Groundwater

Technology Spotlight/ Technology Editor / Chris Lowry

Jupyter Notebooks for Parameter Estimation, Uncertainty Analysis, and Optimization with PEST++

Chanse Ford[1,2], Wonsook Ha[3], Katherine Markovich[4], and Johanna Zwinger[5]

Introduction

The complex interactions of climate, hydrology, and geology often require numerical simulations to understand how those interactions shape the flow and quantity of groundwater. Regardless of the spatial or temporal scales, numerical groundwater modeling simulations quickly evolve into a complex convolution of input data, model parameters, and the underlying governing equations of the hydrologic environment during the development process. Efficient optimization of parameter values requires the application of regularized inverse modeling techniques. One inverse modeling code widely adapted in the hydrologic modeling community is the Parameters ESTimation (PEST) software, first released in 1994 (Doherty 2018), and its subsequent iteration PEST++, first released in 2012 (White et al. 2020). While PEST is still being developed and used, PEST++ differs in that it enables users to access and automize its functionalities with the Python coding language. The package pyEMU, for example, allows users to setup and run PEST++ with Python.

PEST/PEST++ went through several development cycles in the preceding thirty years. In 2018, PEST++ was completely overhauled to be a suite of decision support model partner software designed to be modular and extensible, so that continuous development enables PEST++ to better solve evolving decision-making problems. Perhaps the most substantial addition was the iterative ensemble smoother (PESTPP-IES; White et al. 2020) which efficiently handles highly parameterized inversion. This version also resulted in improved partner software tools such as global sensitivity algorithm (SEN), management optimization algorithm (OPT) and several completely new decision support tools, such as data assimilation (DA) and multi-objective optimization under uncertainty (MOU). All PEST++ tools have a built-in fault-tolerant, multithreaded parallel run manager and are model independent, using the same protocol as the widely used PEST software suite (White et al. 2020).

To aid new users in learning the functionality of the PEST++ suite of software, a collection of interactive tutorials was created by the developers from the Groundwater Modeling Decision Support Initiative (GMDSI) and the U.S. Geological Survey (USGS). These tutorials are available to the public via GitHub (https://github.com/gmdsi/GMDSI_notebooks) and come in the form of Python scripts embedded in the Jupyter Notebooks interface. The objectives of this technical review are to briefly describe the current and previous incarnations of PEST and PEST++, provide instructions for accessing and using the GMDSI tutorials, and provide insight into the effectiveness of the tutorials at introducing new users to PEST/PEST++ concepts and implementation.

The reviewers come from a variety of backgrounds and experience levels with groundwater modeling and parameterization, but all are early career groundwater modelers learning to use PEST++. Given the early career status of the reviewers, the assessment of these tutorials is provided within the context of the gap between the modeling training received at university programs and applied modeling techniques outside academia. Anecdotally, graduate level education in hydrologic modeling provides adequate instruction in the use of the Modular Hydrologic Model (MODFLOW; Harbaugh 2005) for groundwater modeling in terms of the fundamental science of numerical groundwater simulation and construction of these

[1]Corresponding author: United States Geological Survey, Upper Midwest Water Science Center, Lansing, MI; cford@usgs.gov

[2]United States Geological Survey, Upper Midwest Water Science Center, Lansing, MI

[3]United States Geological Survey, Upper Midwest Water Science Center, Madison, WI

[4]INTERA Incorporated, Albuquerque, NM

[5]INTERA Incorporated, Perth, WA, Australia

Received September 2024, accepted October 2024.

doi: 10.1111/gwat.13447

simulations for research. However, the groundwater modeling instruction is often lacking with respect to parameter estimation and uncertainty analysis techniques. This is likely caused by a desire to focus educational efforts on the foundations of groundwater modeling. After reviewing the tutorials, we discuss how well these tutorials address this perceived lack of relevant training in academia.

Running the Tutorial Notebooks

The GMDSI GitHub repository (2023) for the notebooks contains easy to follow installation instructions including how to download the notebooks, how to install Python, and how to create a Python environment. The notebooks can be downloaded in two different ways, either by downloading the folder manually or by cloning the repository. Using git to clone the repository requires a basic knowledge of git but has the advantage of automatically syncing a local git repository with updates from the remote main repository. In addition, since these notebook tutorials are intended to be "living" notebooks, the community is able to contribute to them via the "fork and pull" model of git-hosted software development. After installing Python, dependencies, and Conda, one can open an Anaconda prompt and create an environment using the .yml file in the repository. This file contains the versioned packages, which should enable the user to run the Jupyter Notebooks without having any versioning issues. Overall, even though familiarity with git, Conda, and Jupyter help during the installation, the instructions on the GitHub site are clear and concise, making it easy for users without the aforementioned experience to follow and get the notebooks running.

Tutorial Notebooks Overview

At the time the authors cloned the repository, 36 tutorial notebooks were developed. They fall into three general categories: (1) Uncertainty Analysis and Uncertainty Quantification (UA/UQ) Concepts; (2) PEST++ Concepts and Mathematics; and (3) Decision Support Modeling with PEST++. The following sections provide a brief overview of the categories, their respective notebooks, and learning objectives.

Category 1: UA/UQ Concepts

This category of tutorial notebooks provides an introduction to the statistical and mathematical concepts of inverse methods and sets the stage for the rest of the notebooks by introducing the demonstration model and pyEMU (White et al. 2016), a helpful Python wrapper for building PEST++ workflows. While the learning objectives of these notebooks are aimed at a level for practitioners who are brand new to parameter estimation, they also serve as references for revisiting the basics (Table 1).

Category 2: PEST++ Concepts and Mechanics

This category of notebooks lays out the fundamental concepts of PEST++, which can help practitioners understand the Category 3 notebooks in this tutorial series. In general, these notebooks can be run in any order. Their purpose is to introduce the concepts of non-uniqueness, identifiability, first order second moment (FOSM) and uncertainty analysis using prior Monte Carlo, as well as explaining how regularization can help achieve non-uniqueness. They further give an introduction on how to set up the PEST control file with pyEMU and advantages of a highly parameterized approach for history matching (Table 2).

Category 3: Decision Support Modeling with PEST++

The final group of notebooks delves into the various tools available in the PEST++ suite in more detail. The first two notebooks must be run before the subsequent notebooks in this section that the user is interested in (Table 3).

Table 1
A List of Names and Learning Objective Summaries for Notebooks in Category 1

Name	Learning Objectives
Intro to Regression	Fit simple model to noisy data, introduce sum of squared weighted errors (SSE), introduce Jacobian, explore degrees of freedom and over/underfitting
Intro to Bayes	Introduce conditional probabilities, introduce Bayes theorem, explore impact of prior and likelihood mean and standard deviation on posterior distribution
Intro to Geostatistics	Introduce variogram models (spherical, exponential, Gaussian), use ordinary kriging to interpolate sparse data based on model fit, introduce spectral simulation
Intro to SVD	Apply Singular Value Decomposition (SVD) to PEST++ to solve the inverse problem (model calibration) using a Jacobian matrix
Intro to Freyberg Model	Domain and time discretization, boundary conditions, properties, observations, and forecasts for the demonstration model
Intro to pyEMU	"*Pst*" class, PEST control data, PEST++ options, parameter, observation, and phi utilities, automatic weight adjustment, geostatistics, prior information equations, matrix manipulation, linear analysis, ensemble objects

Table 2
A List of Names and Learning Objective Summaries for Notebooks in Category 2

Name	Learning Objectives
Trial and error	Learn the concept of trial and error by adjusting hydraulic conductivity (K) and recharge (R) manually
Pest setup	Explore the basics of PEST++ using the Freyberg groundwater model and learn how to construct the PEST template, instruction, and control files for the Gauss-Levenberg-Marquardt (GLM) run
Calibrate k	Calibrate the Freyberg groundwater model and run an uncertainty analysis using PESTPP-GLM with hydraulic conductivity (K) update, check the goodness of fit at the end of model run, and get familiarized with pyEMU
Calibrate k and r	Re-calibrate with PESTPP-GLM and perform the uncertainty analysis on the Freyberg model by adding recharge (R) in addition to K, and investigate the non-uniqueness and parameter correlation of K and R
Calibrate k r fluxobs	Examine the effect of adding head and flux observations to history-matching, parameter uncertainty, and forecast uncertainty from the Freyberg model with K and R adjustment
GLM response surface	Visualize the objective function surface, investigate how adding flux observations change the objective function surface, and understand how the Marquardt lambda is used in calculation of parameter upgrades during the objective function gradient-based inversion process
Pilotpoints setup	Adds pilot point parameters for K for the Freyberg model using pyEMU
Pilotpoints Run	Runs a PESTPP-GLM using pilot points created in the previous notebook, then visualizes the outputs
Regularization	The addition of the pilot points in the previous tutorials resulted in overfitting from too many parameters relative to the number of observations; this tutorial addresses that by using Tikhonov regularization and the PHIMLIM tool of PEST/pyEMU
Intro to FOSM	Delves into the equations governing uncertainty quantification in PEST++ using FOSM and contextualizes them within the Freyberg model constructed by the pilot points tutorials
Local and global sensitivity	Two notebooks visualizing and examining the sensitivity analyses performed on the Freyberg pilot points model on a single set of parameter values (local) and a range of parameter values (global)
Monte Carlo	Performs a Monte Carlo run on the Freyberg model, then examines the uncertainties and compares them to the prior forecast distributions

Table 3
A List of Names and Learning Objective Summaries for Notebooks in Category 3

Name	Learning Objectives
PstFrom Pest Setup	Sets up PEST++ input files for the Freyberg model utilizing the PstFrom class contained in the pyEMU software
Obs and Weights	Examines the observations and weights in the constructed Freyberg PEST++ model and discusses some approaches for deciding the observational weights to apply
Prior Monte Carlo	Performs Prior Monte Carlo runs for the Freyberg model using the SWP and IES tools, explores the outputs, and incorporates forecasts
GLM	Two notebooks that describe the PESTPP-GLM tool. The first discusses the relevant changes to the PEST setup, then the second runs PESTPP-GLM for the Freyberg model
FOSM and Dataworth	Explores linear uncertainty analysis, or FOSM, by applying the techniques to the outputs generated in the preceding PESTPP-GLM runs
IES	Another tutorial with three notebooks. The first applies the PESTPP-IES tool to the Freyberg model and discusses the outputs, while the second discusses and applies localization techniques to the outputs from the first notebook. The third one explains how to restart IES using existing results
DA	This tutorial contains a notebook for prepping the inputs for a PEST++ data assimilation (DA) run, then another notebook which runs PESTPP-DA on the Freyberg model and discusses the results
OPT	Discusses optimization and uncertainty within the context of the Freyberg model using the PESTPP-OPT tool
MOU	The final two notebooks get into multi-objective optimization under uncertainty (MOU) utilizing PESTPP-MOU. These two notebooks can take very long to run on a local machine such as a typical laptop (running both notebooks utilizing all 8 cores of an 11th Gen Intel Core i9-11950H processor took 9.5 h)

Discussion

As a whole, the tutorial notebooks are easy enough to follow and run, assuming users have sufficient familiarity with Python and Conda environments. Hardware requirements are relatively minimal, with the only real issue being extended run times while working with only a few processing cores. The content provides some deep dives into the fundamental processes governing PEST++, introducing concepts to users to aid the comprehension of the processes. Despite the ease of use and depth of knowledge presented in the notebooks, there are still some areas that could improve their reach and adoption by wider audiences.

The notebooks follow a step-by-step procedure to learn background and the basic concepts of PEST++ in Categories 1 and 2 and to implement the gained uncertainty analysis skills to their own projects using advanced notebook examples in Category 3. While these notebooks strive to demonstrate basic-to-advanced PEST++ use cases all in one place, a few notebooks are very lengthy. This is typically noted by the notebook text, but this condition could hinder beginners from fully engaging with the notebooks and lead to mindless clicking of the notebook cells without fully knowing about the content. These longer notebooks could be enhanced by requiring additional interaction by users, such as manipulation of the settings being run, ideally leading to more engagement.

Despite the improvements suggested for the tutorials, they have the potential to fill some important gaps in groundwater modeling education. Formal training in groundwater modeling in the university setting is heavily dependent on the department and individual advisors. While some students receive a thorough enough education in numerical modeling to smoothly transition to applied modeling in their careers, oftentimes, temporal constraints on degree completion can lead to the formal modeling training being cut short. The most common areas to fall victim to these limitations are model calibration and uncertainty analysis, as these are typically performed near the end of the modeling project phase. Few universities offer more than a handful of courses on groundwater modeling at the undergraduate or even at the graduate level, leading to much of the initial years of graduate school spent learning the fundamentals of groundwater models. Based on the experience of the authors, classes are often based on learning groundwater modeling with a GUI as it does not require knowledge of a programming language. It is even less common to include the problem decomposition and identification of relevant prediction of interest in groundwater modeling classes.

Providing the notebooks in the Jupyter Notebook format allows anyone with basic familiarity with Python scripting environments to replicate the processes at their own pace, teaching both the foundations of model parameterization and uncertainty analysis and how to apply the PEST++ software suite in a groundwater modeling environment. Deciding which PEST++ module is appropriate for an individual model will require some critical thinking and judgment calls by users, but the overview of each module is sufficient to guide users to the best path forward. This could be aided by further categorizing the notebooks into smaller sections, making the use of these materials for teaching purposes a little easier to structure. Ideally these notebooks can be incorporated early in the model training process, supplementing the education received in academia while engraining the "uncertainty-first" approach into the student's modeling workflow. Ultimately this should lead to more prepared modelers when they start applying models beyond academia.

While the GMDSI notebooks are a powerful tool for supplementing groundwater modeling education, improvements to these tutorials could further their reach. Prominently, the notebooks are presented in only the scripting format with short introductory texts. They may appeal to a broader audience by incorporating audio or video explanations, aiding students who learn more easily in a lecture type format. Several areas in the notebooks ask the user questions about the topic, but fail to provide an answer for those questions, leaving the user wondering if their answer is the "correct" answer. More of these types of questions, with some sort of answer key available would greatly aid in user comprehension of the topics, especially in the earlier notebooks covering the foundational knowledge. Finally, the Freyberg model applied throughout the notebooks is an idealized construction. Application of the methods presented in the notebooks to a real, applied model would lead to a deeper understanding of the value of PEST++ to applied modeling. This could be achieved by providing an additional section of notebooks for this model or linking the user to a repository containing an applied model that utilizes the PEST++ software.

Conclusions

The tutorial notebooks for the PEST++ parameter estimation and uncertainty analysis software created for the GMDSI provide a comprehensive introduction to the underlying concepts and mathematics driving PEST++, the various modules included in the software suite, and demonstrations of how to apply the PEST++ techniques to a groundwater modeling framework. The notebooks allow anyone with a base level familiarity with Python scripting and Jupyter Notebooks within a Conda environment to familiarize themselves with setting up and executing PEST++ workflows. While the tutorials are constructed well, there are still areas for improvement that could promote engagement as well as broaden the reach of these educational materials. Nevertheless, these are valuable teaching tools for parameter estimation and uncertainty analysis techniques in groundwater modeling, addressing some of the commonly missing aspects of groundwater modeling education.

Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

References

Doherty, J.E. 2018. PEST, *Model-Independent Parameter Estimation—User Manual*, 7th ed. Brisbane: Watermark Numerical Computing.

GMDSI GitHub site. 2023. Groundwater Modeling Decision Support Initiative. https://github.com/gmdsi/GMDSI_notebooks (accessed August 3, 2023).

Harbaugh, A.W. 2005. MODFLOW-2005, The U.S. Geological Survey modular ground-water model – The Ground-Water Flow Process: U.S. Geological Survey Techniques and Methods Report 6-A16. https://pubs.usgs.gov/publication/tm6A16 (accessed February 1, 2023).

White, J.T., M.N. Fienen, and J.E. Doherty. 2016. A python framework for environmental model uncertainty analysis. *Environmental Modelling & Software* 85: 217–228. https://doi.org/10.1016/j.envsoft.2016.08.017

White, J.T., R.J. Hunt, M.N. Fienen, and J.E. Doherty. 2020. Approaches to highly parameterized inversion: PEST++ version 5, a software suite for parameter estimation, uncertainty analysis, and management optimization and sensitivity analysis. U.S. Geological Survey Techniques and Methods Report 7C26. https://pubs.usgs.gov/publication/tm7C26 (accessed February 1, 2023).