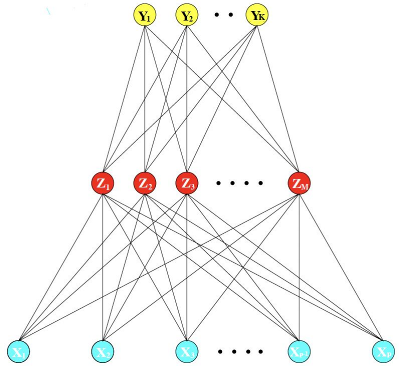
A deep dive into Physics-Guided Deep Learning

Tian Zheng

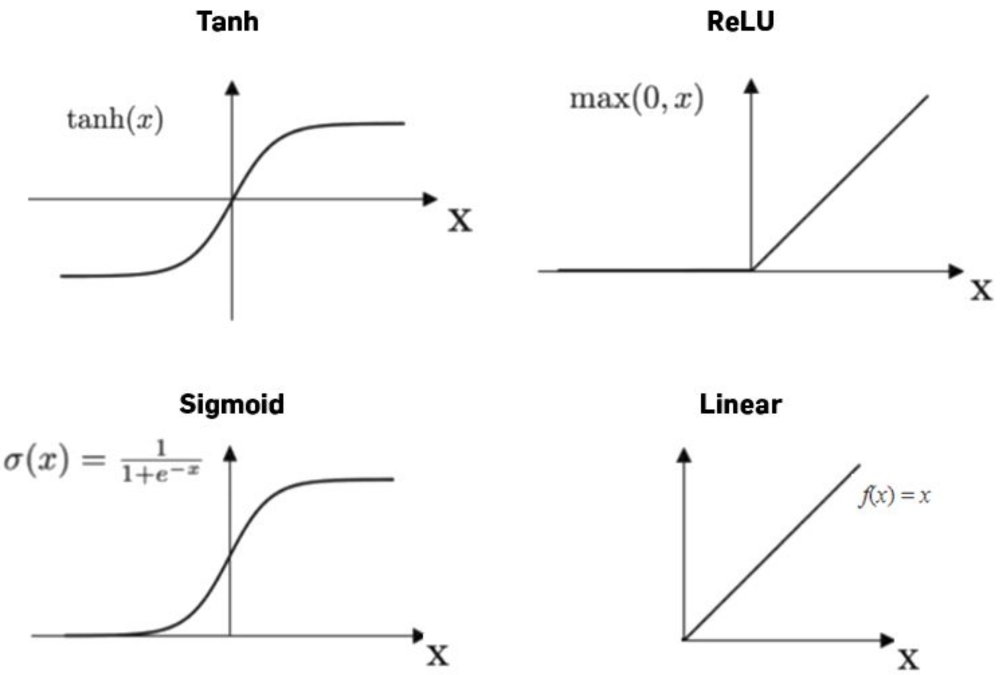
Read, J. S., Jia, X., Willard, J., Appling, A. P., Zwart, J. A., Oliver, S. K., et al. (2019). Process-guided deep learning predictions of lake water temperature. Water Resources Research, 55, 9173–9190.

Deeping Learning: the simplest setup

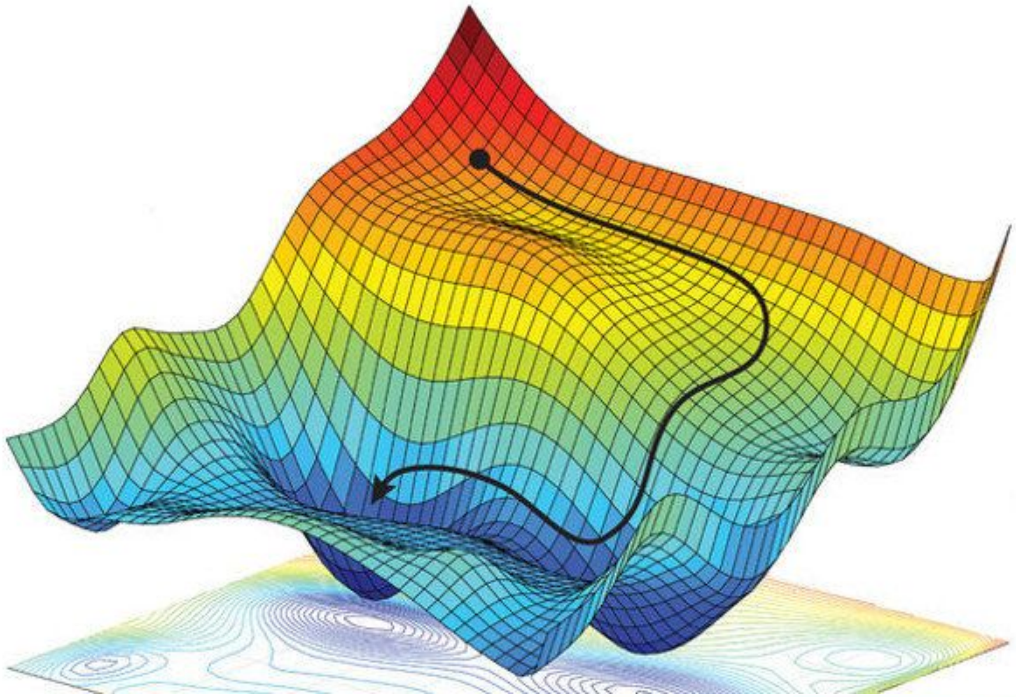
- Outputs (targets): $Y_1, \ldots, Y_K$.
- Inputs: $X_1, \ldots, X_p$.
- Hidden nodes: $Z_1, \ldots, Z_M$.
- Model ("vanilla" neural net):
  - $$\begin{aligned} Z_m &= \sigma(\alpha_{0m} + \alpha_m^T X), m = 1, \ldots, M, \\ T_k &= \beta_{0k} + \beta_k^T Z, k = 1, \ldots, K, \\ \mathrm{E}(Y_k|X) = f_k(X) &= g_k(T), k = 1, \ldots, K. \end{aligned}$$
  - Single *hidden layer*.



Activation functions



Fitting neural networks - Gradient Descent



Fitting neural networks - backpropagation

$$R_i(\theta) = \frac{1}{2}\Big(y_i - \beta_0 - \sum_{k=1}^{K} \beta_k g\big(w_{k0} + \sum_{j=1}^{p} w_{kj} x_{ij}\big)\Big)^2. \qquad (10.28)$$

$$z_{ik} = w_{k0} + \textstyle\sum_{j=1}^{p} w_{kj} x_{ij}.$$

$$\begin{aligned}\frac{\partial R_i(\theta)}{\partial w_{kj}} &= \frac{\partial R_i(\theta)}{\partial f_\theta(x_i)} \cdot \frac{\partial f_\theta(x_i)}{\partial g(z_{ik})} \cdot \frac{\partial g(z_{ik})}{\partial z_{ik}} \cdot \frac{\partial z_{ik}}{\partial w_{kj}} \\ &= -(y_i - f_\theta(x_i)) \cdot \beta_k \cdot g'(z_{ik}) \cdot x_{ij}. \qquad (10.30)\end{aligned}$$

Fitting neural networks - backpropagation

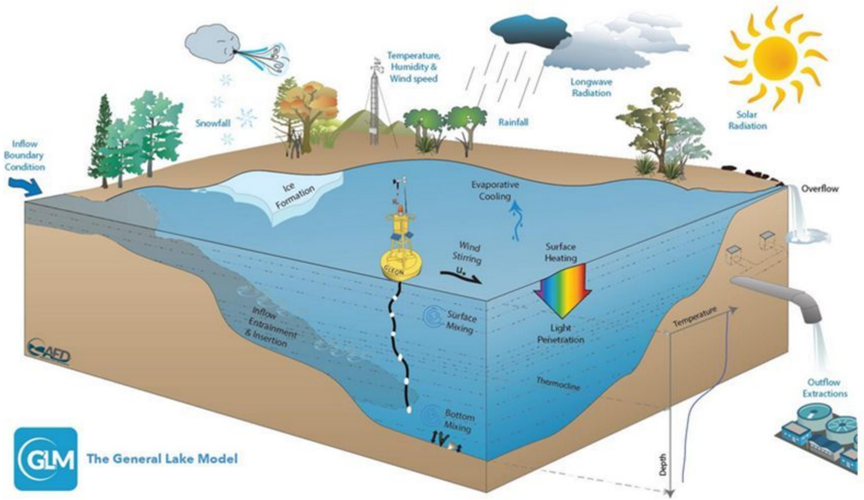
- Features
- Hidden layers
- Learning rate
- Regularization (L1, L2), regularization rate
- Batch size

https://playground.tensorflow.org/

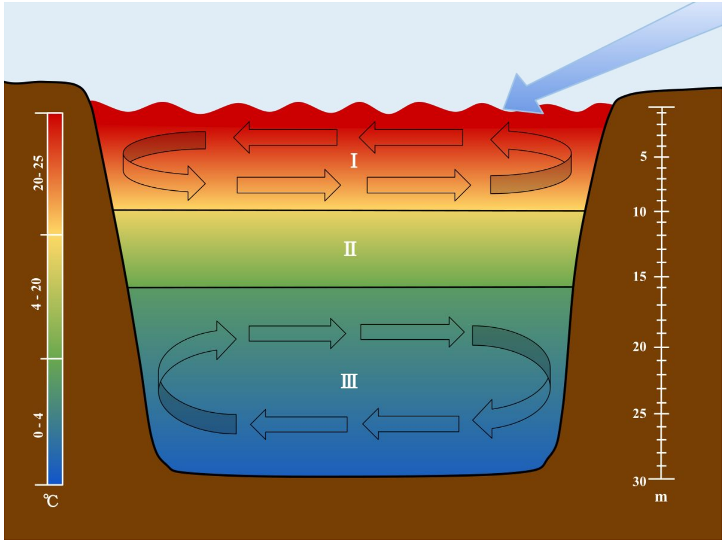
Lake Temperature Prediction

Accurate prediction of lake water temperature requires that models incorporate **temperature changes from prevailing weather conditions** while also reproducing features resulting from the presence, absence, strength, and duration of **thermal stratification**, including differing dynamics of surface and bottom waters.

Generic Lake Model (GLM)



1

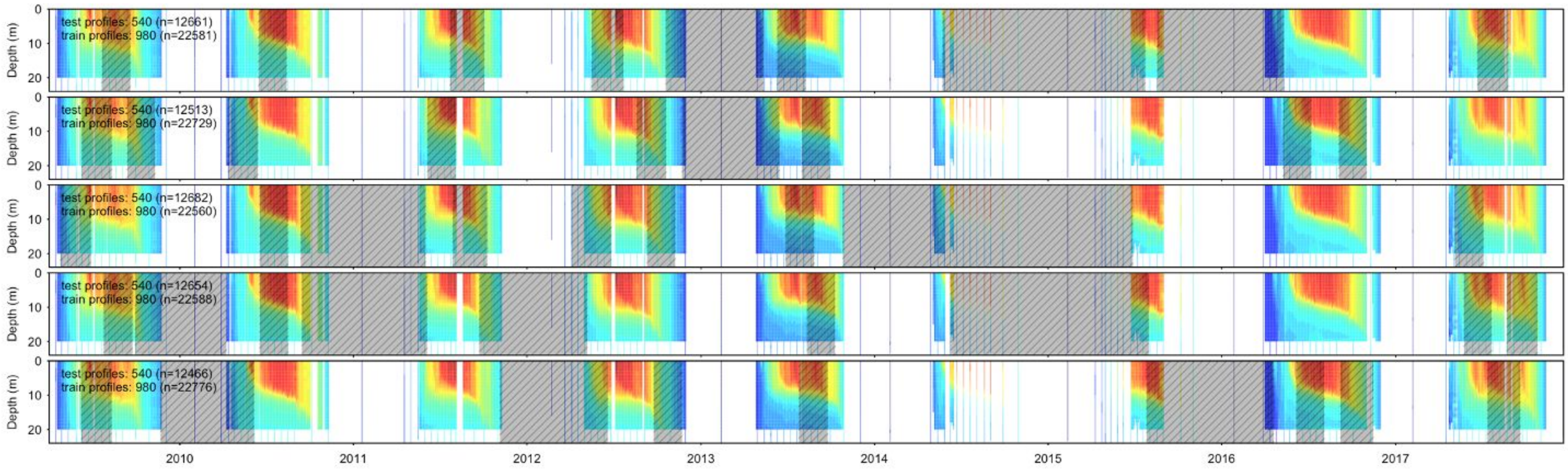


LEAP

Understanding LSTM

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Sparse observed spatio-temporal data



LEAP

http://colah.github.io/posts/2015-08-Understanding-LSTMs/

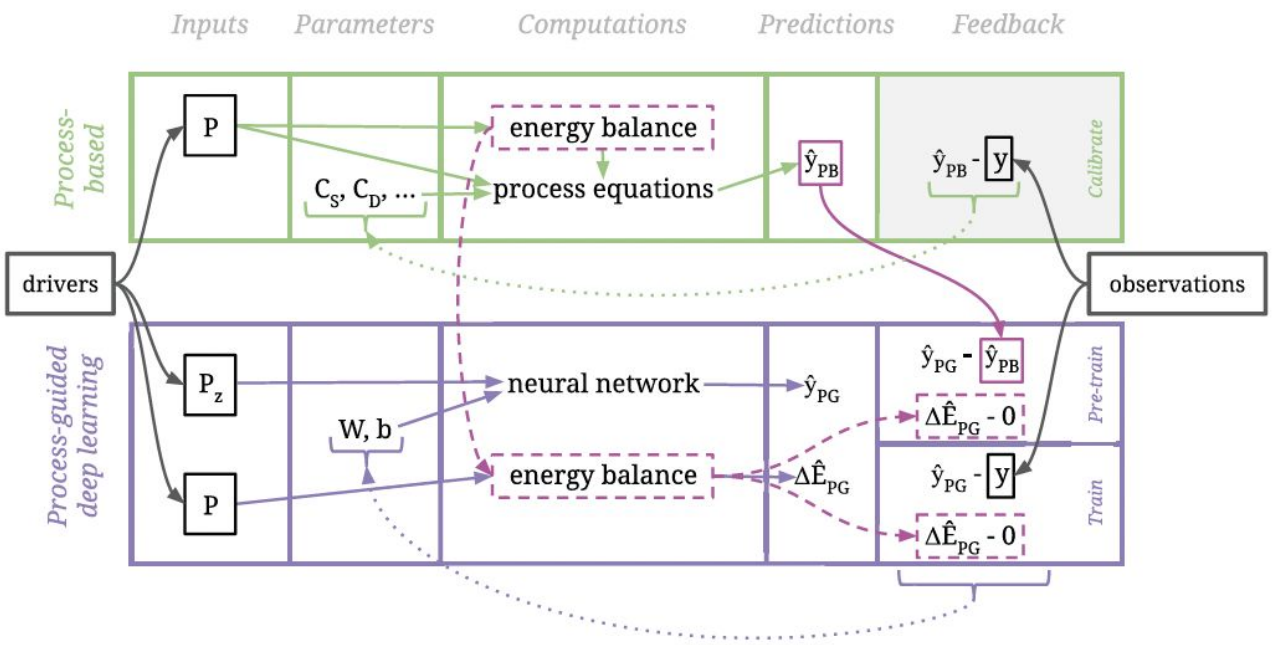


Figure 1. Comparison between process-based model (PB) and process-guided deep learning model (PGDL). Conceptual links between the two are in pink, showing the integration of the energy balance concept (dashed pink lines) and process-model-generated pretraining data (solid pink line) from PB into PGDL. Both models accept data in the form of drivers and observations (black lines; P = predictors, P_z = normalized predictors, y = temperature). Although the models differ greatly in their structures, they have in common that they accept the same raw inputs, use parameters and computations to generate predictions (green and purple solid lines), and revise the parameters based on feedback (called "calibration" for PB or "pretraining" and "training" for PGDL; green and purple dotted lines). PB calibration (gray box and green dotted line) is used for calibrated PB models but is omitted for the experiments in this manuscript when generating uncalibrated predictions ($\hat{y}_{PB}$) for use in PGDL pretraining.

Algorithm evaluation