National Aeronautics and Space Administration

UQ-aware Machine Learning for Uncertainty Quantification

Principal Investigator: Amy Braverman (398); Co-Investigators: Jouni Susiluoto (398), Margaret Johnson (398), Houman Owhadi (Caltech), Houman Owhadi (Caltech)

Program: FY21 R&TD Topics **Strategic Focus Area: Uncertainty Quantification**

Objectives

The high-level objective of this project is to develop and demonstrate the use of a new machine learning technology called Kernel Flows (KF) [1] which, when combined with Gaussian Process (GP) prediction, emulates complex physical forward models at the core of remote sensing retrieval algorithms. GP's model a correlated field in the model's input space as an infinite dimensional Gaussian distribution with a covariance structure- the kernel- having parameters estimated from data. KF is a method for learning the kernel (and its parameters) from data. We refer to the use of this kernel to carry out GP prediction as a *Kernel Flows emulator* (KF emulator).
The specific objectives of this project are to 1) implement a KF emulator for the forward model used by NASA's upcoming Surface Biology and Geology (SBG) mission, including radiative transfer calculations; and 2) quantify the performance characteristics of this KF emulator, and compare them to those of the most commonly used methods of increasing forward model speed in operational retrievals in imaging spectroscopy applications.

Background

The direct motivation for this research is the need to be run very large numbers of retrievals in the context of uncertainty quantification. Retrievals for mission like SBG and other Earth System Observatory missions (indeed also for existing missions like OCO-2/3, MLS, and AIRS) are computationally intensive largely because the forward models embedded inside them are computationally intensive, and because they must be run multiple times in an iterative search. Quantifying uncertainty on retrieval outputs using techniques like those described in [2] require repeating those retrievals many times to create a distribution of outputs. It is simply not feasible to do this using full-physics models.

The use of emulators to relieve computational burden in this setting is not new. "Look-up tables" have been traditionally used in the remote sensing community as a sort of poor-person's emulator. Neural networks have also achieved impressive performance [3,4] as emulators, but neither of these approaches can quantify the uncertainty they induce because they are not based on probability models. KF emulators, which are based on GPs, can quantify this uncertainty.

Approach and Results (1)

Our approach is to emulate the radiative transfer (RT) portion of the SBG forward model as shown in Figure 1. We fit a GP to a training set of RT inputs and outputs to learn a predictor (and its uncertainty) into which a new input can be fed. RT inputs are atmospheric properties (**x**atm): water vapor and aerosol optical depth. RT outputs are path radiance (**r**), transmittance (**t**), and spherical albedo (**s**).

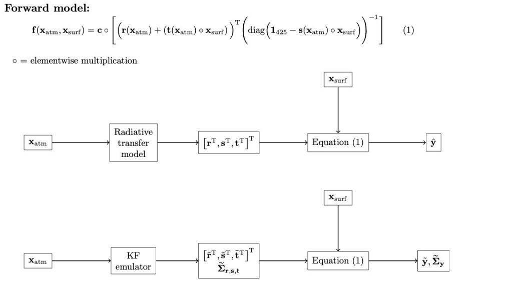

Forward model:
$\mathbf{f}(\mathbf{x}_{\mathrm{atm}}, \mathbf{x}_{\mathrm{surf}}) = \mathbf{c} \circ \left[\left(\mathbf{r}(\mathbf{x}_{\mathrm{atm}}) + (\mathbf{t}(\mathbf{x}_{\mathrm{atm}}) \circ \mathbf{x}_{\mathrm{surf}}) \right)^{\mathrm{T}} \left(\mathrm{diag}\left(\mathbf{1}_{425} - \mathbf{s}(\mathbf{x}_{\mathrm{atm}}) \circ \mathbf{x}_{\mathrm{surf}} \right) \right)^{-1} \right]$ (1)
$\circ$ = elementwise multiplication
$\mathbf{x}_{\mathrm{surf}}$
$\mathbf{x}_{\mathrm{atm}}$
Radiative transfer model
$[\mathbf{r}^{\mathrm{T}}, \mathbf{s}^{\mathrm{T}}, \mathbf{t}^{\mathrm{T}}]^{\mathrm{T}}$
Equation (1)
$\hat{\mathbf{y}}$
$\mathbf{x}_{\mathrm{surf}}$
$\mathbf{x}_{\mathrm{atm}}$
KF emulator
$[\tilde{\mathbf{r}}^{\mathrm{T}}, \tilde{\mathbf{s}}^{\mathrm{T}}, \tilde{\mathbf{t}}^{\mathrm{T}}]^{\mathrm{T}}$ $\widetilde{\boldsymbol{\Sigma}}_{\mathbf{r},\mathbf{s},\mathbf{t}}$
Equation (1)
$\tilde{\mathbf{y}}, \widetilde{\boldsymbol{\Sigma}}_{\mathbf{y}}$

Figure 1. We replace the computationally expensive radiative transfer model with a KF emulator. Tilde indicates estimates, and T indicates vector or matrix transpose.

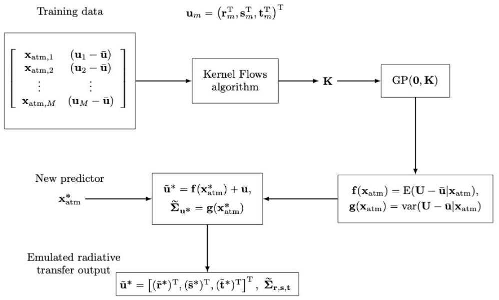
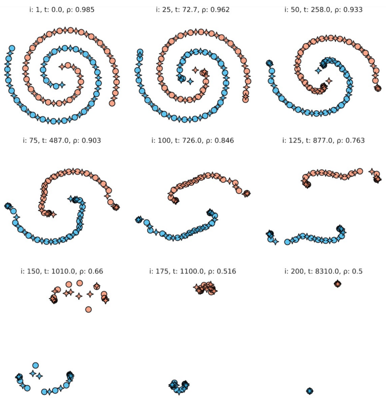

Training data
$\mathbf{u}_m = (\mathbf{r}_m^{\mathrm{T}}, \mathbf{s}_m^{\mathrm{T}}, \mathbf{t}_m^{\mathrm{T}})^{\mathrm{T}}$
$\begin{bmatrix} \mathbf{x}_{\mathrm{atm},1} & (\mathbf{u}_1 - \bar{\mathbf{u}}) \\ \mathbf{x}_{\mathrm{atm},2} & (\mathbf{u}_2 - \bar{\mathbf{u}}) \\ \vdots & \vdots \\ \mathbf{x}_{\mathrm{atm},M} & (\mathbf{u}_M - \bar{\mathbf{u}}) \end{bmatrix}$
Kernel Flows algorithm
$\mathbf{K}$
$\mathrm{GP}(\mathbf{0}, \mathbf{K})$
New predictor
$\mathbf{x}^*_{\mathrm{atm}}$
$\tilde{\mathbf{u}}^* = \mathbf{f}(\mathbf{x}^*_{\mathrm{atm}}) + \bar{\mathbf{u}}$, $\widetilde{\boldsymbol{\Sigma}}_{\mathbf{u}*} = \mathbf{g}(\mathbf{x}^*_{\mathrm{atm}})$
$\mathbf{f}(\mathbf{x}_{\mathrm{atm}}) = \mathrm{E}(\mathbf{U} - \bar{\mathbf{u}} | \mathbf{x}_{\mathrm{atm}})$, $\mathbf{g}(\mathbf{x}_{\mathrm{atm}}) = \mathrm{var}(\mathbf{U} - \bar{\mathbf{u}} | \mathbf{x}_{\mathrm{atm}})$
Emulated radiative transfer output
$\tilde{\mathbf{u}}^* = \left[(\tilde{\mathbf{r}}^*)^{\mathrm{T}}, (\tilde{\mathbf{s}}^*)^{\mathrm{T}}, (\tilde{\mathbf{t}}^*)^{\mathrm{T}} \right]^{\mathrm{T}}, \ \widetilde{\boldsymbol{\Sigma}}_{\mathbf{r},\mathbf{s},\mathbf{t}}$

Figure 2. The Kernel Flows algorithm uses training data to estimate the covariance structure (**K**) of the (centered) input data. Then, when a new input is encountered, a new output is estimated by Gaussian Process prediction (with uncertainty, $\widetilde{\boldsymbol{\Sigma}}_{\mathbf{r},\mathbf{s},\mathbf{t}}$).

Approach and Results (2)

The most difficult part of fitting a GP is estimating its covariance function (kernel), **K**. For stationary, isotropic processes, it's relatively easy to fit a parametric kernel of pre-defined form by estimating its parameters from data. However, stationarity and isotropy are restrictive assumptions that may not be realistic. For nonstationary, anisotropic fields, KF deforms the input space by iteratively minimizing a loss function that compares the quality of output predictions derived from samples, to predictions made from subsamples taken from those samples. At each iteration, the input points are moved (slightly) in the direction that makes the loss smaller. In the deformed space, the stationary, isotropic covariance assumption does hold.



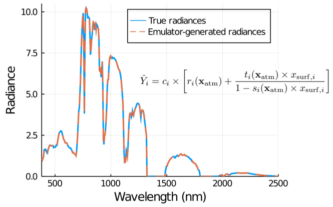


Figure 4. Radiance vectors from the SBG forward model using true (RT, in blue) and emulated (red) radiative transfer. The blue spectrum is y! and the dashed red spectrum is *y#* in Figure 1. Scene is one pixel in the Beckman Lawn test case [7].

Figure 3. Nonparametric Kernel Flows applied to the "Swissroll Cheesecake" example [1]. Input data lie in a two-dimensional space, and output (in this example) are binary (+1/-1). In 200 iterations, the input data are successively deformed in a way that increases the ability to predict the output as quantified by the Kernel Flows loss function (here, "leave-one-in"). Panels show current data positions at 25-iteration increments. Deformations are determined from randomly selected subsamples (training data, circles) at each iteration, but applied to all data including "test" points (stars).

Significance

JPL and NASA missions are in need of rigorous uncertainty quantification (UQ) for their data products in order to enable rigorous hypothesis testing and confirmatory analysis. At present, computational bottlenecks due to expensive forward models embedded in retrieval algorithms prevent large-scale Monte-Carlo-based implementation of UQ methods that would produce these uncertainties. Emulators have the potential to break this bottleneck by providing fast, accurate approximations to forward models, along with estimates of the emulator's own prediction uncertainties. KF provides a flexible, computationally efficient way to learn information required without difficulties encountered in maximum likelihood or Bayesian estimation, and without restrictive assumptions.

SBG provides a particularly timely and useful testbed for developing KF. Recent work [5, 6] show the importance of quantifying uncertainties of retrieval estimates for achieving the science objectives of the mission. Further, as data volumes for new and existing missions grow, it becomes harder to keep up with processing. This is already a problem for OCO-2: the data set is now so large that reprocessing all of it is not feasible. Future missions like SBG may have no choice but to use forward model emulators in their retrieval algorithms. KF emulators not only enable proper UQ, but will position JPL to meet the challenges of future mission data processing.

National Aeronautics and Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

www.nasa.gov

References:

[1] H. Owhadi, and G.R. Yoo, DOI: 10.1016/j.jcp.2019.03.040.
[2] A. Braverman et al., DOI: 10.1137/19M1304283.
[3] B.D. Bue et al., DOI: 10.5194/amt-12-2567-2019.
[4] P. G. Brodrick et al., DOI: 10.1016/j.rse.2021.112476.
[5] N. Carmon et al., DOI: 10.1016/j.rse.2018.07.003.
[6] D. R Thompson et al., DOI: 10.1016/j.rse.2020.112038.
[7] D.R. Thompson et al., DOI: 10.1016/j.rse.2020.111898.

PI/Task Mgr Contact
Email: Amy.J.Braverman@jpl.nasa.gov

Clearance Number: CL#21-4745
RPC/JPL Task Number: R21130