Transfer learning applications in hydrologic modeling

Joseph Hamman and Andrew Bennett

EGU 2020

UNIVERSITY of WASHINGTON

NCAR
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH

Abstract

Early work in the field of Machine Learning (ML) for hydrologic prediction is showing significant potential. Indeed, it has provided important and measurable advances toward prediction in ungauged basins (PUB). At the same time, it has motivated a new research targeting important ML topics such as uncertainty attribution and physical constraints. It has also brought into question how to best harness the wide variety of climatic and hydrologic data available today. In this work, we present initial results employing transfer learning to combine information about meteorology, streamflow, surface fluxes (FluxNet), and snow (SNOTEL) into a state of the art ML-based hydrologic model. Specifically, we will present early work demonstrating how relatively simple implementations of transfer learning can be used to enhance predictions of streamflow by transferring learning from flux and snow station observations to the watershed scale. Our work is shown to extend recently published results from Kratzert et al. (2018) using the CAMELS data set (Newman et al. 2014) for streamflow prediction in North America.

- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., and Herrnegger, M.: Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks, Hydrol. Earth Syst. Sci., 22, 6005-6022, https://doi.org/10.5194/hess-22-6005-2018, 2018a.
- Newman; K. Sampson; M. P. Clark; A. Bock; R. J. Viger; D. Blodgett, 2014. A large-sample watershed-scale hydrometeorological dataset for the contiguous USA. Boulder, CO: UCAR/NCAR. https://dx.doi.org/10.5065/D6MW2F4D

Hello

- I am a Research Scientist in the Terrestrial Sciences Section at the National Center for Atmospheric Research (NCAR).
- My research focuses on emerging data science approaches in the climate and hydrologic modeling space.
- I help lead the Pangeo Project.
- I contribute to open source scientific Python projects like Xarray, Dask, and Jupyter.

jhamman

HammanHydro

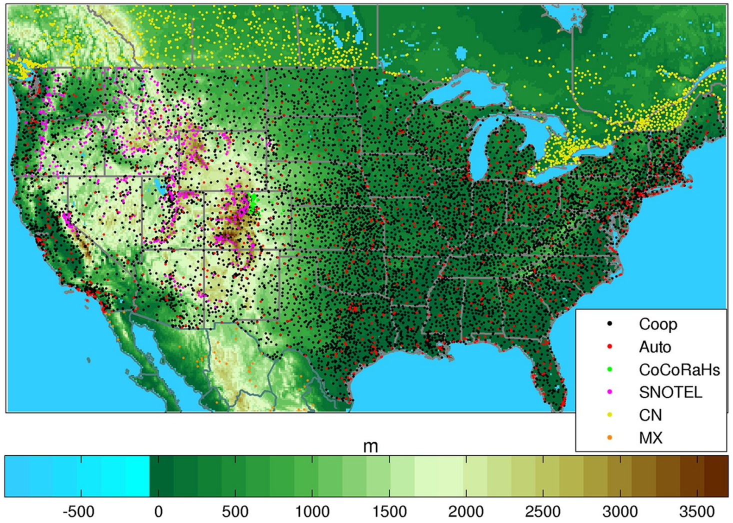
@jhamman

NCAR
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH

The sparse data problem

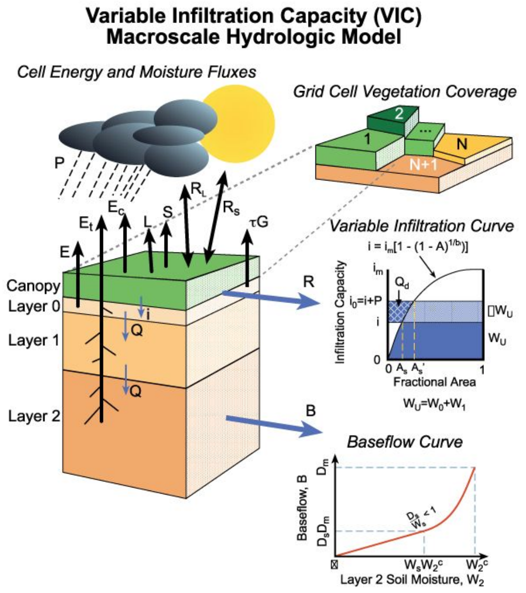
- We only measure meteorological quantities in a few places
- Some quantities are relatively common (e.g. precipitation and temperature)
- While others are very sparse (e.g. shortwave/longwave radiation)

- BUT we want evaluate our hydrologic model everywhere



North American meteorological stations. Note the sparse distribution in the Western US. Figure from Newman et al., 2015.

The sparse data problem

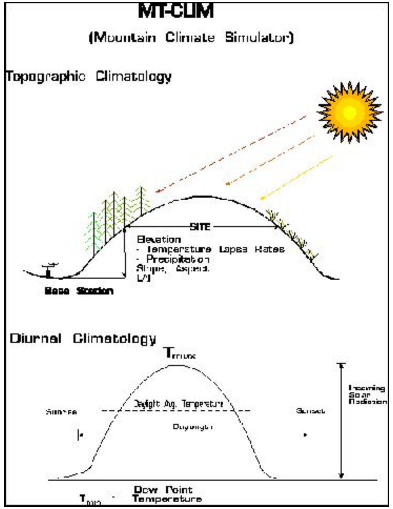


An example processed based model, [VIC](VIC). VIC requires "the big 7" forcing variables.

- Our hydrologic models have long assumed we have some knowledge of the fluxes at the land surface
- Many "processed based" models require sub-daily forcings ("the big 7"):
    - Precipitation
    - Temperature
    - Shortwave Radiation
    - Longwave Radiation
    - Humidity
    - Pressure
    - Wind Speed

An old solution

- Many approaches have been developed to close the sparse problem.
- Statistical approaches tend to develop empirical relationships between meteorological variables, e.g.:
  - MTCLIM, Thorton & Running, 1997
  - DAYMET, Thorton et al. 2014
  - GridMET, Abatzoglou 2013
  - ...



You know it's been around when this is the best resolution figure you can find on the [internet](). 😉

`Precip, Tmin, Tmax`

↓

`Black Box`

↓

`SW, LW, RH, Pres`

This workflow is employed in popular ML applications

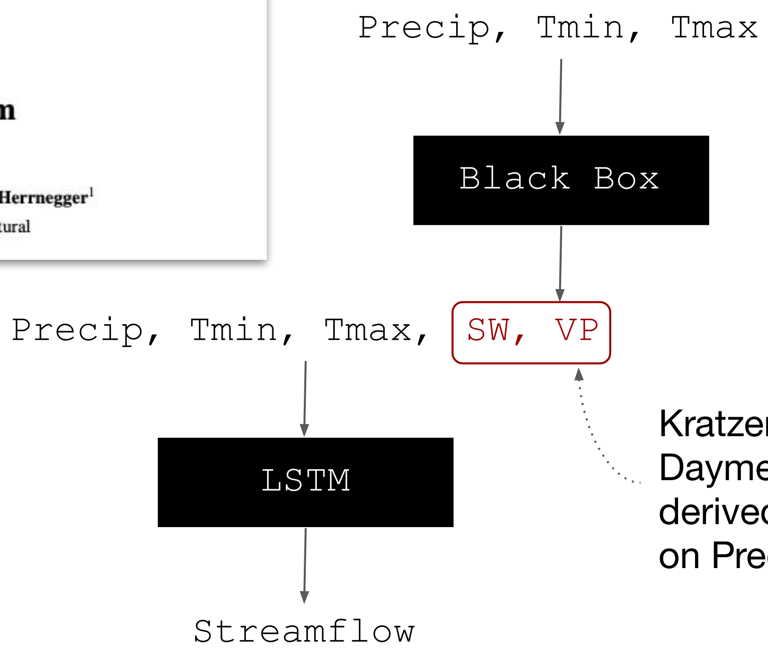
Hydrol. Earth Syst. Sci., 22, 6005–6022, 2018
https://doi.org/10.5194/hess-22-6005-2018

Hydrology and Earth System Sciences
Open Access
EGU

Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks

Frederik Kratzert[1,*], Daniel Klotz[1], Claire Brenner[1], Karsten Schulz[1], and Mathew Herrnegger[1]

[1]Institute of Water Management, Hydrology and Hydraulic Engineering, University of Natural Resources and Life Sciences, Vienna, 1190, Austria



Kratzert et al. (2018) used Daymet. SW, VP are derived quantities based on Precip, Tmin, Tmax.

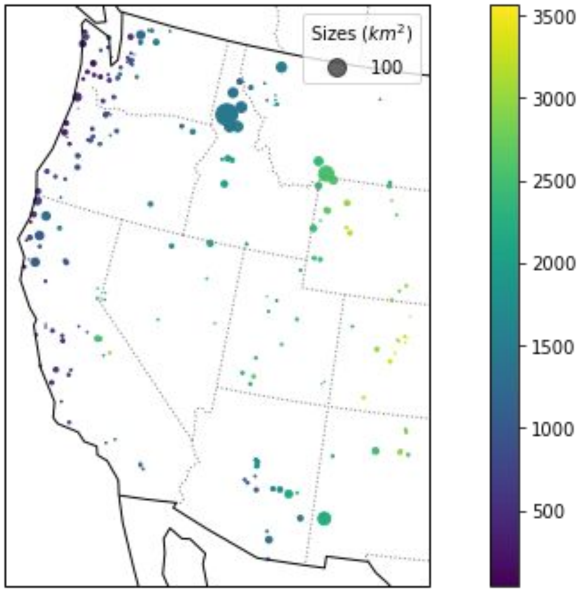
Study Questions

1. Does training with derived variables (e.g. VP, SW as in Kratzert et al. 2018) contribute to additional model skill?
2. Can we encode the “derivation” inside the ML model itself, enabling transfer learning from disparate data sources?

Part 1: Setup

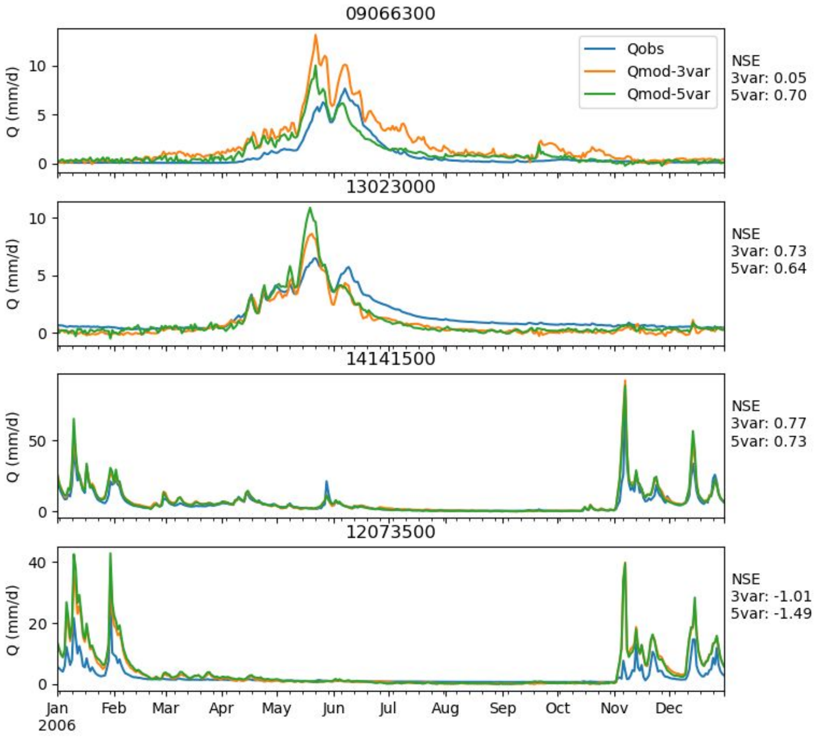
- Data: Camels NA
  - All basins W of -105°W
  - 126 (50%) training, 42 (25%) validation, 42 (25%) holdout
  - Same preprocessing as in Kratzert et al. 2018

- Two Simple LSTMs:
  1. "3var": `Q = LSTM(Precip, Tmin, Tmax)`
  2. "5var": `Q = LSTM(Precip, Tmin, Tmax, VP, SW)` [same as Kratzert]



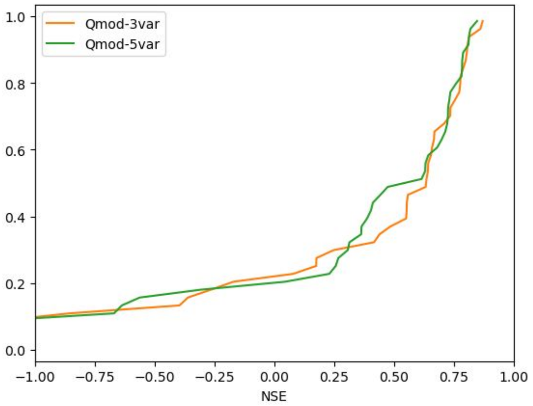
Camels NA stations used in this study. Colors represent mean basin elevation (m) and circle sizes represent the size of the basin.

Part 1: Results



(Left) Hydrographs for 4 random out-of-sample basins.

(Right): CDF of the nash-sutcliffe efficiency (NSE) for the 42 holdout basins.



Including Daymet shortwave radiation and vapor pressure does not meaningfully improve model performance.

NSE is not the only metric we should explore. What about transferability, stability, and physical consistency. (Needs more work)

Part 2: Setup

Inputs: Precip, Tmin, Tmax

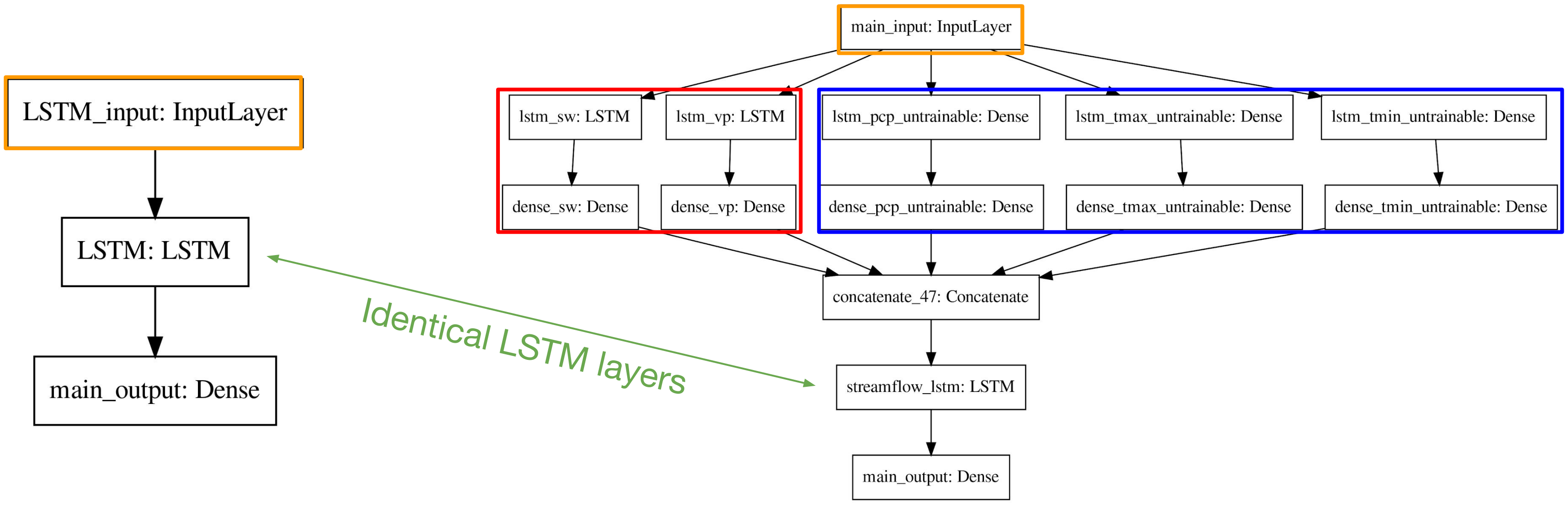
Frozen “pass through” layers

Frozen “pre-trained” layers

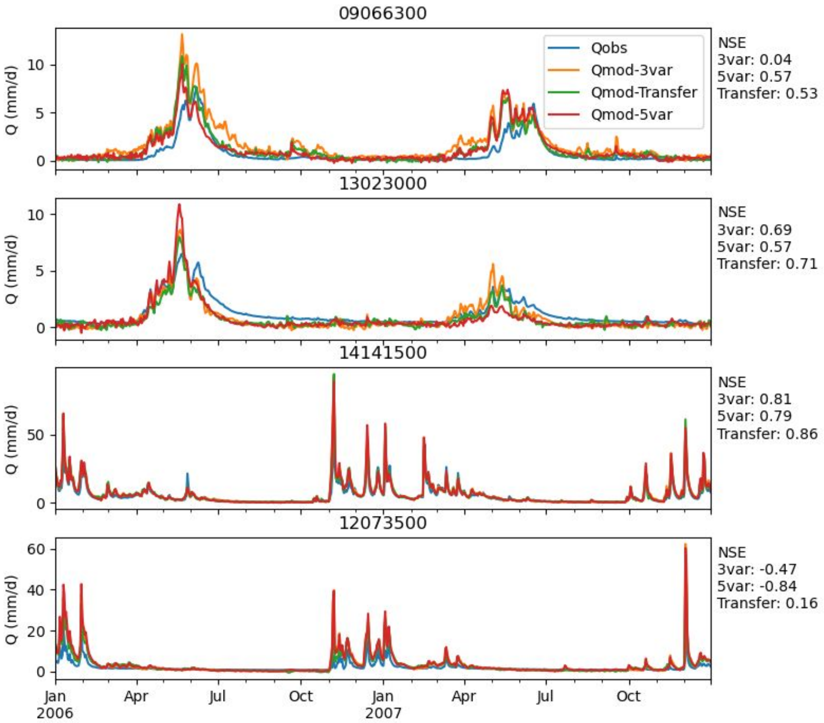
- Two Neural Nets, “3var” (Part 1) and a Transfer Learning configuration

3 or 5-var

Transfer Learning

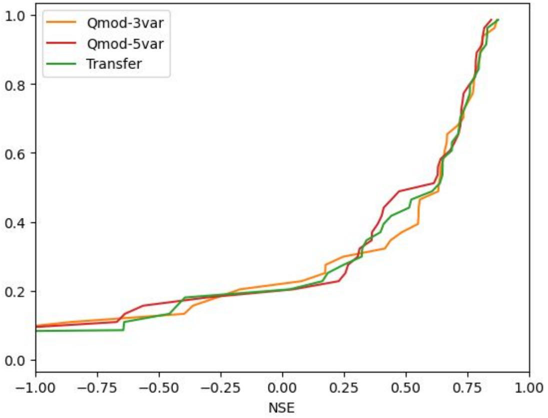


Part 2: Results



(Left) Hydrographs for 4 random out-of-sample basins.

(Right): CDF of the nash-sutcliffe (NSE) efficiency for the 42 holdout basins.



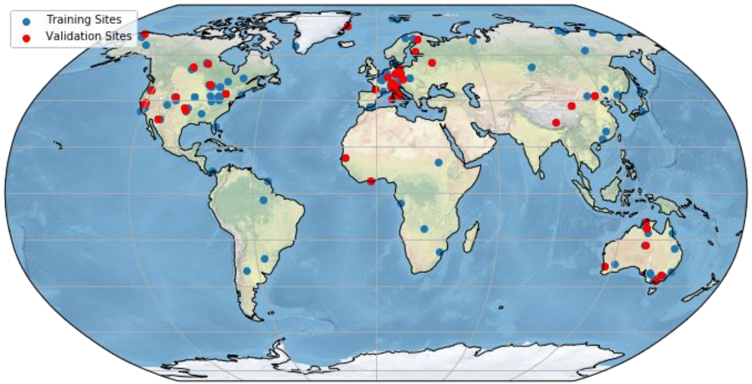
In most basins, transfer learning leads to modest improvements in NSE.

Main takeaway points:

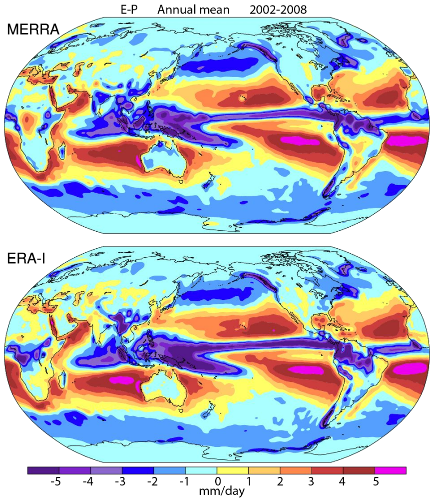
- It is possible to “wire” a transfer learning workflow into the Kratzert LSTM.
- More research is needed to understand why some basins do better without additional data?

Pre-training with model data

- Few high-quality flux tower observations exist (~200 long term Fluxnet sites)
- Strong sample bias toward NA and Europe
- Could we use global reanalysis to pre-train the meteorological parts (SW, LW, VP, Pres.) of our transfer learning model before training (tuning) on the flux tower data?



https://github.com/jhamman/met-ml



Next steps and conclusions

The work shown here is in very early stages. Looking for feedback and suggestions…

1. Our transfer learning models have, in total, more tunable parameters. What is the best way to normalize for model complexity?
2. Part 2 was still a bit of a toy example. We need to develop a holistic strategy for transferring information from various data sources & scales.
3. Toward compositional ML for hydrology -- what are the ResNet/ImageNet equivalents for streamflow prediction?

Thanks!

Get in touch if you have questions about these ideas or if you are interested in collaborating.

Email: jhamman@ucar.edu

Github: https://github.com/jhamman/met-ml