The Journal of Open Source Software

DataAssimilationBenchmarks.jl: a data assimilation research framework.

Colin Grudzien[1] **and Sukhreen Sandhu**[2]

1 Department of Mathematics and Statistics, University of Nevada, Reno **2** Department of Computer Science and Engineering, University of Nevada, Reno

DOI: 10.21105/joss.04129

Software

- Review
- Repository
- Archive

Editor: Bita Hashemineżhad

Reviewers:

- @peanutfun
- @tmigot

Submitted: 03 December 2021
Published: 03 February 2022

Summary

Data assimilation (DA) refers to techniques used to combine the data from physics-based, numerical models and real-world observations to produce an estimate for the state of a time-evolving random process and the parameters that govern its evolution (Asch et al., 2016). Owing to their history in numerical weather prediction, full-scale DA systems are designed to operate in an extremely large dimension of model variables and observations, often with sequential-in-time observational data (Carrassi et al., 2018). As a long-studied "big-data" problem, DA has benefited from the fusion of a variety of techniques, including methods from Bayesian inference, dynamical systems, numerical analysis, optimization, control theory and machine learning. DA techniques are widely used in many areas of geosciences, neurosciences, biology, autonomous vehicle guidance and various engineering applications requiring dynamic state estimation and control.

The purpose of this package is to provide a research framework for the theoretical development and empirical validation of novel data assimilation techniques. While analytical proofs can be derived for classical methods such as the Kalman filter in linear-Gaussian dynamics (Jazwinski, 2007), most currently developed DA techniques are designed for estimation in nonlinear, non-Gaussian models where no analytical solution may exist. Similar to nonlinear optimization, DA methods, therefore, must be studied with rigorous numerical simulation in standard test-cases to demonstrate the effectiveness and computational performance of novel algorithms. Pursuant to proposing a novel DA method, one should likewise compare the performance of a proposed scheme with other standard methods within the same class of estimators.

This package implements several standard data assimilation algorithms, including widely used performance modifications that are used in practice to tune these estimators. This software framework was written specifically to support the development and intercomparison of the novel single-iteration ensemble Kalman smoother (SIEnKS) (Grudzien C. & Bocquet, 2021). Details of the primary DA schemes, including pseudo-code for the methods detailing their implementation, and DA experiment benchmark configurations, with root mean square error and ensemble spread diagonostics for estimator validation, can be found in the above principal reference. Additional details on numerical integration schemes used in this work for simulating the Lorenz-96 model are found in the secondary reference (C. Grudzien et al., 2020).

Statement of need

Standard libraries exist for full-scale DA system research and development, e.g., the Data Assimilation Research Testbed (DART)(Anderson et al., 2009), but there are fewer standard options for theoretical research and algorithm development in simple test systems. DataAssimilationBenchmarks.jl provides one framework for studying ensemble-based filters and sequential smoothers that are commonly used in online, geoscientific prediction settings. Validated methods, and methods in development, focus on evaluating the performance and the structural stability of techniques over wide ranges of hyper-parameters that are commonly used to tune estimators in practice. Specifically, this is designed to run naively parallel experiment configurations over independent parameters such as ensemble size, static covariance inflation, observation operator / network designs that affect the estimator stability and performance. Templates for running naively parallel experiments using Juila's core parallelism, or using Slurm to load experiments in parallel with a queueing system are provided.

Comparison with similar projects

Similar projects to DataAssimilationBenchmarks.jl include the DAPPER Python library (Raanes & others, 2018), DataAssim.jl used by (Vetra-Carvalho et al., 2018), and EnsembleKalmanProcesses.jl (Constantinou & others, 2021) of the Climate Modeling Alliance. These alternatives are differentiated primarily in that:

- DAPPER is a Python-based library which is well-established, and includes many of the same estimators and models. However, numerical simulations in Python run notably slower than simulations in Julia when numerical routines cannot be vectorized in Numpy (*Julia Benchmarks*, 2021). Particularly, this can make the wide hyper-parameter search intended above computationally challenging without utilizing additional packages such as Numba (*Numba Documentation*, 2021) for code acceleration such as faster for-loops.
- DataAssim.jl is another established Julia library, but notably lacks an implementation of ensemble-variational techniques which were the focus of the initial development of DataAssimilationBenchmarks.jl. For this reason, this package was not selected for the development and intercomparison of the SIEnKS, though this package does have implementations of a variety of standard stochastic filtering schemes.
- EnsembleKalmanProcesses.jl is another established Julia library, but notably lacks traditional DA approaches such as the classic, perturbed observation EnKF/S and the classic ETKF/S. For this reason, this package was not selected for the development and intercomparison of the SIEnKS.

Validated methods currently in use

Estimator / implemented techniques	Tuned inflation	Adaptive inflation	Linesearch	Multiple data assimilation
EnKF	X	X	NA	NA
ETKF	X	X	NA	NA
MLEF	X	X	X	NA
EnKS	X	X	NA	NA
ETKS	X	X	NA	NA
MLES	X	X	X	NA
SIEnKS	X	X	X	X
Gauss-Newton IEnKS	X	X		X

The future development of the DataAssimilationBenchmarks.jl package is intended to expand upon the existing, ensemble-variational filters and sequential smoothers for robust intercomparison of novel schemes and the further development of the SIEnKS scheme. Novel mechanistic models for the DA system are also in development. Currently, this supports state and joint state-parameter estimation in the L96-s model (C. Grudzien et al., 2020) in both ordinary and stochastic differential equation formulations. Likewise, this supports a variety of observation operator configurations in the L96-s model, as outlined in (Grudzien C. & Bocquet, 2021).

Installation

The main module DataAssimilationBenchmarks.jl is a wrapper module including the core numerical solvers for ordinary and stochastic differential equations, solvers for DA routines and the core process model code for running twin experiments with benchmark models. These methods can be run stand-alone in other programs by calling these functions from the DeSolvers, EnsembleKalmanSchemes and L96 sub-modules from this library. Future solvers and models will be added as sub-modules in the methods and models directories respectively.

In order to get the full functionality of this package one needs to install the dev version. This provides the access to edit all of the outer-loop routines for setting up twin experiments. These routines are defined in the modules in the "experiments" directory. The "slurm_submit_scripts" directory includes routines for parallel submission of experiments in Slurm. Data processing scripts and visualization scripts (written in Python with Matplotlib and Seaborn) are included in the "analysis" directory.

Installing a dev package from the Julia General registries

In order to install the dev version to a Julia environment, one can use the following commands in the REPL

```
pkg> dev DataAssimilationBenchmarks
```

The installed version will be included in

```
~/.julia/dev/
```

on Linux and the analogous directory with respect Windows and Mac systems.

Alternatively, you can install this from the repository Github directly as follows:

```
pkg> dev https://github.com/cgrudz/DataAssimilationBenchmarks.jl
```

Documentation

Documentation on the usage of the methods in the current version of the package is included in the README.md for the Github package above.

Acknowledgements

Colin Grudzien wrote all core numerical code in the DataAssimilationBenchmarks.jl package. Sukhreen Sandhu supported development of the package by building and validating code test cases, and supporting the development of the package structure and organization. This work was supported by the University of Nevada, Reno, Office of Undergraduate Research's Pack Research Experience Program (PREP) which supported Sukhreen Sandhu as a research assistant. This work benefited from the DAPPER library which was referenced at times for the development of DA schemes.

References

Anderson, J., Hoar, T., Raeder, K., Liu, H., Collins, N., Torn, R., & Avellano, A. (2009). The data assimilation research testbed: A community facility. *Bulletin of the American Meteorological Society*, *90*(9), 1283–1296.

Asch, M., Bocquet, M., & Nodet, M. (2016). *Data assimilation: Methods, algorithms, and applications*. SIAM.

Carrassi, A., Bocquet, M., Bertino, L., & Evensen, G. (2018). Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *Wiley Interdisciplinary Reviews: Climate Change*, *9*(5), e535.

Constantinou, N. C., & others. (2021). *EnsembleKalmanProcesses.jl*. https://github.com/CliMA/EnsembleKalmanProcesses.jl

Grudzien, C., & Bocquet, M. (2021). A fast, single-iteration ensemble kalman smoother for sequential data assimilation. *Geoscientific Model Development Discussions*, 1–62.

Grudzien, C., Bocquet, M., & Carrassi, A. (2020). On the numerical integration of the lorenz-96 model, with scalar additive noise, for benchmark twin experiments. *Geoscientific Model Development*, *13*(4), 1903–1924.

Grudzien, C., Bocquet, M., & Carrassi, A. (2020). On the numerical integration of the lorenz-96 model, with scalar additive noise, for benchmark twin experiments. *Geoscientific Model Development*, *13*(4), 1903–1924.

Jazwinski, A. H. (2007). *Stochastic processes and filtering theory*. Courier Corporation.

Julia benchmarks. (2021). Accessed: 2021-11-29. https://julialang.org/benchmarks/

Numba documentation. (2021). Accessed: 2021-11-29. https://numba.readthedocs.io/en/stable/

Raanes, P. N., & others. (2018). *Nansencenter/DAPPER: Version 0.8* (Version v0.8) [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.2029296

Vetra-Carvalho, S., Van Leeuwen, P. J., Nerger, L., Barth, A., Altaf, M. U., Brasseur, P., Kirchgessner, P., & Beckers, J. M. (2018). State-of-the-art stochastic data assimilation methods for high-dimensional non-gaussian problems. *Tellus A: Dynamic Meteorology and Oceanography*, *70*(1), 1–43.