# Inverse Estimation in Streamflow Modeling with Variational Inference

**Christopher Krapu**[1], Mukesh Kumar[1,2], Mark Borsuk[1]

christopher.krapu@duke.edu mukesh.kumar@duke.edu mark.borsuk@duke.edu

[1]Department of Civil and Environmental Engineering, Duke University, [2]Nicholas School of the Environment, Duke University

## Introduction

**Estimating unknown inputs for a dynamical system conditioned on output data can be difficult due to cross-timestep dependencies and a poorly constrained inverse solution space. Applying prior distributions to the model inputs helps constrain the problem but usually necessitates Markov chain Monte Carlo (MCMC) which is frequently too slow in many environmental applications. This work addresses this shortcoming by employing variational inference in tandem with a hydrology model to estimate rainfall volumes given observations of streamflow.**

## Background

- Inverse modeling can be viewed as a special case of the more general Bayesian inferential problem of calculating the posterior distribution where the unknown inputs are described as model parameters. **In the inverse case, a large number of inputs are typically estimated.** The Metropolis-Hastings algorithm performs poorly in this setting as the probability of generating suitable candidate parameter values decreases rapidly with increasingly large parameter sets.

- More advanced MCMC methods such as Hamiltonian Monte Carlo (HMC) perform better in high-dimensional parameter spaces but require calculating the gradient of the posterior density with regard to the model parameters. For recurrent or dynamical system models, applying HMC requires propagating gradients through each timestep. **Unfortunately, for environmental process models defined on relatively modest periods of time (~1000 timesteps), each HMC sample can require prohibitively large amounts of time**

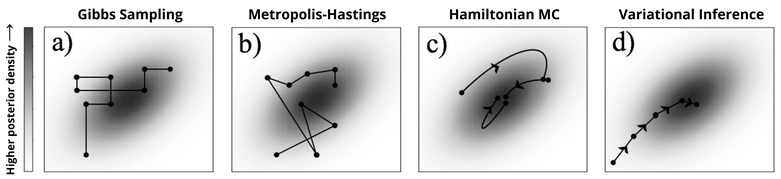


**Figure 1: Conceptual overview of Bayesian estimation methods.** (a) - (c) all indicate Markov chain Monte Carlo methods. Gibbs sampling is able to deal with difficult posterior distributions by making one-parameter-at-a-time proposals which are drawn directly from the joint density conditioned on all other model parameters. However, it can be difficult to identify the proper conditional distributions for complicated models. The Metropolis algorithm makes proposals in random directions, leading to poor performance in high-dimensional problems. Hamiltonian Monte Carlo (c) simulates physical dynamics on a surface defined by the model's posterior density. Variational inference (d) employs gradient ascent to iteratively converge towards a good approximation of the posterior.

- An alternative to MCMC is to identify an easily-optimized parametric distribution and attempt to match it to the true posterior. This approach describes variational inference (VI), a broad class of methods aimed at rapid estimation. While these are not guaranteed to converge to the posterior like sampling-based approaches, **VI can be many orders of magnitudes faster for problems with large numbers of parameters or observations[1].**

- A major shortcoming of VI-based work in the past is that each statistical model required customized estimation algorithms tailored to the structure of the model. Recent work in black-box inference methods such as automatic differentiation variational inference[2] (ADVI) has enabled the generic application of variational inference by non-experts to a much broader class of problems. **We employ ADVI in this work to estimate inputs to a hydrology model relating precipitation and evaporation to streamflow.**

## Methods

### Hydrology model structure

We implemented GR4J[3], a four parameter hydrology model, in Theano and PyMC3[4] to enable calculation of model gradients. **This model simulates the dynamics of a dual reservoir system at a daily timestep** subject to rainfall inputs and losses from streamflow and evaporation. Two parameters control the size of the soil and stream reservoir, respectively. A single parameter controls the speed of streamflow routing through the system and the fourth parameter accounts for a constant loss term intended to represent recharge to the groundwater table.

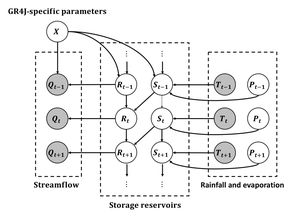


**Figure 2: GR4J model structure**

### Rainfall model priors

We assumed a **log-normal distribution of rainfall volumes** with diffuse log-normal priors on both parameters of the volume distribution. In each application presented here, we also provided a binary variable indicating whether or not it was raining on a given day. In future work, we hope to extend this to include cross-day correlations in the presence of rain as well as temporal structure in rainfall volume.

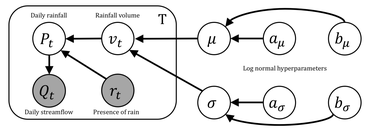


**Figure 3: Stochastic rainfall model**

## Case Studies

### Simulation study

- **800 days of simulated daily streamflow were generated** with GR4J using rainfall, evaporation and streamflow data from the catchment of the Strawberry River near Poughkeepsie, Arkansas (USGS Stream Gauge 07074000).
- T**hree different estimation algorithms were employed** to recover the rainfall inputs: the Metropolis-Hastings algorithm, the No-U-Turn Sampler (NUTS) and automatic differentiation variational inference (ADVI). We employed standard implementations from PyMC3 in all three cases.
- W**e evaluated each algorithm for its ability to estimate rainfall,** employing Bayesian $R^2$ as a measure of predictive accuracy[5]. We also calculated the time-per-iteration for each method. This experiment was repeated across streamflow sequences of varying length from 10 days up to 800 days.

### Multidecadal estimation

- We then attempted to e**stimate rainfall across T = 9000 days** (24.6 years) of observed streamflow and inputs from the Strawberry River catchment.
- Approximately 61% of these days had measurable precipitation and consequently **~5500 daily rainfall volumes needed to be estimated.**
- Prior to rainfall estimation, we calculated point estimates of GR4J parameters using a single year of full precipitation, evaporation and streamflow data. We then treated these as ground truth values in the rainfall estimation model.
- We evaluated the accuracy of the inverse model estimates via Bayesian $R^2$ applied to daily rainfall as well as 3-day, 10-day and 30-day rainfall totals calculated over temporally disjoint periods.

## Results

### Simulation study

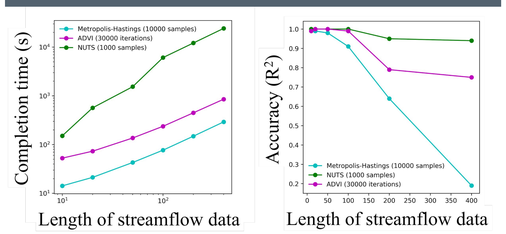


**Figure 4: Completion time and predictive accuracy.** While the Metropolis algorithm requires the least time per iteration, its performance suffers dramatically for sequences of simulated streamflow longer than >100 timesteps. NUTS compares favorably with regard to accuracy but requires unacceptably large amounts of time to draw samples for larger parameter sets. ADVI appears to scale well in terms of both accuracy and completion time with larger datasets.

### Multidecadal estimation

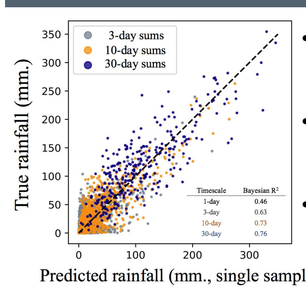


**Figure 5: Predicted vs. observed rainfall.**

- The unexplained variation in precipitation could have many potential causes. Mismatch between model structure and real-world processes is known to be especially great for highly simplified abstractions of catchment dynamics.
- The 3-day, 10-day and 30-day sums show higher $R^2$ scores in part because of the daily discretization inherent in GR4J which precludes same-day increases in streamflow due to rainfall.
- Completing 30,000 iterations of ADVI for this model required approximately 3.5 hours on a standard desktop CPU.

## Conclusions

- **Writing environmental models in probabilistic programming frameworks like PyMC3/Theano enables powerful variational methods to be employed.**
- **ADVI is effective for inverse estimation of nonlinear dynamical systems and scales well in comparison to MCMC.**
- **Inverse estimation for >$10^3$ inputs within a matter of hours is feasible.**

*References*

[1]*Blei, D.M., Kucukelbir, A., McAuliffe, J.D., 2017. Variational Inference: A Review for Statisticians. Journal of the American Statistical Association 112, 859–877.*
[2]*Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., Blei, D.M., 2016. Automatic Differentiation Variational Inference. arXiv:1603.00788.*
[3]*Perrin, C., Michel, C., Andréassian, V., 2003. Improvement of a parsimonious model for streamflow simulation. Journal of Hydrology 279, 275–289.*
[4]*Salvatier, J., Wiecki, T.V., Fonnesbeck, C., 2016. Probabilistic programming in Python using PyMC3. PeerJ Computer Science 2, e55.*
[5]*Gelman, A., Goodrich, B., Gabry, J., Ali, I., 2017. R-squared for Bayesian regression models.*

Duke NICHOLAS SCHOOL OF THE ENVIRONMENT

NASA