OSPO Internship Projects Spring 2025

Project Descriptions

Project 1

The Cersonsky lab is looking for an OSPO intern to participate in the development of scikit-matter (https://github.com/scikit-learn-contrib/scikit-matter), a collection of machine learning utilities emerging from the materials science and chemistry communities. Interested interns will gain experience with state-of-the-art machine learning techniques developed by the lab and their applications in atomistic simulation, and they will work closely with a researcher with a background in atomistic machine learning. The primary responsibility will be integrating these techniques into the scikit-matter platform, which will involve submitting and managing pull requests, creating examples with materials science datasets, and updating documentation. Since scikit-matter is designed to be compatible with the widely used machine learning library scikit-learn, the intern will also learn the scikit-learn API and become familiar with the scikit-learn codebase as well. There will also be opportunities for exposure to the current research taking place in the lab, including topics in molecular crystallization, glassy dynamics, battery electrolyte design with ionic liquids, and machine learning representations of materials.

Intern needs: To be successful, interns should have an interest in computational materials science, molecular simulation, and machine learning. Experience with git/GitHub, python (scikit-learn, numpy, scipy) are a plus but are not required.

Project 2

clubSandwich (https://github.com/jepusto/clubSandwich) is a popular R package (in the top 10 percent of all R package downloads) that provides several cluster-robust variance estimators for ordinary and weighted least squares linear regression models, two-stage least squares regression models, and generalized linear models. Several adjustments are incorporated to improve small-sample performance. The package follows an object-oriented design to allow for straight-forward extension to other classes of regression models. Several models are not currently supported but are on the roadmap. An intern would be responsible for 1) learning about a statistical model that is not currently supported, 2) developing required methods so that the clubSandwich works for the model, 3) writing unit tests and examples to verify correct implementation.

Intern needs: R programming, knowledge of statistical theory at advanced undergraduate level

Project 3

lmeInfo (https://github.com/jepusto/lmeInfo) is an R package that provides analytic derivatives and information matrices for fitted linear mixed effects (lme) models and generalized least squares (gls) models estimated using nlme::lme() and nlme::gls(), respectively. The package includes functions for estimating the sampling variance-covariance of variance component parameters using the inverse Fisher information. The package does not currently provide methods for calculating degrees of freedom that are standard in other software packages (such as SAS) but this is a top priority on the roadmap for the project. An intern would be responsible for 1) developing functions that calculate degrees of freedom for the supported models, following known formulas and 2) writing unit tests and examples to verify correct implementation.

Intern needs: R programming, matrix algebra, knowledge of statistical theory at advanced undergraduate level

Project 4

scdhlm (https://github.com/jepusto/scdhlm) and SingleCaseES (https://github.com/jepusto/SingleCaseES) are R packages that provide specialized effect size calculation methods used in certain areas of social science research. Both packages include interactive web apps written in Shiny. The web apps allow users who are not comfortable with R to use the functionality of the packages via a point-and-click interface. The apps also automatically generate R code for reproducing calculations carried out interactively. The web apps do not currently have adequate test suites. An intern would be responsible for developing a set of robustness tests (using the Shinytest2 system) that exercise the main functionality of the web apps and verify that the results and auto-generated R code are correct.

Intern needs: R programming

Project 5

The JuliaPhylo project is a github organization (https://github.com/juliaphylo), offering tools for phylogenetic analysis in Julia. Its core package is PhyloNetworks.jl (https://github.com/JuliaPhylo/PhyloNetworks.jl), which was built to handle phylogenetic trees and networks. Other packages offer tools to visualize phylogenies (PhyloPlots.jl), to infer phylogenies from genetic data (SNaQ.jl), to analyze the evolution of traits along phylogenies (PhyloTraits.jl), and more. Developers of these packages are currently collaborating on a major refactoring effort, towards a smaller core package PhyloNetworks and a more modular ecosystem.

The intern will assist in creating a website for the Julia organization. This website will explain the ecosystem of packages, provide links to resources for users, guidance to contributed new code for developers, etc. The intern will also assist with the development of Julia packages and the implementation of new tools, such as a recently described distance to compare semidirected

phylogenetic networks (https://arxiv.org/abs/2405.16035), or a refactoring of the search for the most parsimonious network, with relaxed assumptions compared to the existing implementation.

Intern needs: To be successful, the intern should have prior experience with version control and git/github (github pages / website would be a plus), and experience with some object-oriented programming language and principles of software development. Experience with Julia is highly desirable, but not required. Prior knowledge about biology or phylogenetics is *not* required.

Project 6

FaaS (function-as-a-service) platforms allow developers to write small functions that are executed in isolated environments (e.g., Docker containers) in response to events (e.g., a user visiting a URL). AWS Lambda is the most well-known FaaS platform. OpenLambda (https://github.com/open-lambda/open-lambda) is an open-source FaaS platform written in Go and Python. FaaS platforms typically have a rich set of events that can trigger function invocations, including HTTP requests, cron events, file changes, streamed messages, and more. Current status: works well on a single computed; mostly used as a basis for FaaS research. Goals: (1) make it easy to run as a distributed system on a cluster of machines, (2) make it robust and complete enough for production workloads, (3) support a wide variety of event trigger types.

An intern could help write code to improve OpenLambda. OpenLambda currently only supports HTTP request events, so one possible project would involve implementing a richer set of function triggers. Alternatively, a project could focus on making OpenLambda easily deployable as a distributed system (so far, most work has been towards making an efficient execution engine that runs on a single machine).

The main mentor for this project is Tyler Caraza-Harter (teaching Faculty in the Comp Sci department). He has developed and teaches Intro to Big Data Systems (COMP SCI 544). He has mentored >20 students (via directed studies), many of whom have contributed to OpenLambda.

Intern needs: Some familiarity with systems concepts (in particular, virtual memory and multi-threaded programming), such as taught in COMP SCI 537 is essential. Experience with Go and Python programming would be ideal, but that could be learned as needed.

Project 7

Interactive Live Scripts for Science Education - Professor Carlsmith has developed roughly 90 open source interactive MATLAB Live Script tutorials/labs available at the MATLAB File Exchange introducing physics, astronomy, and engineering students to computation in science. These Live Scripts span topics from analysis of open source gravitational wave and exoplanet data to hands-on measurement of the speed of sound with a mobile phone and laptop. Many have been used in Physics 247-8. They are designed to advance computation throughout the curriculum. Internship tasks - The Live Scripts interleave code, documentation, and interactive

devices along with ‘Try this’ suggestions, ‘Challenges’, references, and hyperlinks to MathWorks documentation. Mentorship will involve a weekly virtual check-in meeting.

Intern needs: The intern will test and improve the usability of existing Live Scripts and may have an opportunity to make algorithmic improvements and to develop new scripts.

Project 8

RainyDay is an open-source Python-based software for generating large numbers of realistic extreme rainfall scenarios based on modern gridded rainfall data from weather radar, satellites, and climate models. These rainfall scenarios can then be used to examine extreme rainfall statistics for a location, or to drive flood simulations. RainyDay has been run on systems ranging from personal laptops to supercomputers. Its user base includes researchers in North America, Europe, and Asia, and it has been used by several weather/climate risk corporations and startups. Recently, the RainyDay methodology has been adopted by the Federal Emergency Management Agency and the US Army Corps of Engineers as the centerpiece for their Future of Flood Risk Data (FFRD), a $2+ billion initiative to modernize flood risk calculation and communication nationwide. Although FFRD won’t use RainyDay directly, Dr. Wright’s team has been contracted to provide R&D support for the project. This will include testing assumptions and modeling approaches, using RainyDay as the experimental “sandbox.” In addition, prior research has shown that the multi-billion dollar urban stormwater management community could benefit from converting RainyDay’s capabilities into a user-friendly web application or executable. Improvements to the usability, accessibility, and performance of RainyDay can thus offer technical and economic benefits in several important sectors. The intern would work with Dr. Wright and with Ashar Hussain, a PhD student in Dr. Wright’s team.

Interns could focus on the following activities: identify possible improvements to RainyDay (e.g., comments/errors/warnings, object-oriented structure, performance improvements); developed “containerized” (e.g. Docker/Apptainer) version of RainyDay, along with a “how to” guide for its usage to facilitate usage by non-experts, particularly in the stormwater management community; or develop a web-based app version of RainyDay.

Intern needs: Experience in software development using Python is required. Experience with refactoring of existing code is preferred, as is prior experience with geospatial data.

Project 9

The ALIVE (Advanced Baseline Imager Live Imaging of Vegetated Ecosystems) project estimates variables that are important for land surface function, including ecosystem carbon dioxide uptake and loss, evapotranspiration, land surface temperature, and the amount of solar radiation that reaches the land surface. We estimate these things in near real time using geostationary (‘weather’) satellite observations, land surface observations, and machine learning models. Our project website (https://alive-abi.github.io/alive/index.html) describes our goals and initial results, and includes our estimates of yesterday’s ecosystem carbon dioxide

uptake (the gross primary productivity, GPP) on a five-minute basis across the conterminous U.S. (CONUS) and surrounds: https://alive-abi.github.io/alive/daily-GPP.html.
We have created a cloud-based workflow, and currently save our five-minute GPP estimates as zarr libraries stored in Tigris Data S3-compatible object storage buckets using the Arraylake data lake platform to manage multidimensional data arrays and associated metadata. We are moving toward creating hourly estimates of multiple variables that build off of the machine learning methods that we have developed1–5 based on datasets that we have collected6. Because scientific and data engineering aspects of our project are moving quickly, we would like help curating code on GitHub and refining coding lessons to teach students and the public about our data products and scientific workflows.

We are interested in mentoring an intern who helps curate code for our GitHub repository (https://github.com/ALIVE-ABI/alive/tree/main); we are particularly interested in interns who are interested in education to help publish our existing Colab code examples that have been used for undergraduate courses and scientific workshops to be open for the wider community, e.g. https://colab.research.google.com/drive/1MHYQWYRDVLnrkKbqF0VXeL7c7LwfIbOY?usp=sharing

Dr. Paul Stoy will oversee mentorship and the intern will work with the ALIVE project team including Data Scientists Danielle Losos and Sophie Hoffman, as well as project PhD student Sadegh Ranjbar.

Intern needs: Basic Python programming skills and experience with GitHub would be preferred but not necessary. We can teach these skill sets as a training opportunity. There will be opportunities to contribute to peer-reviewed manuscripts for people interested in scientific publication.

Project 10

Water transport through polymeric membranes is a fundamental process critical to various applications, from seawater desalination to wastewater remediation. For over a century, the dominant theory in the field has been the solution-diffusion mechanism, where water is assumed to dissolve into the polymer matrix and diffuse across it under a constant pressure profile. However, recent advancements in atomistic molecular simulations have challenged this long-held assumption, suggesting that the traditional view may not fully capture the underlying transport phenomena (Science Advances 2023, 9, eadf8488). The simulation data indicates a linear pressure gradient across the membrane and evidence of water transport via nanoscopic porous structures that spans the polymeric membrane. These findings align more closely with a pore-flow mechanism, fundamentally different from solution-diffusion and requiring a revaluation of existing theoretical models. This project will utilize state-of-the-art molecular simulations to further investigate these observations and test their implications.

In addition to revisiting the transport dynamics across polymeric membranes, the project will explore water transport across organic solvent films, traditionally expected to adhere to solution-diffusion due to their incompressible nature. This will be a key test to validate the

applicability of molecular simulations to study non-equilibrium solvent transport phenomena. If monomers of a polymeric membrane do not polymerize, they would behave similar to a organic solvent film and transport mechanism is expected to follow the solution-diffusion mechanism. However, once fully polymerized the transport mechanism is expected to transition to a pore-flow mechanism. Therefore, a central research question arises: How does the degree of polymerization, branching, and crosslinking influence the transport mechanism in polymeric membranes? We aim to determine whether these chemical structural variations in polymers lead to a gradual transition between solution-diffusion and pore-flow mechanisms, or if the shift is abrupt and dependent on specific structural properties of a polymer.

This project will address fundamental and open questions about the mechanisms governing water transport in polymers, with the potential for improving the design principles of next-generation membrane materials. By bridging the gap between theoretical models and simulation-based evidence, our research aims to provide novel insights that could redefine how we understand and engineer polymeric membranes for enhanced performance in industrial applications. Postdoctorates Dr. Subhamoy Mahajan and Dr. Hengyu Xu in our group would guide and mentor the interns during the time on the project.

Intern needs: BASH scripting, Basic knowledge of LAMMPS molecular dynamics package

Project 11

The OSPO seeks a student to assist with communications and outreach efforts in our recently established office. Our vision is to be an accessible resource for developers, learners, and researchers alike who want to code in the open. As the OSPO is committed to removing barriers in the open source community, we encourage students from historically underrepresented groups to apply. This intern will report to the OSPO Program Manager and work closely with the OSPO Outreach Specialist.

Internship duties will include interviewing open source practitioners for our Faces of Open Source spotlight, developing OSPO content for the Data Science Updates newsletter, crafting OSPO messaging for various audiences, assisting with publicizing events and initiatives, updating the OSPO website with news items, and additional projects and duties as assigned.

Intern needs: Enthusiasm for communicating science and technology developments to broad audiences, Experience in writing and communications courses or roles, Familiarity with digital media used in communications, Experience with Microsoft Office

Proposals to Contribute to Major Open Source Projects

Students may submit proposals to contribute to a well-established open source project. The proposal could describe contributions to the code, documentation, or other needs that have been identified by the open source project. Since these projects will not necessarily have strong mentorship from people based at UW–Madison, we only recommend this option for students that are familiar with collaborating via GitHub / GitLab. The targeted project should have well

established contributor guidelines and specific set of open issues to be addressed. Ideally, the applicant can identify a person from the project that agrees to some level of advising and mentorship.

Students proposing a self-directed project must include the following in their cover letter:

- Point to the project and their contributor guidelines
- Point to specific issues the project will address.
- Demonstration of experience with contributions to open source or collaborative software development on GitHub / GitLab.
- Identify a person from the open source project (or a local advisor) that will serve as a mentor.