

Opportunities for open-source software to accelerate research in applied geophysics



Lindsey J. Heagy¹, Seogi Kang², Joseph Capriotti³, Dominique Fournier⁴, Rowan Cockett⁵, and Douglas W. Oldenburg¹

<https://doi.org/10.1190/tle43020084.1>

Abstract

The potential for open-source software and open-science practices to accelerate research in applied geophysics and thereby contribute to solutions of geoscientific problems impacting society is considered. We provide context on the definition of open source and give a brief history of open-source software in applied geophysics. Drawing from our experience on the SimPEG project, which develops software for simulation and inversion of geophysical data, we provide two examples where research was accelerated because of open-source software. These include the reuse of regularization methods for different geophysical problems (magnetics and time-domain electromagnetics) and the combination of multiple geophysical data types in joint inversions. We also provide an example where research code was repurposed for education and humanitarian projects. Each of these examples was made possible because of the availability of code and the practices adopted by the community of collaborators involved in the project. We conclude with our perspective on how practices adopted by open-source communities that enable collaboration among researchers with different backgrounds, skills, and interests can be applied more broadly in research. This will ultimately increase the use and effectiveness of geophysics in helping solve applied problems.

Introduction and motivation

Geophysics has an important role to play in addressing numerous societal challenges by noninvasively imaging the subsurface. These include locating critical minerals for transition to an electric economy, finding and managing groundwater resources, sequestering CO₂ to prevent more warming of the planet, and further developing other more sustainable sources of energy (geothermal, wind, solar, etc.). Capello et al. (2021) and Capello (2022) provide a vision of how geophysics can have an impact on the United Nations' Sustainable Development Goals. The problems to be addressed are complex and multidisciplinary. Their solutions require observational data, computational methods for working with those data, and interaction between researchers with a range of backgrounds and skills. Our hypothesis is that research in applied geophysics to help reach solutions to these problems can be accelerated by adopting open-source practices in the way we conduct research.

Applied problems are intrinsically multidisciplinary, and their solutions require input from many different disciplines. Geophysics

can provide important information, but its effectiveness can be realized only when we connect with other disciplines. The first connection point requires that we be able to pose the question at hand in terms of relevant physical properties. This could entail collaborating with a geologist to build a conceptual ore-deposit model for mineral exploration or connecting with a geochemist to characterize how physical properties change as CO₂ is injected and reacts with the surrounding formation or fluids. Once data have been collected, we then aim to interpret those data in light of the original question. In this stage, we are faced with the challenges inherent with any inverse problem, particularly that solutions are nonunique. Additional data and a priori information can help constrain a solution or interpretation. This again requires communication between practitioners in multiple fields but also requires tools at the computational level to bring in data and information in a quantitative manner. This is especially relevant with advances in machine learning (ML) that enable a wide array of data types to be analyzed and combined. Improvements in efficiency and effectiveness in the communication of information and data between disciplines, alongside advancements within each discipline, can help contribute to accelerating discovery.

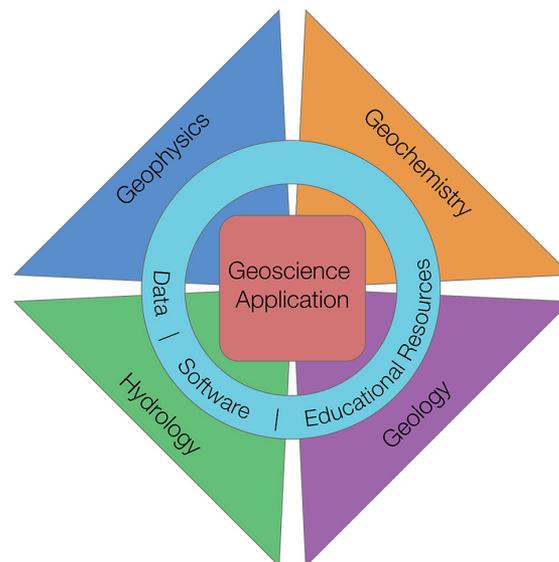


Figure 1. Any application requires perspectives and input from multiple disciplines. Data, software, and educational resources can help provide a bridge between disciplines. After Oldenburg et al. (2020).

Manuscript received 15 October 2023; accepted 18 December 2023.

¹University of British Columbia, Department of Earth, Ocean and Atmospheric Sciences, Vancouver, British Columbia, Canada. E-mail: lheagy@eos.ubc.ca; doug@eos.ubc.ca.

²Stanford University, Department of Geophysics, Stanford, California, USA. E-mail: sgkang09@stanford.edu.

³Colorado School of Mines, Department of Geophysics, Golden, Colorado, USA. E-mail: jcapriot@mines.edu.

⁴Mira Geoscience, Vancouver, British Columbia, Canada. E-mail: dominiquef@mirageoscience.com.

⁵Curvenote Inc., Canmore, Alberta, Canada. E-mail: rowan@curvenote.com.

Progress in science is achieved through the continual development of ideas and the collaborative effort of building upon one another's work. Open-source software enables us to build upon and extend prior work in a very tangible way. In addition to the lines of code, there are practices that have been adopted by many projects to help facilitate collaboration among researchers. Practices such as testing code and peer review of work enable multiple people to work on a common project. Having open discussions about challenges to overcome or new ideas to implement supports the building of connections between researchers from a variety of backgrounds. With computation and scientific software being such an essential part of research in applied geophysics, we see opportunities for open-source software, and more broadly, open-science practices, to accelerate research within geophysics and broaden the impact of our work.

In this article, we illustrate that open-source software and practices can accelerate the exchange of ideas and facilitate collaborations between researchers. We will provide some background and context about open-source software in geophysics and illustrate examples of impacts in research based on our experience in developing and contributing to SimPEG, an open-source project for simulation and inversion of geophysical data.

Background: Open-source software

Before making an argument for open-source software, we begin by defining what we mean by "open source." The Open Source Initiative (OSI) is a nonprofit organization that maintains the Open Source Definition (Open Source Initiative, 2007), which specifies 10 criteria for a software project to be considered open source. Source code must be accessible, and, importantly, the license that is used must allow for the use, adaptation, and redistribution of the original or derived works. The criteria also specify that the license should not discriminate against use by any group or for any field of application.

Broadly speaking, there are two categories of open-source licenses: copyleft licenses and permissive licenses. Copyleft licenses, such as the GNU General Public License (GPL), require any derived works to also be distributed with the same license. Permissive licenses allow derived works to be distributed as open or closed software. Common examples of permissive licenses include the MIT License, BSD licenses, and Apache licenses.

Within the scientific Python community, permissive licenses have been widely adopted. This choice has helped facilitate connections and collaboration among open-source developers and commercial entities by enabling the redistribution of code for commercial purposes. Many resources such as <https://choosealicense.com> and <https://opensource.org> provide further background on software licenses for the interested reader.

Brief overview of open source in applied geophysics

Distributing software with an open-source license is not a new concept in geophysics. Seismic Un*x (Stockwell, 1999) was started in the 1970s. In addition to being a useful code base, the project was important in demonstrating how open-source codes can facilitate computational reproducibility (Claerbout and Karrenbach, 1992), that is, the ability to consistently obtain the same results using the same input data and computational methods as were used by the original authors. Other early open-source projects in and related to geophysics include MODFLOW for simulating groundwater flow (Langevin et al., 2017), Occam 1D for inverting controlled source electromagnetics and magnetotelluric data in 1D (Constable et al., 1987), and generic mapping tools for plotting and visualizing geospatial data. Community mailing lists and websites such as MTNet provided avenues for distributing these codes, which continue to be maintained and widely used.

Several key developments have catalyzed the proliferation of open-source software packages that we see in geophysics today. In the early 1990s, the first version of the Linux kernel was released, providing the basis for many operating systems used in scientific computing (and elsewhere). The Python language was also started in the '90s, followed by Julia, R, and others. These are interpreted languages, as compared to compiled languages such as Fortran and C, and are arguably much easier to use in writing scientific code. The ability to rapidly write useable code is further aided by the existence of packages such as NumPy, SciPy, and Matplotlib, which provide useful functions for scientific computing. These components provide a base layer for the open-source scientific "stack," on top of which other projects can be built.

The next key developments were version control and platforms that provide infrastructure for collaborating on code. The combination of distributed version control with Git, and cloud-hosted platforms like GitHub for managing Git repositories, enables

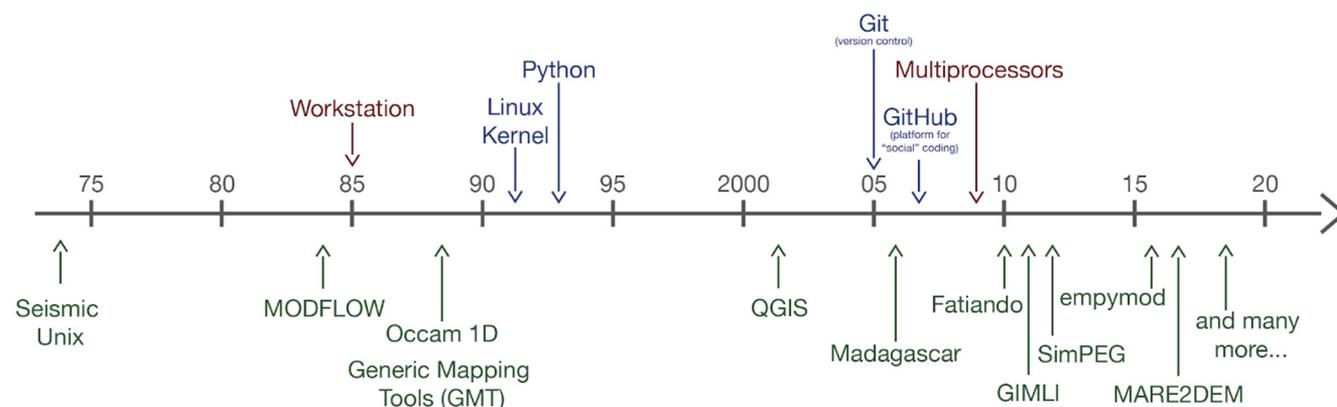


Figure 2. A sampling of open-source projects in geophysics through time.

collaboration on code and facilitates software engineering practices. Having a forum to discuss issues or changes to code, infrastructure to automatically run a test suite on a code base, and tools for packaging and distributing code as well as for launching web-based documentation enables collaboration among researchers and supports the development of communities around these projects. These tools and practices have all contributed to the growth of what we now consider the modern open-source ecosystem.

In addition to enabling technologies, social factors are also driving the open-source movement. These are illustrated by the renewed discussions on reproducibility and replicability in science, particularly in computational work (National Academies of Sciences, Engineering, and Medicine, 2019), as well as the establishment of principles such as the FAIR (findable, accessible, interoperable, and reusable) principle for scientific data management. Contributing to open-source software can also be an avenue for individuals to advance their careers and profiles; many researchers now list open-source projects to which they contribute in their curricula vitae or include links to sources such as a GitHub profile that show contributions through time. This is further evidence that open-source software is a valued part of the scientific conversation.

Much progress has been made in the open-source community, and we are now at a point where one can find open-source software packages in nearly every field and for a wide range of specialized tasks. There are even open-source projects to help navigate the available software for various scientific domains, including in the geosciences (Gosses et al., 2023).

Impacts of open-source software in applied geophysics

Assessing the “impact” of scholarly outputs, whether papers, software, data or otherwise, is a challenging objective, and there are a variety of strategies that can be adopted. Quantitative metrics such as number of citations or downloads can provide an indication. There are also studies that have sought to quantify the economic impacts of open-source software on jobs and gross domestic product (Blind et al., 2021). In what follows, we use examples to illustrate how open-source software can impact research. Rather than making an argument based on some metric for the value of open source, our goal is to spark ideas for where and how researchers can have an impact by adopting open-source practices.

Looking to other fields, we can see some places where the use of open-source software has become a standard. Notably, much of the growth of ML research has been facilitated by tools like scikit-learn, PyTorch, and TensorFlow that provide a base layer for that research. Additionally, it has become more the norm rather than the exception that code is made available alongside publications in ML research. Neural networks can be complex, and even with a well-written methods section in a paper, it is generally not straightforward to reproduce and extend work. Seeing the code authors have written (even if it is messy) provides a starting point and can save time that would have otherwise been spent reinventing the same work.

Having access to source code to reproduce a given result is particularly important in methods-oriented research where the approach for solving a problem is the scientific contribution. This is true when designing a neural network and when simulating

complex equations or solving inverse problems in geophysics. Inverse problems in particular are inherently nonunique, and the solution that is obtained depends on the implementation and the choices made by the user.

There are multiple open-source projects for simulating and inverting geophysical data (e.g., Key, 2016; Rücker et al., 2017; Blanchy et al., 2020; Biondi et al., 2021; Mardan et al., 2023), and for working with geophysical data more broadly (e.g., Uieda et al., 2013; Krieger and Peacock, 2014; Stanton et al., 2016; Warren et al., 2016; Louboutin et al., 2019). Each of these projects undoubtedly has examples that could be shared that illustrate impacts on research in geophysics. Our focus in the following sections is on SimPEG because it is the project that we work on and have context with.

We started the SimPEG project to develop a framework and toolbox for solving inverse problems in geophysics (Cockett et al., 2015). Promoting transparency and reproducibility, facilitating collaboration, and enabling researchers to rapidly experiment with the work of others were prime motivating factors to start SimPEG as an open-source project. In the following material, we illustrate some of the ways in which we have seen research accelerated because of the open-source principles and practices used within the SimPEG community.

Accelerated research, education, and outreach in the SimPEG community

SimPEG solves finite-volume numerical simulations and gradient-based inversions. It uses the permissive MIT license, which allows for academic and commercial use and adaptation. It is designed to be modular so that pieces can be interchanged and extended independently of each other but be able to work together.

The general inverse problem solved by SimPEG is an optimization problem that consists of a data misfit (ϕ_d) and a regularization term (ϕ_m) with a trade-off parameter (β) between them, as shown in equation 1:

$$\underset{m}{\text{minimize}} \phi(m) = \phi_d(m) + \beta \phi_m(m) \quad \text{s.t. } \phi_d \leq \phi_d^* \quad (1)$$

Solving this problem requires the ability to simulate geophysical data, compute sensitivities with respect to model parameters, define a regularization, and implement heuristics such as a β -cooling strategy to obtain a solution that fits the data (Oldenburg and Li, 2005).

A conceptual diagram that illustrates how we organize the components of the inverse problem is shown in Figure 3. The measured data, an estimate of the uncertainties, knowledge of the governing physical equations, and a priori information and assumptions about the geologic question are inputs to the inversion. In the implementation, we then arrange the various components as modules that are combined to solve the inverse problem. First, we need a mechanism to simulate data. For this, we take a finite-volume approach, meaning that a *mesh* (e.g., a tensor or Octree mesh) is designed and provided to the *simulation*, which has the necessary components to solve the equations of interest (e.g., time-domain electromagnetics or gravity). The *survey* contains the information about the sources and receivers (type and locations)

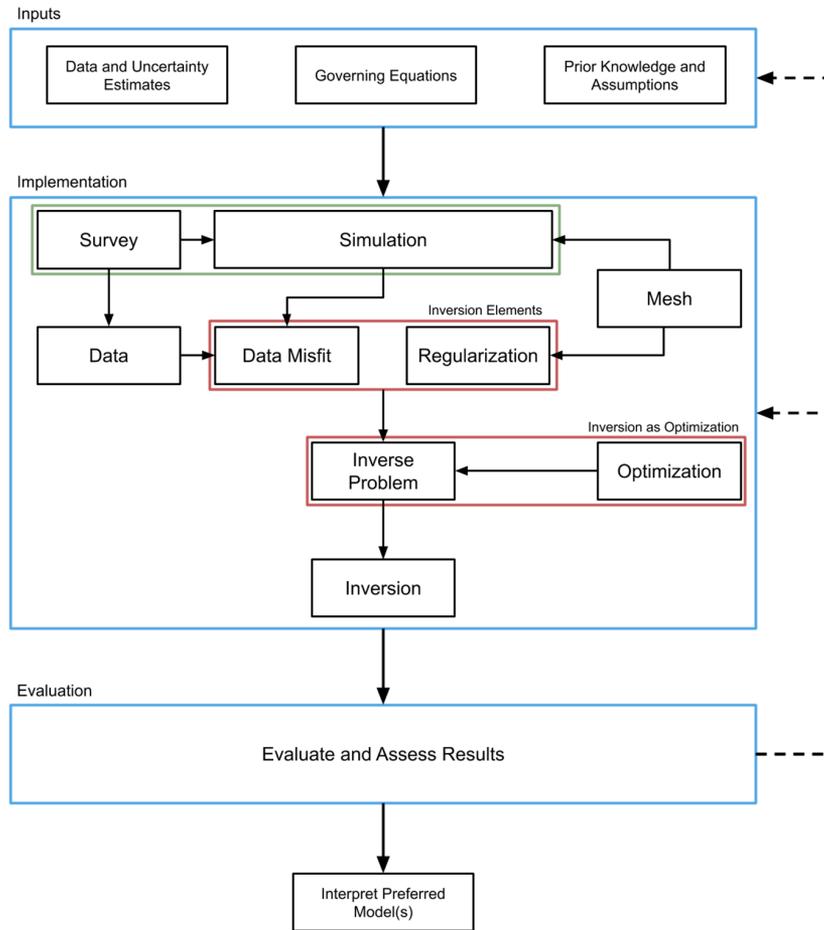


Figure 3. Framework for solving inverse problems used by SimPEG. After Cockett et al. (2015).

that are employed. With this information provided, we can then simulate predicted data. Note that the simulation also contains machinery to compute sensitivities (or products of a vector with the sensitivity) for gradient-based optimization.

To perform the inversion, we then define the observed *data* and construct the *data misfit* and *regularization* terms that are combined to define the objective function for the *inverse problem* (equation 1). We choose an *optimization* strategy that will be used for minimizing the objective function (e.g., inexact Gauss–Newton). The *inversion* module brings together all of these components along with additional instructions (that are called *directives* in SimPEG), such as the use of a β -cooling schedule or a stopping criterion based on a target misfit value, that are enacted when the inversion is run.

SimPEG’s modular framework is common across multiple geophysical techniques. This promotes interactions between researchers having different backgrounds and expertise. Further, it provides an ideal environment for integrating multiple geophysical data sets (e.g., joint inversions) and experimenting with components of the inversion (e.g., regularization).

SimPEG is developed and maintained by a community of contributors that includes graduate students, researchers, and industry professionals. The development follows modern open-source practices including automated testing, code reviews via pull requests, and discussion about improvements through GitHub issues and in regular open meetings. The combination of the code

base, the community, and the practices adopted by the community are what we refer to as the SimPEG project.

In the following sections, we show three examples from the SimPEG community that were enabled by the open-source nature of the project. The first example illustrates the value of a modular framework by showing the reuse of methods implemented initially for one geophysical method — magnetics — for a different application — time-lapse electromagnetic imaging. The second example discusses joint inversions, which are enabled by having multiple methods implemented in a common framework. The final example shows how research code can be repurposed for educational and outreach purposes. In each of these examples, ideas have been exchanged, and built upon, more rapidly than would have been possible if the code and community of geoscientists supporting that software were working in isolation.

Regularization with sparse and compact norms

The first example we present focuses on the choice of regularization in the inverse problem. As the inverse problem is ill-posed, the regularization plays an

important role in allowing us to incorporate existing geologic knowledge and/or assumptions about the model into the inversion.

The regularization term consists of a smallness term and first-order smoothness terms that penalize gradients in each of the spatial dimensions, as shown in equation 2:

$$\begin{aligned} \phi_m(m) = & \alpha_s \int_V w_s |m - m_{\text{ref}}|^{p_s} dV + \alpha_x \int_V w_x \left| \frac{\partial m}{\partial x} \right|^{p_x} dV \\ & + \alpha_y \int_V w_y \left| \frac{\partial m}{\partial y} \right|^{p_y} dV + \alpha_z \int_V w_z \left| \frac{\partial m}{\partial z} \right|^{p_z} dV. \end{aligned} \quad (2)$$

The first term is referred to as the “smallness” term and penalizes differences between the model (m) and a reference model (m_{ref}) and the three other terms are often referred to as “flatness” or “first-order” smoothness terms in the x, y, z directions as they penalize spatial derivatives. The influence of each of these terms in the regularization is weighted by a scalar parameter, $\alpha_{s,x,y,z}$. The $p_{s,x,y,z}$ values indicate the choice of norm. Weights, such as sensitivity weighting, can be applied in the regularization using the $w_{s,x,y,z}$ terms.

A standard inversion uses ℓ_2 norms (e.g., $p_{s,x,y,z} = 2$), resulting in the recovery of smooth structures. The work in Fournier and Oldenburg (2019) develops the approach that is used in SimPEG for solving problems where $p_{s,x,y,z} < 2$. This enables us to recover models with sharp transitions at geologic interfaces and compact

structures by promoting sparsity in each term in the regularization. This approach uses iteratively reweighted least squares (IRLS) in performing the optimization. To overcome challenges in optimizing nonconvex solutions introduced by using sparse norms ($p_{s,x,y,z} < 2$), this method first seeks an ℓ_2 norm solution and then finds an ℓ_p norm solution.

The initial implementation by Fournier et al. (2020) was tested on potential fields data collected over the Kevitsa deposit in Finland. The results shown in Figure 4 were obtained from a magnetic vector inversion (MVI) parameterized in spherical coordinates. Regularization is applied to each of the three terms describing the vector magnetization (e.g., repeating equation 2 for the amplitude and two angles of magnetization). To promote coherent magnetization directions, an approximate ℓ_0 norm is applied to the smallness and smoothness terms on the angles of the magnetization. To explore the model space, the norms applied to the amplitude of the magnetization are varied. Figure 4 shows north–south cross sections through two recovered models at the center of the known nickel deposit. The top image is the result when ℓ_2 norms are applied to the smallness and smoothness terms on the amplitude while the bottom result shows when an approximate ℓ_0 norm is applied to the smallness term on the amplitude. Using an approximate ℓ_0 norm on the orientation and amplitude of the magnetization enables the recovery of more distinct structures, each with a relatively uniform magnetization.

An interesting adaptation of the use of sparse norms was presented by Kang and Knight (2022) for time-lapse inversion of airborne electromagnetic data in a saltwater intrusion problem. SkyTEM data were collected in 2017 and 2019, and the aim is to use both data sets to infer where seawater intrusion may be happening. Performing a 1D spatially constrained inversion of the 2017 data results in the model shown in the top left of Figure 5; ℓ_2 norms were applied in the vertical and lateral regularizations. Simply inverting the 2019 data following the same setup and taking the difference between the two recovered models results in the image shown in the top center of Figure 5. However, inversions are nonunique. Without including any additional information, there are many differences between the two models, making it challenging to interpret.

In this work, they also adapt the inverse problem to include a constraint in the time dimension of the form,

$$\phi_t(m) = \alpha_t \int_V \omega_t \left| \frac{\partial m^{\ell_t}}{\partial t} \right| dV. \quad (3)$$

This equation takes the same form as the first-order smoothness terms in equation 2, but now we are comparing model changes along the time dimension. Using an ℓ_2 norm constraint in time results in the image shown in the top

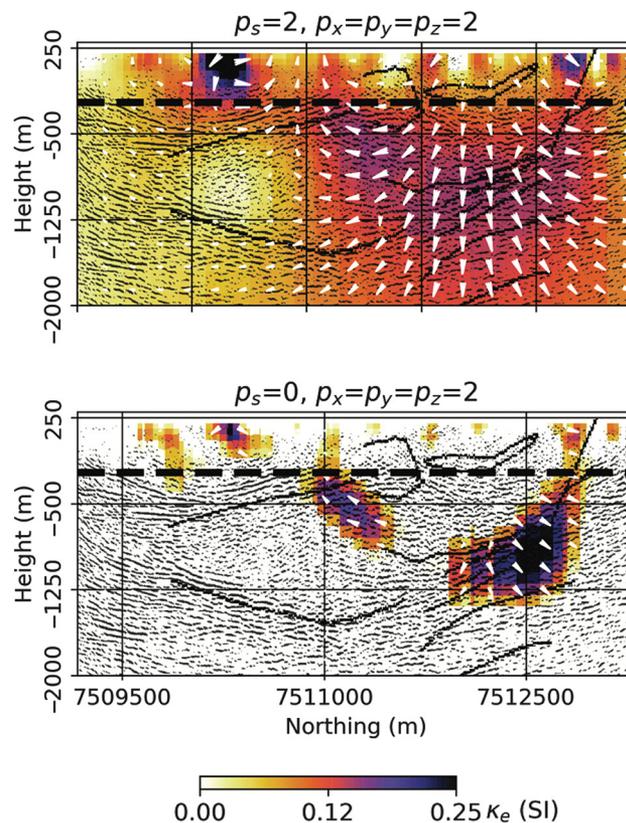


Figure 4. Cross section through recovered models from MVI inversions performed at the Kevitsa deposit in Finland using two different choices of norms (Fournier et al., 2020). Geologic interpretations of a 2D seismic reflection survey are shown in black for reference.

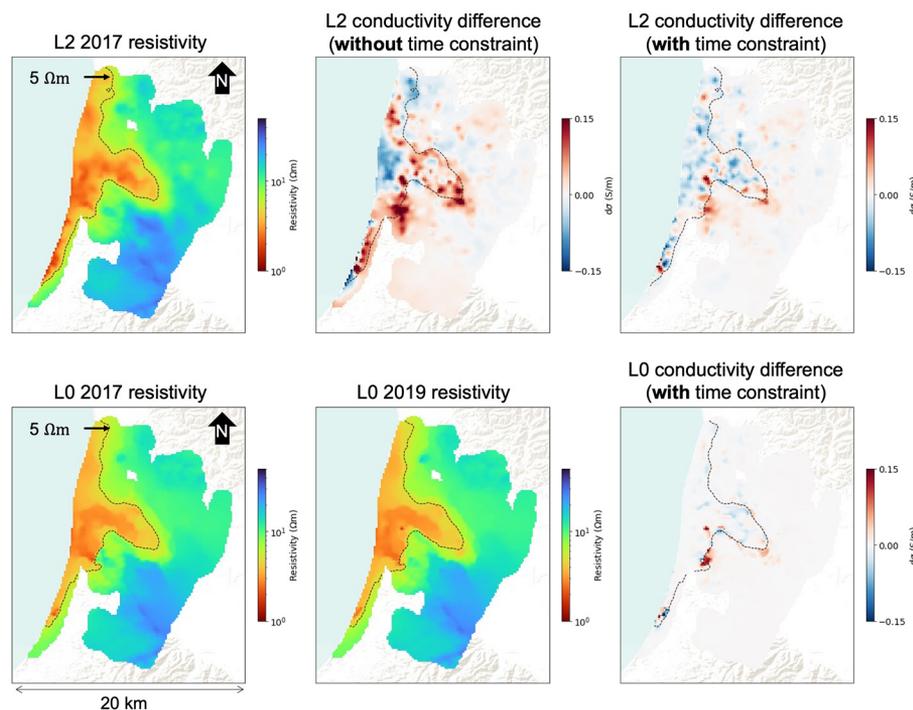


Figure 5. Depth slices through inversion results of airborne time-domain electromagnetic data to assess seawater intrusion in California (Kang and Knight, 2022). The top row shows ℓ_2 inversion results. The bottom row shows inversion results that make use of sparse and compact norms for both the spatial regularization and the time constraint. The black line on each image is a 5 Ohm-m contour indicating a potential interface between saltwater and freshwater.

right of Figure 5. There are fewer changes between the two years due to imposing the assumption that there should be smooth changes in time.

Hypothesizing that changes of a model value through a time interval (a year) should be sparse, with most of the model remaining constant through time and the changes in conductivity being localized at the saltwater-freshwater interface, the authors impose an approximate ℓ_0 norm in time. Approximate ℓ_0 norms are also applied laterally and vertically to promote sharp transitions of conductivity values at the saltwater-freshwater interface. The bottom row of Figure 5 shows the recovered models for 2017 and 2019 along with the difference between the two when ℓ_0 norms are used. By using an approximate ℓ_0 norm in time, the changes between the two years can be isolated to a few specific areas. In the recovered model, the difference between the two models is localized to a few specific areas, specifically the red regions near the interpreted freshwater-seawater interface (black line). This may be more realistic in practice and can enable decisions to be made about groundwater management in those areas (e.g., regulating groundwater extraction in those regions).

In the aforementioned magnetic and electromagnetic examples, the same regularization module is used for different geophysical simulations. Adapting from one physics (magnetics) to another physics (electromagnetics) requires the tuning of several hyperparameters in the regularization and IRLS algorithms. In the electromagnetic problem, the time constraint is added as a straightforward extension to the regularization. There are also additional complexities because the data are not collected in exactly the same location across the two years. Importantly, this is where a researcher can spend their time and efforts, as opposed to having to reimplement components from scratch as would have been required if the code bases for magnetics and electromagnetics were closed or developed as monoliths. Additionally, the practices in place for peer review — which promotes clarity of code — and documentation — which provides instructions and examples for usage — facilitate more rapid adoption, extension, and reuse of ideas implemented in the code base.

Joint inversions

The next area of research we consider is the joint inversions of multiple geophysical data sets. Given multiple data sets, a common goal is to produce a model of the subsurface that is consistent with all of those data sets. Joint inversions can be a tool for helping solve this problem if an appropriate relationship between the different physical properties and data can be defined. In a general joint inversion, the inverse problem is adapted to include multiple data misfit terms, multiple regularizations, and a coupling term (ϕ_c) that imposes some similarity measure between the multiple physical property models:

$$\begin{aligned} \underset{m}{\text{minimize}} \quad \phi(m) &= \chi_1 \phi_{d1}(m_1) + \chi_2 \phi_{d2}(m_2) + \beta_1 \phi_{m1}(m_1) + \beta_2 \phi_{m2}(m_2) + \gamma \phi_c(m_1, m_2) \\ \text{s.t.} \quad \phi_{d1} &\leq \phi_{d1}^*, \phi_{d2} \leq \phi_{d2}^* \end{aligned} \quad (4)$$

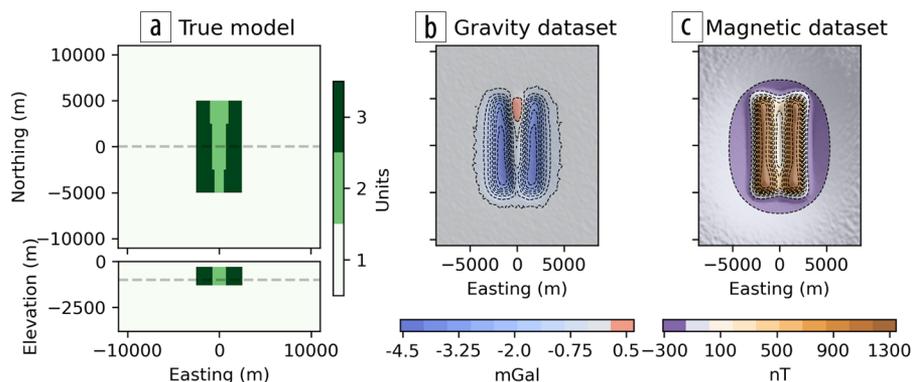


Figure 6. (a) Synthetic model for the carbon mineralization example. The serpentinized unit, unit 3, has a density of 2.7 g/cc, and 0.15 SI susceptibility. The carbonated unit, unit 2, has a density of 3.0 g/cc and 0.05 SI susceptibility. The background, unit 1, has a density of 2.9 g/cc and 0 susceptibility. (b) Gravity and (c) magnetic data are simulated on a 19 by 21 km grid with 250 m grid spacing. After Capriotti et al. (2023).

There are different flavors of joint inversions: multiple data sets may be sensitive to the same physical property (e.g., DC resistivity and electromagnetics), or the physical properties that each data set is sensitive to may be distinct. In the latter case, we then may choose a coupling term that imposes some structural similarity, such as cross-gradient or joint-total variation (Gallardo and Meju, 2003; Haber and Holtzman Gazit, 2013), or a petrophysical constraint, such as fuzzy c -means (Sun and Li, 2016) or the petrophysically and geologically guided inversion (PGI) (Astic and Oldenburg, 2019).

For example, both magnetic and gravity data can be diagnostic in identifying ultramafic rocks with carbon mineralization potential. Ultramafic rocks that have been serpentinized can react with CO_2 to produce carbonate minerals. As compared to fresh ultramafic or carbonated units, serpentinized rocks have a low density and generally higher magnetic susceptibility. The works in Heagy et al. (2022) and Capriotti et al. (2023) use synthetic examples to explore the use of joint inversions for delineating serpentinized rocks. Figure 6 illustrates the model that is considered. The model has a serpentinized unit (dark green; unit 3), with a central region that has been carbonated (light green; unit 2).

Figure 7 illustrates the use of three different joint inversion approaches for the inversion of gravity and magnetic data in a synthetic example for carbon mineralization (Capriotti et al., 2023). The top row shows: the recovered density model (Figure 7a), the recovered susceptibility model (Figure 7b), and a crossplot of the recovered physical properties for a cross-gradient inversion (Figure 7c). The stars in Figure 7c show the physical properties corresponding to the three rock units from Figure 6. The second and third rows show joint inversion results using joint total variation and the PGI, respectively. Each of the joint inversion approaches promotes different structures and features in the physical property crossplots. Cross-gradient promotes co-located nonzero gradients in the model, and in the

physical property crossplot, we can see that it promotes linear features. Joint total variation promotes joint sparsity (e.g., having nonzero values located in the same spatial location). PGI promotes clusters in physical property space. Each approach has advantages and settings where it is more or less well-suited for the problem, but being able to compare multiple strategies enables us to build confidence in common features of the models and also to understand the features one approach tends to promote.

Having a common framework for each physical simulation provides the components for implementing a joint inversion. In SimPEG, to form a combined data misfit, the user simply creates each data misfit term as they would for individual inversions and sums them together (e.g., in code: $\text{phi_d} = \text{chi_1} * \text{phi_d1} + \text{chi_2} * \text{phi_d2}$). There are a number of practical challenges in obtaining a good result from a joint inversion, including

choosing a coupling term, weighting each of the terms in equation 4 appropriately, and designing a strategy for updating parameters as the inversion runs to fit each of the data sets adequately. These are nontrivial and require experimentation by the user. But prior to being able to experiment with approaches, all of the necessary components including the OcTree mesh and operators, implementation of each of the forward simulations, the optimization machinery, etc., must be in place. Each of these components requires different skills and expertise and development can be accelerated through collaborations among researchers with different skill sets. In fact, generating the results shown in Figure 7 required the use of code written by 10 different people from four different academic and industry institutions. Each contributor to this collaborative project brings their own goals and interests. This scale and style of organic collaboration is

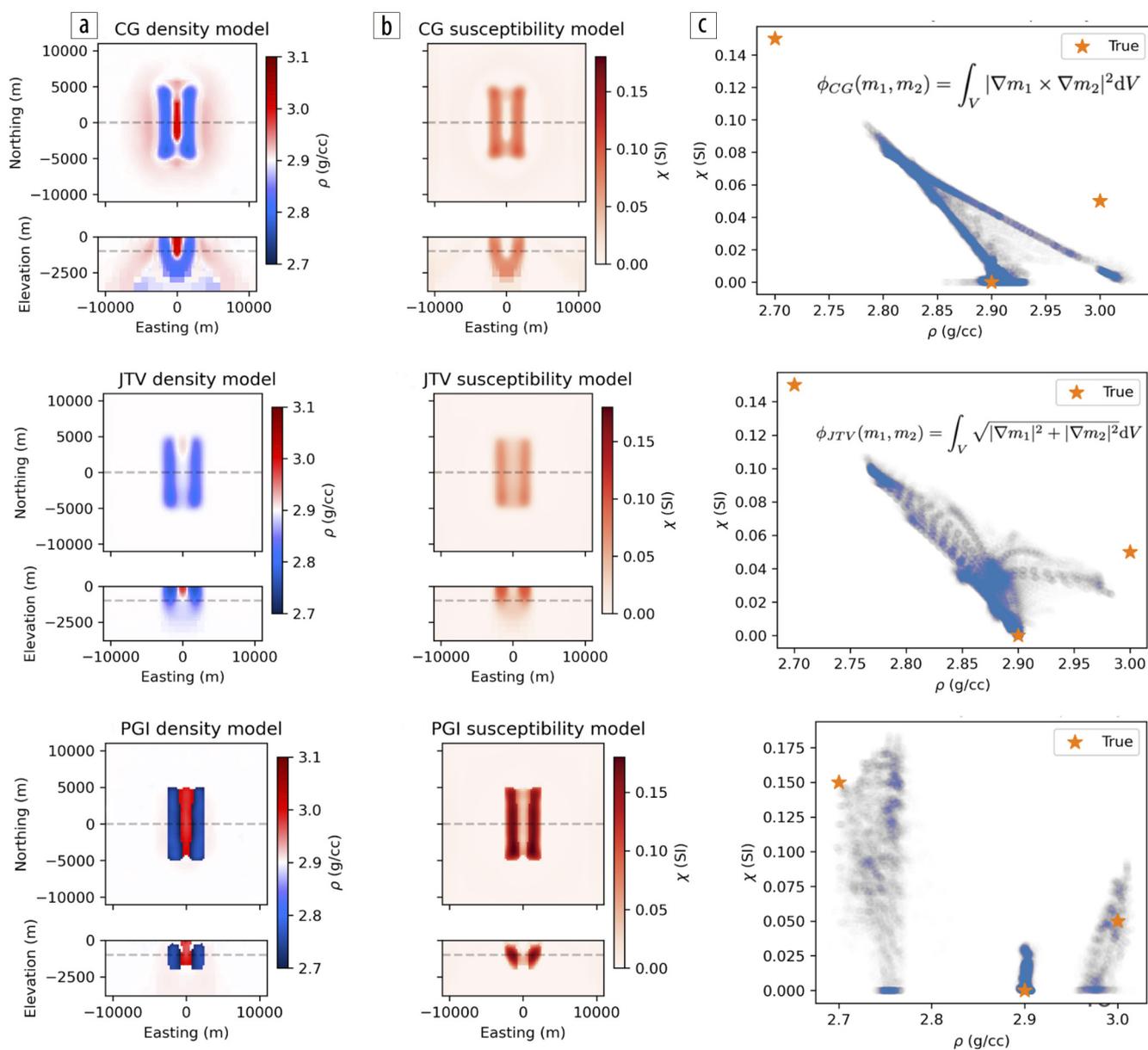


Figure 7. Joint inversions of synthetic gravity and magnetic data for a carbon mineralization application. The central unit of carbonated rock (high density and moderate susceptibility) is embedded in a serpentinized unit (low density and high magnetic susceptibility). After Capriotti et al. (2023).

possible when researchers derive value from and choose to invest efforts in a common set of open-source tools.

Education and outreach

The final area we discuss is the role of educational resources in promoting the use of geophysics. In many cases, the decision to use geophysics (or not) for a given application is made by an engineer, a manager, or someone else who is not themselves a geophysicist. Resources that illustrate the use of geophysics and fundamental concepts about the various geophysical methods can be useful tools for communicating the role that geophysics can play in helping solve an applied problem.

In our experience, we have found that using simulations and enabling others to change various input parameters and visualize outputs has been an impactful way to illustrate fundamental concepts. For the 2017 SEG Distinguished Instructor Short Course, we developed a collection of Jupyter Notebook “apps” in which users could interact with simulations based on SimPEG through widgets (e.g., slide bars, toggle buttons, etc.) (Oldenburg et al., 2021). Similarly, we designed Jupyter Notebook apps for a Geoscientists *Without Borders* (GWB) project aimed at using DC resistivity to decide where to drill for groundwater in Mon State, Myanmar (Fan et al., 2020). We created two styles of apps. The first collection of apps was aimed at facilitating understanding of fundamental concepts, such as how the presence of a conductive or resistive layer alters the flow of currents and therefore the measured potentials at the surface of the earth. The second collection of apps provides template workflows for loading, visualizing and inverting data to obtain a resistivity model using SimPEG. An example of one of the apps is shown in Figure 8.

An additional benefit of building educational resources on top of open-source research tools is that these resources can provide an on-ramp for participants to use these tools in their own work. For the GWB project, participants used the notebooks to identify more than 20 locations to drill for groundwater after the North American team left Myanmar. Educational uses of the SimPEG

software have also led to improvements in the code base. For example, efficiency improvements were made to the DC resistivity code as described in Kang et al. (2020) to support geoscientists running inversions on older laptops with limited RAM. These improvements are equally valuable for inverting large-scale data sets. Having a common set of tools and resources where research code is repurposed for education, and vice versa, drives improvements that benefit both use cases. Importantly, it also provides avenues for “students” to become contributors.

Takeaways

These three examples illustrate how open-source software can enable collaboration, facilitate the exchange and extension of ideas, and promote the use of geophysics to help solve applied problems. In our experience, these examples may not have been possible, or would have required substantially more time and effort, without open-source software and communities. In particular, the use of a permissive license has been key for promoting engagement and contributions from geoscientists in industry as code with a permissive license can be included in commercial products. These connections drive improvements and use in new applications by industry practitioners.

In addition to the licensing terms, SimPEG is a project supported by a community of contributors and the set of open-source practices used in this community have implications on its growth and impact. For example, having comprehensive tests for the code base enables collaborations; it allows others to propose improvements to the code without the fear of breaking it. “Pull requests” facilitate peer review of code; this process can help catch unhandled edge cases that the original author may not have considered or result in changing the names of variables for clarity and maintainability of the code. Documentation and tutorials help users get started with the code, and online forums and meetings provide a space for discussion about both the software and the scientific problems it is being used to address. These components of social infrastructure are key for scalable collaboration.

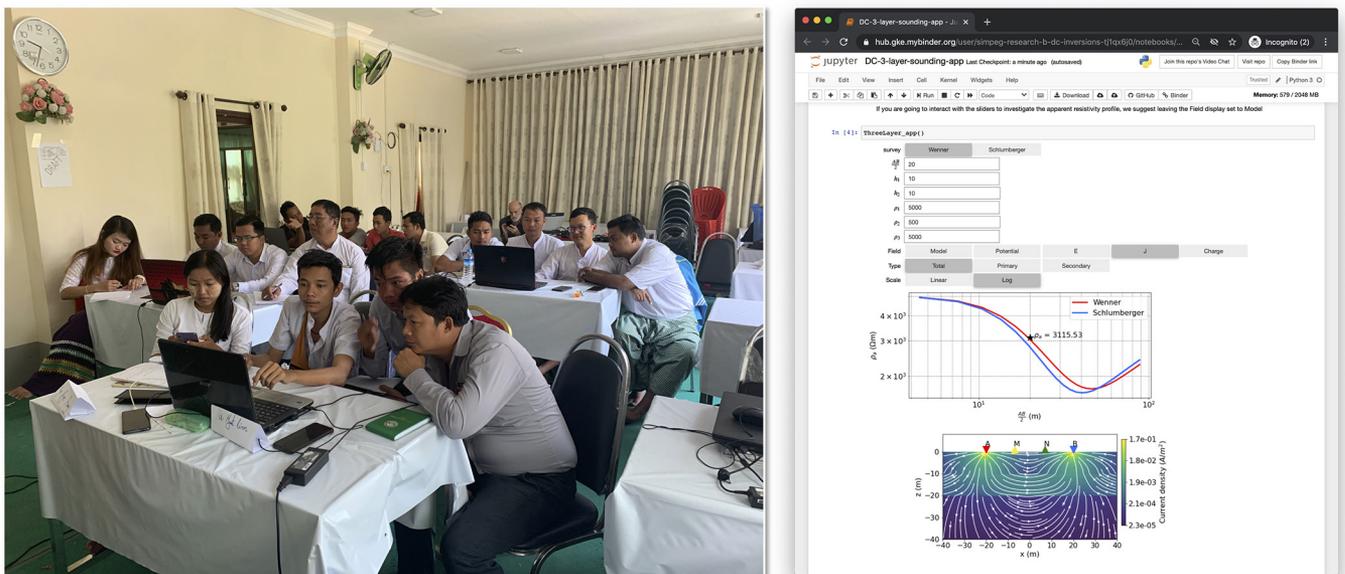


Figure 8. Participants in Mon State, Myanmar, using Jupyter Notebook apps to explore fundamental concepts of DC resistivity as a part of a 2019 GWB project (Fan et al., 2020).

Relationships and partnerships between industry and academic groups are also evolving in light of the shift to open source. Much research has been funded under a model where companies sponsor research groups that produce software that is proprietary to sponsors. An open-source model has several advantages compared to a closed-source sponsorship. Research and development are conducted in the same project where “production” code is deployed, reducing the time and effort necessary for technology transfer. It also opens new opportunities for collaboration between industry practitioners and academic researchers because both groups can suggest changes to the code and make improvements as it is used. Importantly, highly qualified scientists are trained in developing and contributing to the same code base that they will use in their careers. Our experience has shown that there is now a critical mass of industry partners who see value in these aspects and are willing to support this model of partnership going forward.

There are now many open-source projects in geophysics following similar practices and solving common problems. There is some overlap between the scope and purpose of projects. Although this could be viewed as a duplication of efforts, we view this as an advantage, particularly for complex problems because different groups take different approaches and handle some implementation details differently. Studies that compare solutions can help build confidence in a result and differences can be investigated when code is available. For example, works such as Werthmüller et al. (2021) compare forward simulations of the 3D Maxwell’s equations, and Doyoro et al. (2022) compare DC resistivity inversion results between four different inversion codes. These types of comparisons are valuable resources for building confidence in different code bases and illustrating where differences in the details of a given approach impact the results.

We often refer to an open-source project existing within an “ecosystem,” referring to a collection of projects that are interconnected and can build upon each other. For example, SimPEG relies upon several core scientific Python packages and leverages external packages for high-performance computing such as Dask and Numba. This modular, interoperable ecosystem model has several advantages: it allows for the separation of concerns among the various projects so each can remain scoped, and when improvements are made in one project, all groups that rely on that project benefit from those advancements. Improvements in performance and efficiency of code are natural examples. Additionally, when we consider the broader research landscape, an ecosystem model allows us to interoperate with advancements made in other fields. There is much potential and value already being derived from the use of open-source software in ML. For example, the PGI approach discussed earlier relies upon tools in the scikit-learn package (Pedregosa et al., 2011) that has tools for classic ML. Tools for deep learning such as PyTorch and Tensorflow exist within the same ecosystem. Combining methods from ML and deep learning with geophysical methods is an area ripe for innovation and exploration, and this is aided by having accessible, interoperable tools.

Looking to the future

Many of the practices from open source can be applied at different scales and different stages of research. Not every research

project should turn into an “open-source package” that is maintained over time. However, the practice of sharing code promotes transparency, enables deeper conversations about the methods employed, and facilitates reuse and extensions of work. This is true even for unpolished code. In our experience, planning to share code alongside a publication has prompted better software practices such as improving variable names or reducing duplicate code, because we know our peers may look at it. This positive social pressure can promote the development of more reliable research code.

Beyond software, open communication of science broadens conversation to “open science.” UNESCO (2023) defines open science as “a set of principles and practices that aim to make scientific research from all fields accessible to everyone for the benefits of scientists and society as a whole.” An emphasis in the UNESCO recommendations is that research be “as open as possible,” recognizing that there are scenarios where access to data or intellectual property may need to be restricted. In addition to software, open science encompasses ideas such as the FAIR principles for data and sharing publications in an openly accessible way, for example posting preprints of publications on servers such as arXiv or EarthArXiv. Taking steps to make research outputs accessible facilitates transparency, reproducibility, and ultimately extension of ideas. These concepts are core to how science progresses.

In this paper, we illustrated the role of open-source software and practices in three examples: reusing regularization methods with different geophysical surveys, combining simulations to perform joint inversions, and repurposing research code for use in educational and humanitarian projects. Reuse and extension are enabled because the code is freely available for use and adaptation, and importantly because the set of practices around development and communication facilitates collaboration among a community of researchers with different backgrounds, skill sets, and interests. Having input and contributions from multiple people with different strengths accelerates development. Having the developed methods applied to a variety of applied problems and in a range of contexts, including research, education, and industrial applications, improves reliability and robustness, which in turn allows further reuse in other domains.

The applications where geophysics can make a positive impact in society are complex and require data, knowledge, and methods from multiple disciplines. We have experienced how open-science practices have facilitated rapid reuse and extension of work and enabled the types of collaborations across disciplines that are necessary for solving complex problems. These open-science practices are most effective when adopted by a community of researchers and practitioners. Such a community will have immense potential to accelerate the use and effectiveness of geophysics in helping reach solutions to the complex geoscientific problems that society faces. ■■■

Acknowledgments

The authors are grateful to Yaoguo Li and Michael Wilt for organizing this special section and for their feedback on an earlier version of the manuscript. We thank the anonymous reviewer who provided constructive feedback on our manuscript. We are

also grateful to members of the SimPEG community for their contributions to the code base and culture of the community.

Data and materials availability

Data associated with this research are available and can be accessed via <https://doi.org/10.5281/zenodo.596373>.

Corresponding author: lheagy@eoas.ubc.ca

References

- Astic, T., and D. W. Oldenburg, 2019, A framework for petrophysically and geologically guided geophysical inversion using a dynamic Gaussian mixture model prior: *Geophysical Journal International*, **219**, no. 3, 1989–2012, <https://doi.org/10.1093/gji/ggz389>.
- Biondi, E., G. Barnier, R. G. Clapp, F. Picetti, and S. Farris, 2021, An object-oriented optimization framework for large-scale inverse problems: *Computers & Geosciences*, **154**, 104790, <https://doi.org/10.1016/j.cageo.2021.104790>.
- Blanchy, G., S. Saneiyan, J. Boyd, P. McLachlan, and A. Binley, 2020, ResIPy, an intuitive open source software for complex geoelectrical inversion/modeling: *Computers & Geosciences*, **137**, 104423, <https://doi.org/10.1016/j.cageo.2020.104423>.
- Blind, K., M. Böhm, P. Grzegorzewska, A. Katz, S. Muto, S. Pätsch, and T. Schubert, 2021, The impact of open source software and hardware on technological independence, competitiveness and innovation in the EU economy — Final study report: Publications Office of the European Union, <https://doi.org/10.2759/430161>.
- Capello, M. A., 2022, President's Page: Advancing sustainability in SEG: *The Leading Edge*, **41**, no. 1, 6–7, <https://doi.org/10.1190/tle41010006.1>.
- Capello, M. A., A. Shaughnessy, and E. Caslin, 2021, The Geophysical Sustainability Atlas: Mapping geophysics to the UN Sustainable Development Goals: *The Leading Edge*, **40**, no. 1, 10–24, <https://doi.org/10.1190/tle40010010.1>.
- Capriotti, J., L. J. Heagy, and S. Soler, 2023, Joint inversions with the SimPEG framework: Third International Meeting for Applied Geoscience & Energy, SEG/AAPG, Expanded Abstracts, 1678–1682, <https://doi.org/10.1190/image2023-3910791.1>.
- Claerbout, J. F., and M. Karrenbach, 1992, Electronic documents give reproducible research a new meaning: 62nd Annual International Meeting, SEG, Expanded Abstracts, 601–604, <https://doi.org/10.1190/1.1822162>.
- Cockett, R., S. Kang, L. J. Heagy, A. Pidlisecky, and D. W. Oldenburg, 2015, SimPEG: An open source framework for simulation and gradient based parameter estimation in geophysical applications: *Computers & Geosciences*, **85**, Part A, 142–154, <https://doi.org/10.1016/j.cageo.2015.09.015>.
- Constable, S. C., R. L. Parker, and C. G. Constable, 1987, Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data: *Geophysics*, **52**, no. 3, 289–300, <https://doi.org/10.1190/1.1442303>.
- Doyoro, Y. G., P.-Y. Chang, J. M. Puntu, D.-J. Lin, T. Van Huu, D. A. Rahmala, and M.-S. Shie, 2022, A review of open software resources in Python for electrical resistivity modelling: *Geoscience Letters*, **9**, 3, <https://doi.org/10.1186/s40562-022-00214-1>.
- Fan, K., D. W. Oldenburg, M. Maxwell, D. Cowan, S. Kang, L. J. Heagy, and J. Capriotti, 2020, Improving water security in Mon State, Myanmar via geophysical capacity building: 90th Annual International Meeting, SEG, Expanded Abstracts, 3355–3360, <https://doi.org/10.1190/segam2020-3428432.1>.
- Fournier, D., L. J. Heagy, and D. W. Oldenburg, 2020, Sparse magnetic vector inversion in spherical coordinates: *Geophysics*, **85**, no. 3, J33–J49, <https://doi.org/10.1190/geo2019-0244.1>.
- Fournier, D., and D. W. Oldenburg, 2019, Inversion using spatially variable mixed ℓ_p norms: *Geophysical Journal International*, **218**, no. 1, 268–282, <https://doi.org/10.1093/gji/ggz156>.
- Gallardo, L. A., and M. A. Meju, 2003, Characterization of heterogeneous near-surface materials by joint 2D inversion of dc resistivity and seismic data: *Geophysical Research Letters*, **30**, no. 13, <https://doi.org/10.1029/2003GL017370>.
- Gosses, J., J. Dramsch, E. Bianco, D. Werthmüller, A. Moodie, B. Sullivan, M. Niccoli, et al., 2023, softwareunderground/awesome-open-geoscience: Alpha release to test integration with Zenodo (Alpha): Zenodo, <https://doi.org/10.5281/zenodo.8354180>.
- Haber, E., and M. Holtzman Gazit, 2013, Model fusion and joint inversion: *Surveys in Geophysics*, **34**, no. 5, 675–695, <https://doi.org/10.1007/s10712-013-9232-4>.
- Heagy, L. J., T. Astic, J. Capriotti, J. Weis, and D. W. Oldenburg, 2022, Geophysical inversions to delineate rocks with CO₂ sequestration potential through carbon mineralization: *FastTIMES*, **26**, no. 4, <https://doi.org/10.48550/arXiv.2203.13894>.
- Kang, S., J. Capriotti, D. W. Oldenburg, L. J. Heagy, and D. Cowan, 2020, Open-source geophysical software development for groundwater applications: 90th Annual International Meeting, SEG, Expanded Abstracts, 1989–1993, <https://doi.org/10.1190/segam2020-3425913.1>.
- Kang, S., and R. Knight, 2022, Time-lapse inversion of airborne electromagnetic data for monitoring saltwater intrusion in the Salinas Valley of California, USA: Second International Meeting for Applied Geoscience & Energy, SEG/AAPG, Expanded Abstracts, 3125–3127, <https://doi.org/10.1190/image2022-3750766.1>.
- Key, K., 2016, MARE2DEM: A 2-D inversion code for controlled-source electromagnetic and magnetotelluric data: *Geophysical Journal International*, **207**, no. 1, 571–588, <https://doi.org/10.1093/gji/ggw290>.
- Krieger, L., and J. R. Peacock, 2014, MTPy: A Python toolbox for magnetotellurics: *Computers & Geosciences*, **72**, 167–175, <https://doi.org/10.1016/j.cageo.2014.07.013>.
- Langevin, C. D., J. D. Hughes, E. Banta, A. Provost, R. Niswonger, and S. Panday, 2017, MODFLOW 6, the U.S. Geological Survey modular hydrologic model: U.S. Geological Survey, <https://doi.org/10.5066/F76Q1VQV>.
- Louboutin, M., M. Lange, F. Luporini, N. Kukreja, P. A. Witte, F. J. Herrmann, P. Velesko, and G. J. Gorman, 2019, Devito (V3.1.0): An embedded domain-specific language for finite differences and geophysical exploration: *Geoscientific Model Development*, **12**, no. 3, 1165–1187, <https://doi.org/10.5194/gmd-12-1165-2019>.
- Mardan, A., B. Giroux, and G. Fabien-Ouellet, 2023, PyFWI: A Python package for full-waveform inversion and reservoir monitoring: *SoftwareX*, **22**, 101384, <https://doi.org/10.1016/j.softx.2023.101384>.
- National Academies of Sciences, Engineering, and Medicine, 2019, Reproducibility and replicability in science: The National Academies Press, <https://doi.org/10.17226/25303>.
- Oldenburg, D. W., L. J. Heagy, and S. Kang, 2021, Geophysical electromagnetics: A retrospective, DISC 2017, and a look forward: *The Leading Edge*, **40**, no. 2, 140–148, <https://doi.org/10.1190/tle40020140.1>.
- Oldenburg, D. W., L. J. Heagy, S. Kang, and R. Cockett, 2020, 3D electromagnetic modelling and inversion: A case for open source: *Exploration Geophysics*, **51**, no. 1, 25–37, <https://doi.org/10.1080/08123985.2019.1580118>.
- Oldenburg, D. W., and Y. Li, 2005, Inversion for applied geophysics: A tutorial, in D. K. Butler, ed., *Near-surface geophysics: SEG, Investigations in Geophysics No. 13*, 85–150, <https://doi.org/10.1190/1.9781560801719.ch5>.
- Open Source Initiative, 2007, The Open Source Definition (annotated), version 1.9, <https://opensource.org/definition-annotated/>, accessed 8 January 2023.

- Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al., 2011, Scikit-learn: Machine learning in Python: *Journal of Machine Learning Research*, **12**, no. 85, 2825–2830.
- Rücker, C., T. Günther, and F. M. Wagner, 2017, pyGIMLi: An open-source library for modelling and inversion in geophysics: *Computers & Geosciences*, **109**, 106–123, <https://doi.org/10.1016/j.cageo.2017.07.011>.
- Stanton, K. A., M. D. Sacchi, and N. Kazemi, 2016, Seismic data analysis in Julia: 78th Conference and Exhibition, EAGE, Extended Abstracts, <https://doi.org/10.3997/2214-4609.201601652>.
- Stockwell, J. W. Jr., 1999, The CWP/SU: Seismic Un*x Package: *Computers & Geosciences*, **25**, no. 4, 415–419, [https://doi.org/10.1016/S0098-3004\(98\)00145-9](https://doi.org/10.1016/S0098-3004(98)00145-9).
- Sun, J., and Y. Li, 2016, Joint inversion of multiple geophysical data using guided fuzzy c-means clustering: *Geophysics*, **81**, no. 3, ID37–ID57, <https://doi.org/10.1190/geo2015-0457.1>.
- Uieda, L., V. C. Oliveira Jr., and V. C. F. Barbosa, 2013, Modeling the earth with Fatiando a Terra: Proceedings of the 12th Python in Science Conference, 92–98.
- UNESCO, 2023, An introduction to the UNESCO recommendation on open science: UNESCO Digital Library, <https://doi.org/10.54677/XOIR1696>.
- Warren, C., A. Giannopoulos, and I. Giannakis, 2016, gprMax: Open source software to simulate electromagnetic wave propagation for ground penetrating radar: *Computer Physics Communications*, **209**, 163–170, <https://doi.org/10.1016/j.cpc.2016.08.020>.
- Werthmüller, D., R. Rochlitz, O. Castillo-Reyes, and L. Heagy, 2021, Towards an open-source landscape for 3D CSEM modelling: *Geophysical Journal International*, **227**, no. 1, 644–659, <https://doi.org/10.1093/gji/ggab238>.