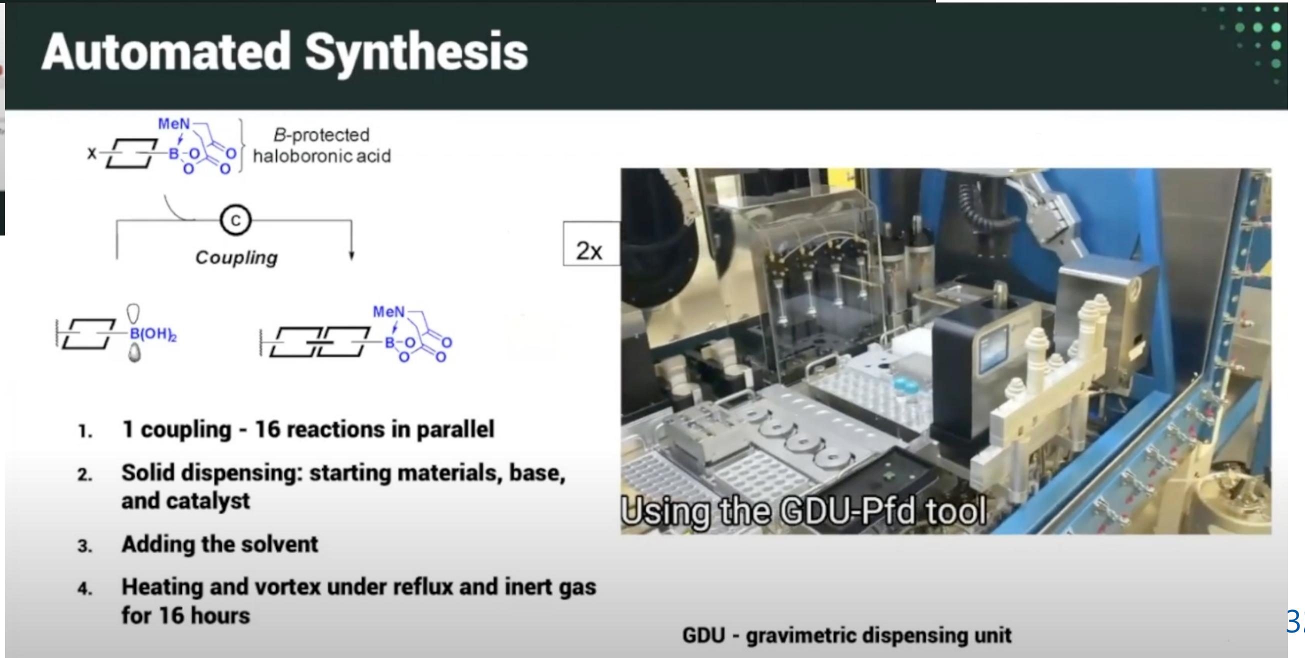
Sánchez-Lengeling and Aspuru-Guzik *Science 2018, 361*, 360.

https://www.youtube.com/watch?v=4WfG7_4B7mM

Self-driving labs, Alan Aspuru-Guzik







Fast model

Hardware experiment lab

Optimization / comparison

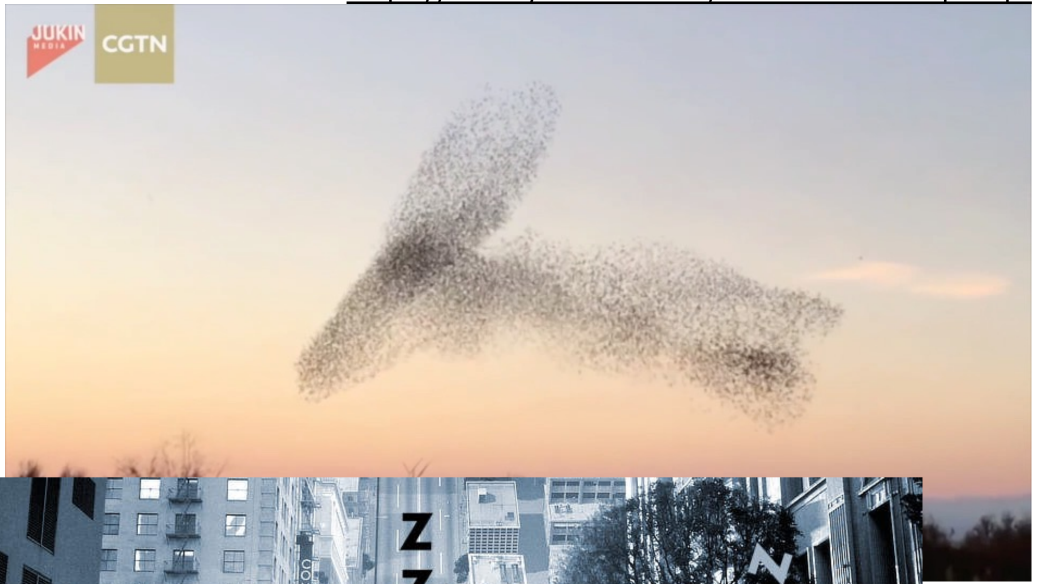
https://www.youtube.com/watch?v=4WfG7_4B7mM

Notable examples

- Robot Scientists, https://owncloud.gwdg.de/index.php/s/vZJiBu7PviP24i3
  - Adam (2009)
  - Eve
  - Genesis
- Kebotix, https://www.kebotix.com/
- Ada, https://www.science.org/doi/10.1126/sciadv.aaz8867
- KIWI, https://kiwi-biolab.de/

Why AI + Physics?

https://www.youtube.com/watch?v=0dskCpuxqtI



atwentwrongwith.com/2020/07/23/what-went-wrong-with-inception-2010/

Challenges for describing complex systems

- Emergence
- Dark matter search
- Quantum vs classical gravity
- Self-driving research

https://www.artstation.com/artwork/AqawDV

Open questions

- Which known scientific paradigms/frameworks can be helpful for building interpretable forward models?
  - Construction theory, Gauge theory, Renormalization theory, Quantum field theory?
- Can we leverage different computation model to speed-up inverse model design?
  - Quantum computing
- Can inverse models be developed on top of trained forward models?
- To what extend a system semantic description can be helpful?
  - How to merge semantic descriptions with differentiable optimization routines?
- How useful can multi-scale hybrid simulation be?

Towards Physics-enabled AI

Governing frameworks

- Scale-invariance via emergent properties identification, Barnett L, Seth A https://arxiv.org/abs/2106.06511
- Constructor theory (Deutsch, David. “Constructor theory.” Synthese 190 (2013): 4331-4359, Chiara Marleto, https://arxiv.org/pdf/1608.02625.pdf)

Building blocks

- Hypothesis generative model
- Inductive Bias Library for representation and forward model
- Dynamic Representation Learning
  - Emergent properties detection / analysis
- Simulation-Exploration cycle / active learning
  - Model ensembling
- Inverse model construction / interpretation
- Hypothesis testing / verification

Conclusion

- New technologies demand new research approaches that can be dramatically improved with data-driven methods (machine learning)
- Interpretability is a tricky matter
  - Decision trees, Robustness tests, Generalization, Symbolic regression
- Physical principles aid towards complex system analysis
  - Emergent properties
  - Invariant search
  - Forward / inverse model construction / calibration
  - …
- Ultimate goal: self-driving research
  - Some progress is already made, but still a long road to go
- *Open for collaboration / internship*

anaderiRu
hse_lambda
austyuzhanin@hse.ru

https://indico.cern.ch/event/1082512/

Backup

Towards a Theory of Evolution as Multilevel Learning,

- P1.Loss function. In any evolving system, there exists a loss function of time-dependent variables that is minimized during evolution.
- P2.Hierarchy of scales. Evolving systems encompass multiple dynamical variables that change on different temporal scales (with different characteristic frequencies).
- P3.Frequency gaps. Dynamical variables are split among distinct levels of organization separated by sufficiently wide frequency gaps.
- P4. Renormalizability. Across the entire range of organization of evolving systems ,a statistical description of faster-changing (higher frequency) variables is feasible through the slower-changing (lower frequency) variables.
- P5.Extension.Evolving systems have the capacity to recruit additional variables that can be utilized to sustain the system and the ability to exclude variables that could destabilize the system.
- P6.Replication. In evolving systems, replication and elimination of the corresponding information processing units can take place on every level of organization.
- P7.Information flow. In evolving systems, slower-changing levels absorb information from faster-changing levels during learning and pass information down to the faster levels for prediction of the state of the environment and the system itself.

Vanchurin V, et al https://arxiv.org/abs/2110.14602

Symbolic regression problem statement

1. Given a dataset $\mathcal{D} = (X_d, Y_d), X_d \subset \mathbb{R}^m, Y_d \subset \mathbb{R}$

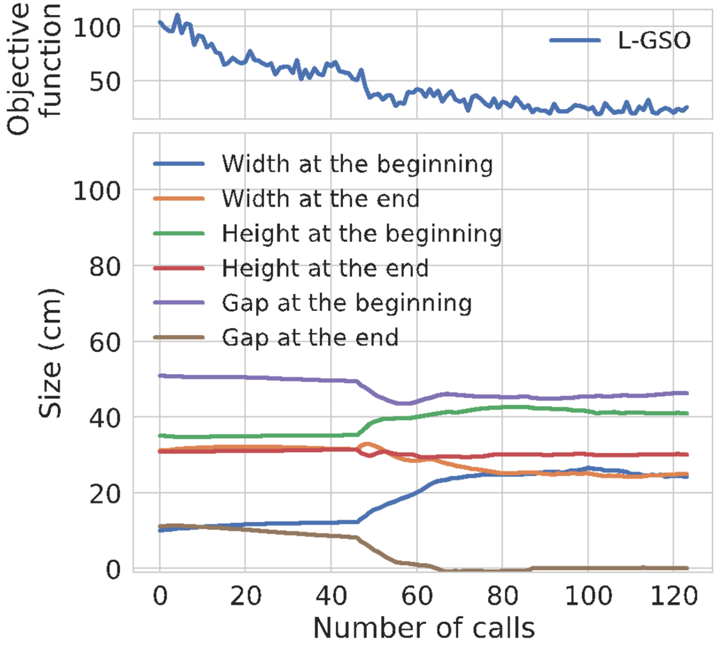
   Find a mathematical formula $y = f(x)$ approximating $\mathcal{D}$:

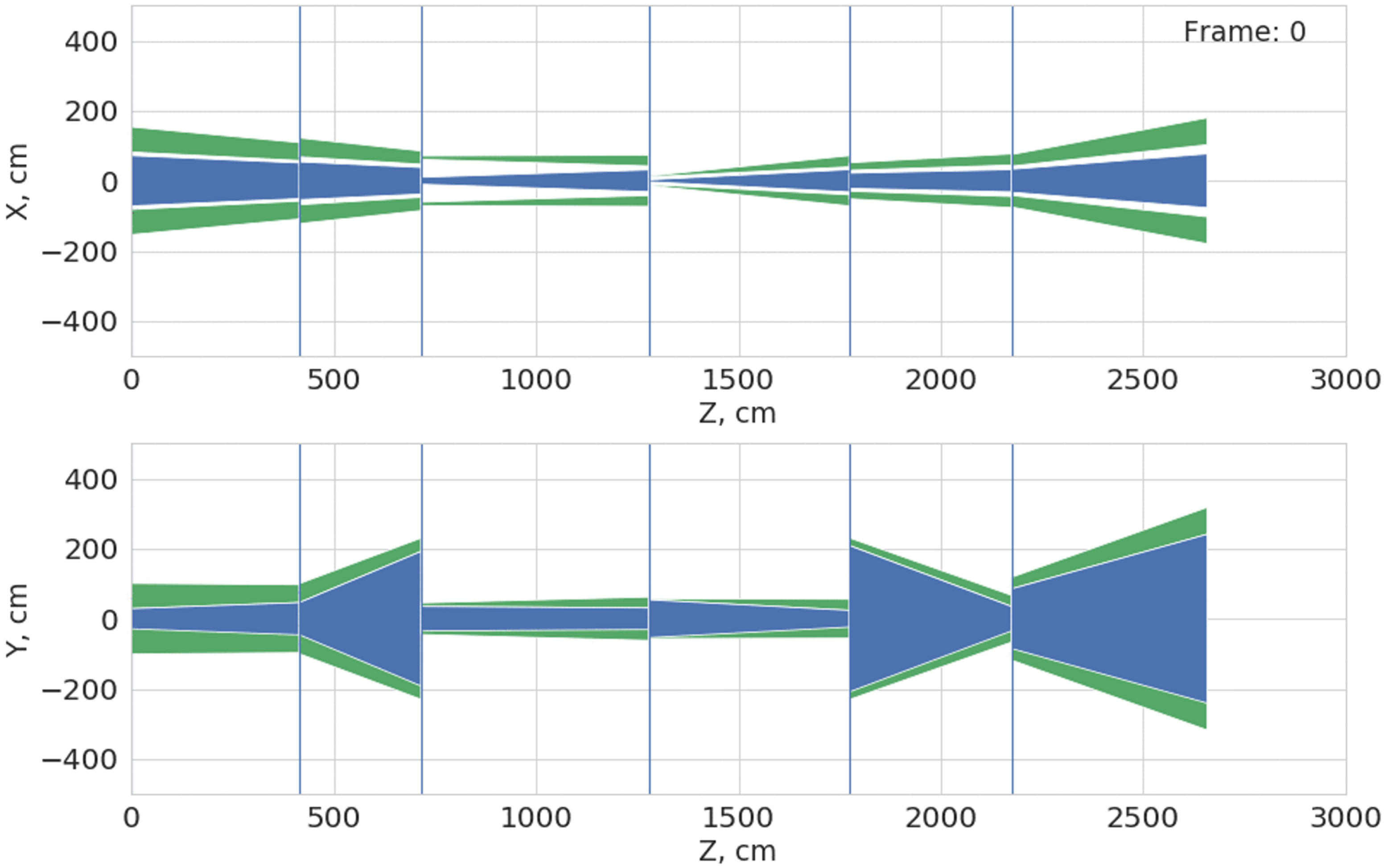
   $$\forall (x_i, y_i) \in \mathcal{D} : y_i \approx f(x_i)$$

2. Given a dataset $\mathcal{D} = (X_d, Y_d), X_d \subset \mathcal{X} \subset \mathbb{R}^m, Y_d \subset \mathcal{Y} \subset \mathbb{R}$

   Choose a point $x \in \mathcal{X}$ to add to the dataset with corresponding y

Design optimisation in 42 dimensional space of physics simulator



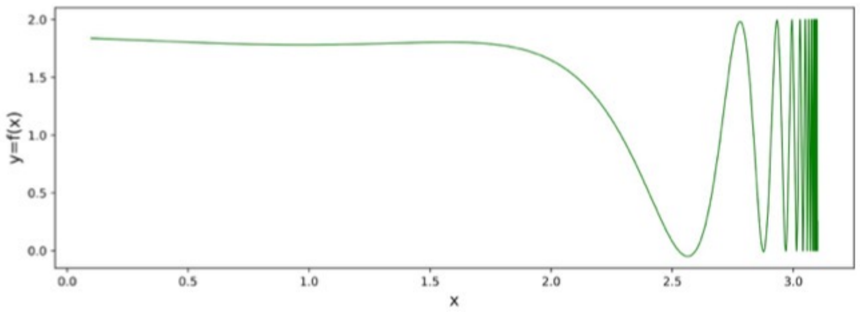


L-GSO improves previous results obtained with BO with the same computational budget.

New design is 25% more efficient.

Shirobokov S., Belavin V., Kagan M, AU, Baydin A., NeurIPS’20 paper
https://arxiv.org/abs/2002.04632

Active Learning: Next Points



$$y = \sin\left(\frac{x}{\sin(x)}\right) + \cos\left(\frac{\sin(x)}{x+\cos(x)}\right)$$