Scientifically-Interpretable Reasoning Network (ScIReN):

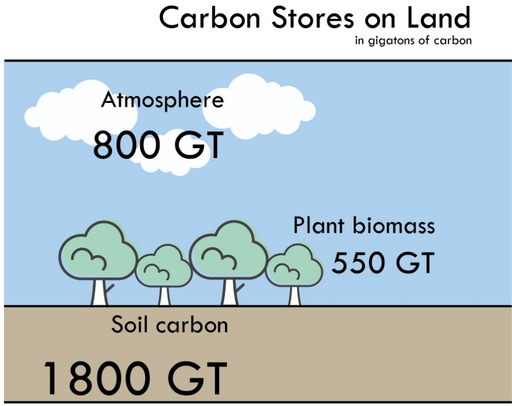
Discovering Hidden Relationships in the Carbon Cycle and Beyond

Joshua Fan[1*], Haodi Xu[2*], Feng Tao[3,4*], Md Nasim[1], Marc Grimson[1], Yiqi Luo[2], Carla P. Gomes[1]

[1]Cornell University, Computer Science [2]Cornell University, Soil & Crop Science
[3]Cornell University, Ecology and Evolutionary Biology
[4]Pennsylvania State University, Informatics and Intelligent Systems & Institute of Energy and the Environment
*Equal contribution

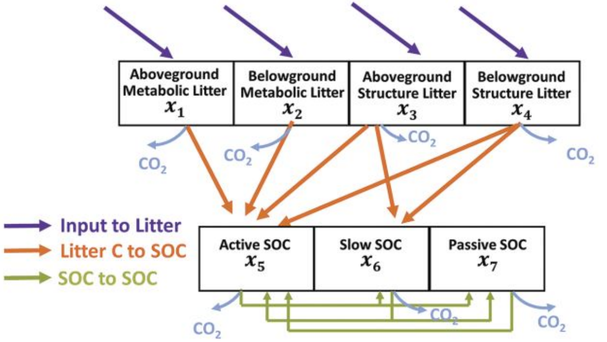
Motivation: Soil Carbon Cycle

- Soils store vast amounts of carbon
  - Potential to remove CO2 from the atmosphere
- However, the **soil carbon cycle** is poorly understood
  - e.g. how long does carbon stay in the soil?
- Based on prior knowledge, scientists develop **process-based models** to simulate how carbon flows through soil
  - Matrix equations enforcing mass conservation
  - For each pool: inputs = outputs



$$\frac{dx_6}{dt} = f_{63}k_3x_3 + f_{64}k_4x_4 + f_{65}k_5x_5 - k_6x_6$$

Change in pool x6 (assume 0) | Input from pool x3 | Input from pool x4 | Input from pool x5 | Output to other pools (+atmosphere)

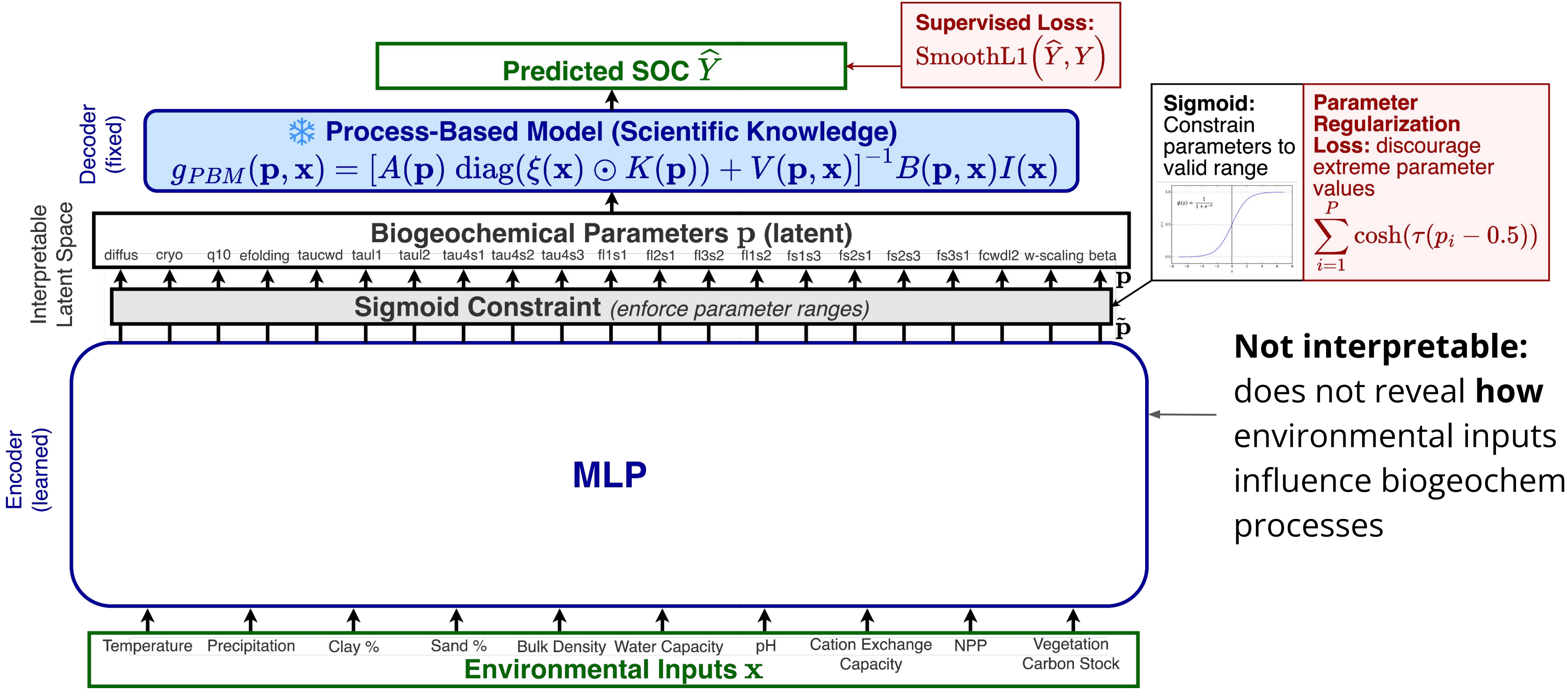


Many unobserved parameters (which vary across space)

Figure sources: (1) University of Wisconsin, https://cropsandsoils.extension.wisc.edu/articles/agricultural-carbon-credits-an-overview-for-farmers-and-landowners/
(2) Luo & Smith, Land Carbon Cycle Modeling textbook, 2nd ed.

Blackbox-Hybrid approaches (e.g. BINN)

- **Prior work:** embed differentiable process-based model inside neural network [1]
- MLP takes environmental inputs at each location, and predicts **latent parameters**
- Process-based model uses these parameters to simulate soil carbon flows and amounts

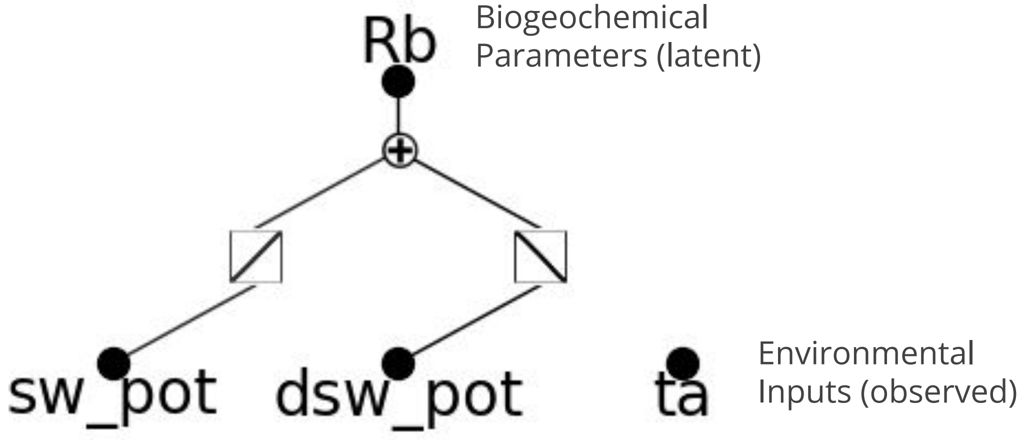


Not interpretable: does not reveal **how** environmental inputs influence biogeochemical processes

[1] Xu*, Fan*, Tao*, et al. "Biogeochemistry-Informed Neural Network (BINN) for Improving Accuracy of Model Prediction and Scientific Understanding of Soil Organic Carbon" In review (Geoscientific Model Development). https://arxiv.org/abs/2502.00672

Neural additive models + KANs

- Kolmogorov Arnold Network (KAN) is an alternative to MLPs that is easier to interpret (sometimes)
  - Learn activation function on **edges,** then add together at nodes
- Start with an **additive model**

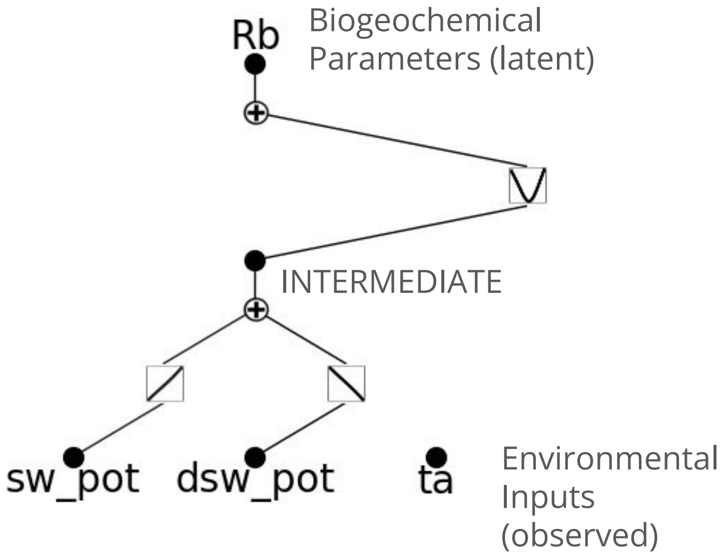


Here, the model estimates
Rb = f_1(sw_pot) + f_2(dsw_pot)

f_1, f_2 are learned from data; can be any function from 1 input → 1 output

Interpretation: As sw_pot increases, Rb increases. As dsw_pot increases, Rb decreases. Assumes contributions from each input are additive.

Example of 2-layer KAN



Here, the model estimates:

INTERMEDIATE = f_1(sw_pot) + f_2(dsw_pot)

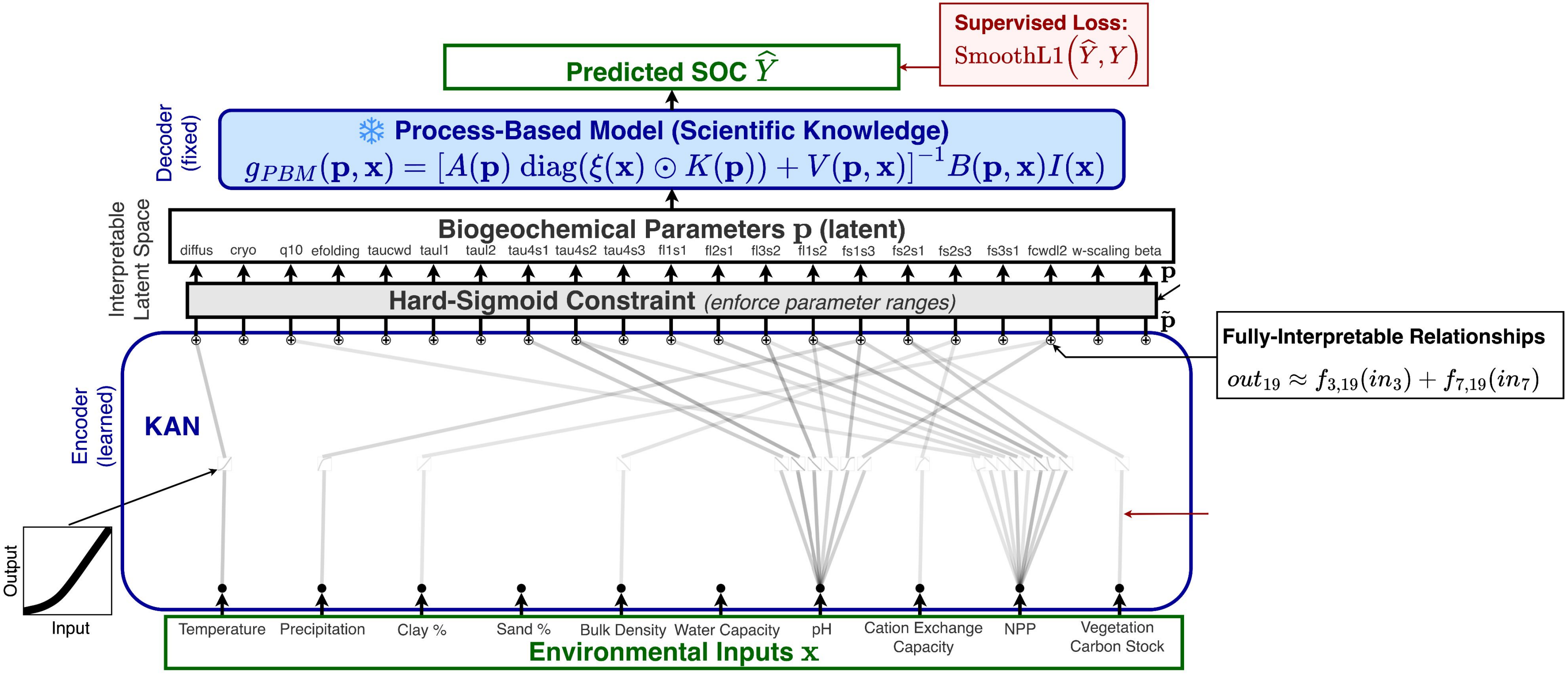
Rb = f_3(INTERMEDIATE)

To be interpretable, network should be:

- **Sparse:** only a small number of connections matter (relative to all possible connections)
- **Smooth splines:** relationships should be as “linear as possible” while fitting the data.

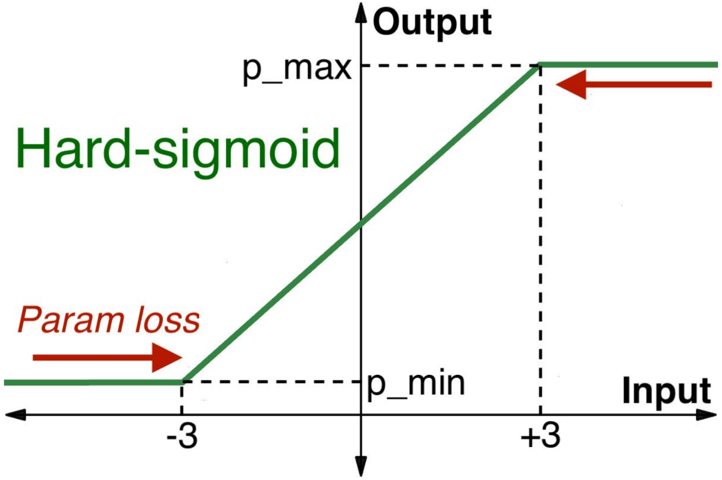
We design regularization losses to encourage these

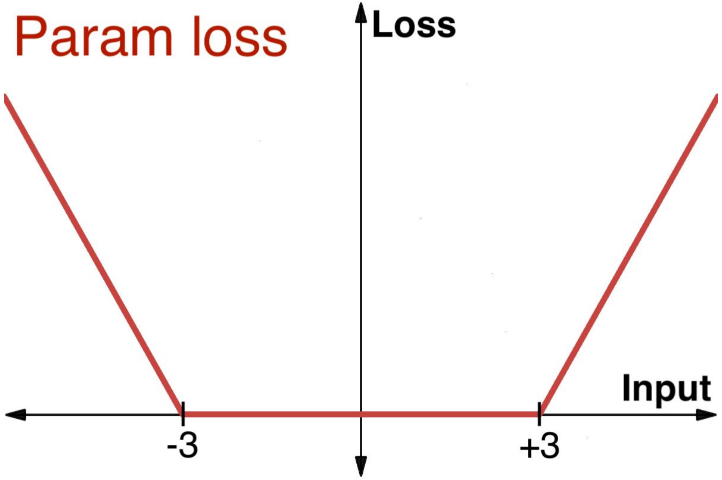
Scientifically-Interpretable Reasoning Network (ScIReN)



Hard-Sigmoid Constraint

- We often want to constrain parameters to be within a given prior range.
- In BINN, we did this with a sigmoid, but this adds nonlinearity, making functional relationships hard to interpret.
- Now, we use a hardsigmoid which clamps the model's prediction to be within the prior range. **Linear** within the prior range
- However, since the derivative is 0 outside the prior range, we add another loss to push predictions out of this flat area





Sparsity Loss: Details

- Compute **edge importance scores:** how much each edge contributes to variation in the final outputs
  - Use a "backpropagation-like" algorithm [1]
- Normalize → probability distribution over edges
- This should have **low entropy:** a small number of connections are important, others don't matter

$$b^l_{i,j} = \frac{B^l_{i,j}}{\sum_{i',j'} B^l_{i',j'}} \quad \text{(normalize edge importance to sum to 1)} \tag{5}$$

$$\mathcal{L}_{entropy} = -\sum_l \sum_{i,j} b^l_{i,j} \log b^l_{i,j};\ \mathcal{L}_{L1} = -\sum_l \sum_{i,j} |B^l_{i,j}| \tag{6}$$

Define $E_{l,i,j}$ as the mean absolute deviation[1]) of the outputs of the $(l, i \to j)$ edge (the edge from layer $l-1$, node i to layer l, node j):

$$E_{l,i,j} = \mathrm{AbsDev}(\phi_{l,i,j}(x_{l-1,i})) \tag{1}$$

Note that the mean absolute deviation is taken over the **batch** dimension.

Let $N_{l,j}$ be the mean absolute deviation of the outputs of node (l, j):

$$N_{l,j} = \mathrm{AbsDev}\left(\sum_{i=1}^{n_{l-1}} \phi_{l,i,j}(x_{l-1,i})\right) \tag{2}$$

We now compute node and edge scores iteratively. Start with last layer, and set output node scores $A_{L,i}$ to be the variance of output i. Then compute scores as follows for each layer $l = L, \ldots, 1$:[2]

$$B_{l-1,i,j} = A_{l,j} \frac{E_{l-1,i,j}}{N_{l,j}} \tag{3}$$

$$A_{l-1,i} = \sum_{j=0}^{n_l} B_{l-1,i,j} \tag{4}$$

Intuitively, $A_{l,j}$ represents how much neuron (l, j) contributes to the variance in all final outputs, and $B_{l,i,j}$ is how much of that variance is contributed by the output of edge $(l, i \to j)$. For the first equation, we first look at the contribution of neuron (l, j) towards the final variances, and then split it across the input edges according to the fraction of this neuron's variance contributed by each incoming edge ($\frac{E_{l-1,i,j}}{N_{l,j}}$). For the second equation, we compute each neuron's contribution towards the final variances by summing over the contributions via each *outgoing* edge.

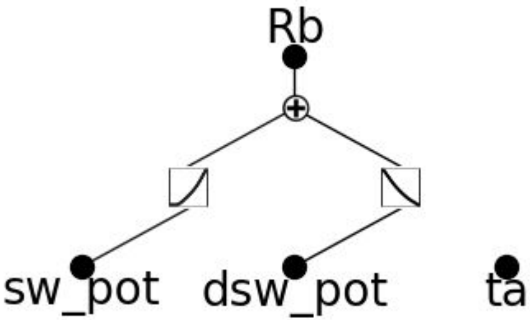
[1] Liu et al. (2024). Kan 2.0: Kolmogorov-arnold networks meet science. *arXiv preprint arXiv:2408.10205*.

Smoothness Loss

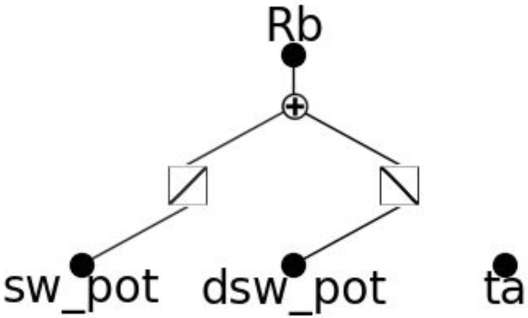
- Functions on each connection can be **any curve** (here: parameterized by B-splines)
- However, we add a “smoothness loss” (2nd derivative penalty) to encourage the curves to be close to linear if possible. Still allows for nonlinearity when needed

If $c_1 \ldots c_G$ are B-spline coefficients, the penalty is

$$\mathcal{L}_{smooth} = \sum_{i=1}^{G-2} ((c_{i+2} - c_{i+1}) - (c_{i+1} - c_i))^2$$



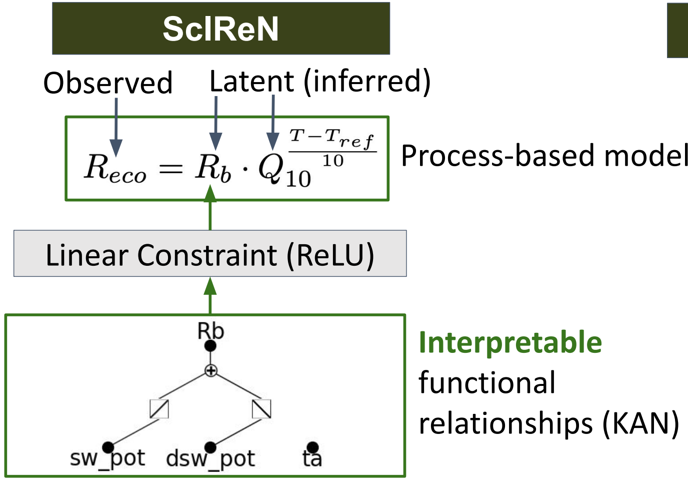
Some unnecessary complexity in the curves



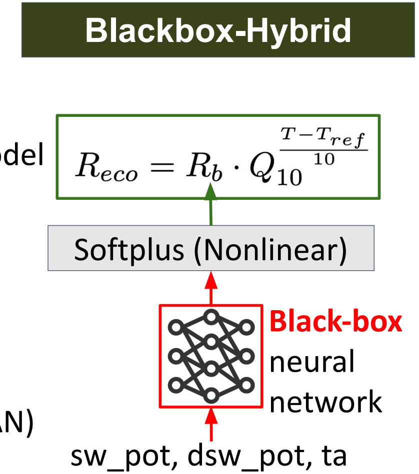
As linear as possible

Results: Ecosystem Respiration

- Latent variable “Rb” only depends on first two features
- ScIReN learned this correctly (sparsity/linearity); Blackbox-Hybrid did not. See paper for numbers



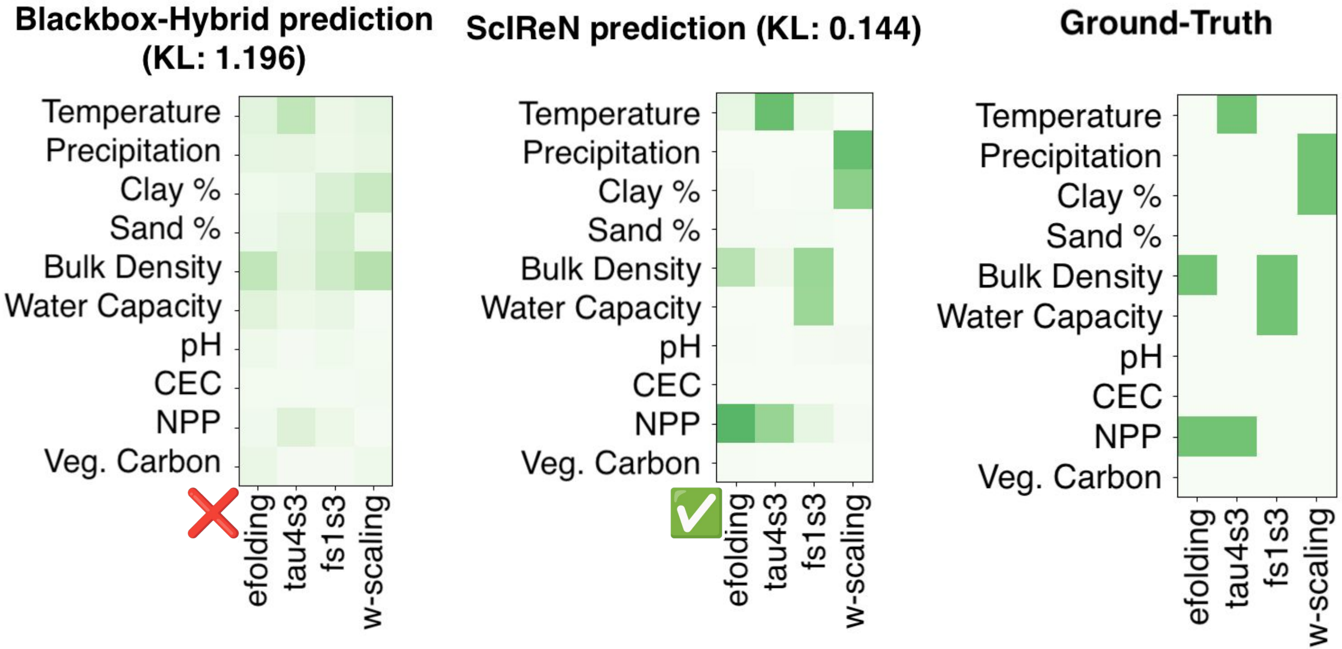
✅ Correct functional relationships
✅ Accurate latent parameters
✅ Extrapolates out-of-distribution



❌ No explicit functional relationships
❌ Wrong “implicit” functional relationships
❌ Inaccurate latent parameters

Results: Soil Carbon Cycle

- Generated synthetic dataset with known functional relationships (right)
- BINN (Blackbox-Hybrid) did not infer correct functional relationships (left), but ScIReN did (center).



- On real data, ScIReN achieves the same accuracy as black-box methods while being **fully-interpretable and transparent**. No need to sacrifice accuracy for interpretability!

Quantitative results

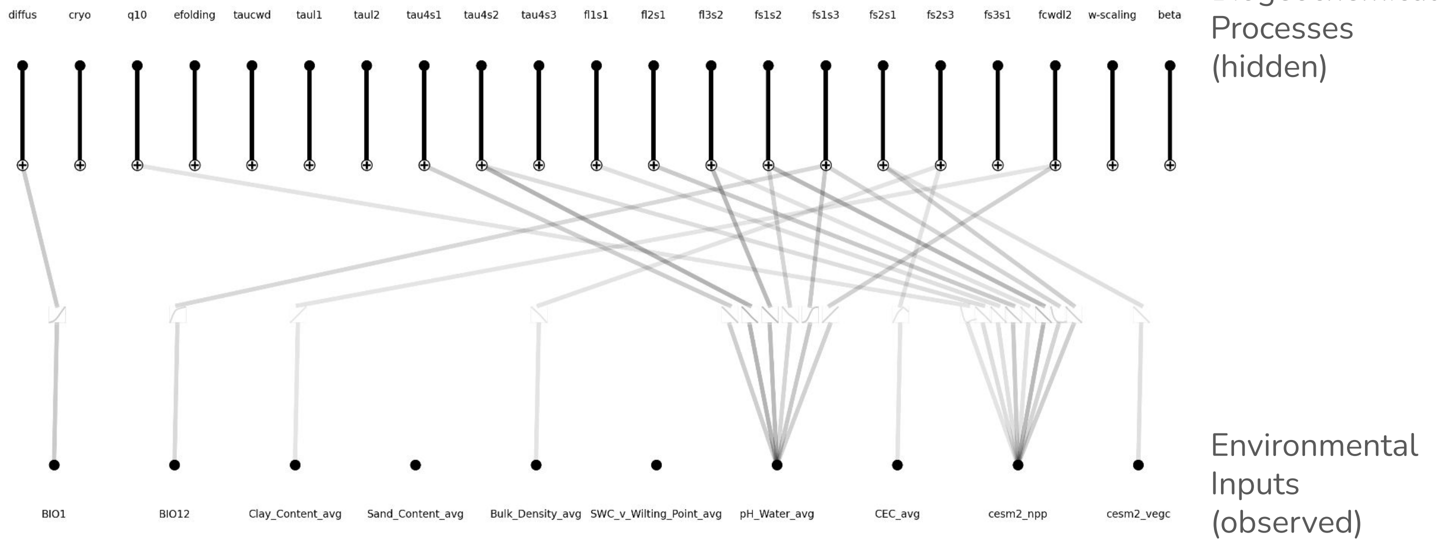
Synthetic labels: ScIReN is by far the best at recovering latent parameters and functional relationships

Method	R^2 (observed, ↑)	R^2 (latent, ↑)	KL, functional relationships (↓)
Pure-NN	0.933 ± 0.015	N/A	N/A
Blackbox-Hybrid, nonlinear constraint	0.996 ± 0.003	0.226 ± 0.800	1.312 ± 0.170
Blackbox-Hybrid, linear constraint	0.995 ± 0.003	0.721 ± 0.226	1.082 ± 0.258
Linear-Hybrid, hardsigmoid	0.973 ± 0.013	0.087 ± 1.014	1.727 ± 0.322
ScIReN, linear constraint (1-layer KAN)	$\mathbf{0.999} \pm 0.002$	$\mathbf{0.989} \pm 0.020$	$\mathbf{0.080} \pm 0.042$

Real labels: With ScIReN, we get **interpretability without sacrificing accuracy**. While we don't have ground-truth for latent parameters/relationships, they seem to match domain knowledge.

Method	R^2(↑)	MAE (↓)	Pearson correlation (↑)
Pure-NN	0.552 ± 0.173	$\mathbf{4609.3} \pm 356.8$	$\mathbf{0.780} \pm 0.053$
Blackbox-Hybrid, nonlinear constraint	0.584 ± 0.082	4726.2 ± 727.3	0.776 ± 0.048
Blackbox-Hybrid, linear constraint	$\mathbf{0.589} \pm \mathbf{0.070}$	4849.7 ± 650.3	0.774 ± 0.040
Linear-Hybrid, hardsigmoid	0.552 ± 0.082	4984.8 ± 771.6	0.761 ± 0.046
ScIReN, linear constraint (1-layer KAN)	0.582 ± 0.080	4708.2 ± 673.1	0.769 ± 0.049
ScIReN, linear constraint (2-layer KAN)	0.571 ± 0.094	4707.3 ± 826.3	0.765 ± 0.052

Qualitative feedback from soil scientists



- Still preliminary, but qualitatively these relationships seem consistent with ecological knowledge
- “We found a positive exponential-like relationship between mean annual temperature (BIO1) and diffusion rate (diffus) in vertical transport, suggesting that higher temperatures will accelerate the vertical movement of organic carbon. Such a relationship agrees well with the conventional understanding that higher temperatures provide more kinetic energy to support faster diffusion (Taylor 1938).
- Meanwhile, we found spreading negative relationships between fresh plant carbon input (NPP) and parameters related to carbon transfer efficiencies (f_ij) and SOC substrate baseline turnover times (tau_i). These emerging functional relationships support a positive long-term priming effect at the continental scale, where higher rates of plant carbon input will likely lead to accelerated SOC decomposition (lower tau_i) and eventually less SOC accrual (lower f_ij) (Kuzyakov 2010)

Conclusion

We propose ScIReN, a method that

- Respects **existing scientific knowledge**, provided by any process-based model
- Reveals **new functional relationships** between environmental inputs and **unobserved biogeochemical processes**

The system is trainable end-to-end, and every part of the model is **fully transparent**.

Potential future directions:

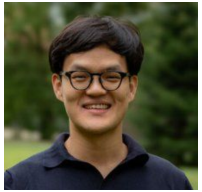
- Apply ScIReN to new domains
- Make ScIReN easier to train
- Understand **uncertainty** of revealed functional relationships
- Improve spatial generalization, e.g. geographic positional embeddings or domain adaptation

Thank you!

Paper link: https://arxiv.org/abs/2506.14054
(or Google "Scientifically-Interpretable Reasoning Network")

Email: jyf6@cornell.edu


Haodi Xu


Feng Tao


Md Nasim

Acknowledgements

SCHMIDT FUTURES
Eric and Wendy Schmidt
AI in Science Postdoctoral Fellowship,
a program of SCHMIDT FUTURES

Cornell Bowers CIS
College of Computing
and Information Science

NCAR CISL

AI 4 Sustainability

Cornell CALS
College of Agriculture and Life Sciences

U.S. National Science Foundation

Institute for Computational Sustainability ICS

ECOLAB OF DR. YIQI LUO

USDA National Institute of Food and Agriculture
U.S. DEPARTMENT OF AGRICULTURE

AI-LEAF


Marc Grimson


Yiqi Luo


Carla Gomes