**User Training**

**Fall 2018 Part I Registrations Now Open!**

You must have a Harvard University ID (HUID) to be able to register for a class. This is required to access the Harvard Training Portal.

Those members of the HMS community who do not have HUIDs - such as employees at affiliate hospitals, or collaborators from other institutions - may self-register for one as a "Person of Interest" with their faculty member's sponsorship. The form may take several days to process, so we encourage non-HUID users to fill out this form ASAP in preparation for upcoming training: [http://iam.harvard.edu/files/iam/files/authorized-identity-request-form.pdf](http://iam.harvard.edu/files/iam/files/authorized-identity-request-form.pdf) (please read through the form for submission details)

**Current Schedule**

<table>
<thead>
<tr>
<th>Class</th>
<th>Date</th>
<th>Time</th>
<th>Room</th>
<th>Page</th>
<th>Registration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intro to O2</td>
<td>08/29/2018</td>
<td>3-5p</td>
<td>Countway 403</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to Matlab</td>
<td>09/12/2018</td>
<td>3-5p</td>
<td>TMEC 328</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to Perl</td>
<td>09/19/2018</td>
<td>3-5p</td>
<td>TMEC 448</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to Python</td>
<td>10/10/2018</td>
<td>3-5p</td>
<td>TMEC 328</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>MathWorks Image Processing and Computer Vision</td>
<td>10/23/2018</td>
<td>1-3p</td>
<td>Armenise 125D</td>
<td></td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to R/Bioconductor</td>
<td>10/24/2018</td>
<td>3-5p</td>
<td>TMEC 446</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to Parallel Computing</td>
<td>10/31/2018</td>
<td>3-5p</td>
<td>TMEC 446</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>MathWorks Demystifying Deep Learning (including GPU Coder)</td>
<td>11/13/2018</td>
<td>1-3p</td>
<td>Armenise 125D</td>
<td></td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to O2</td>
<td>11/14/2018</td>
<td>3-5p</td>
<td>TMEC 328</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to Git/Github</td>
<td>12/05/2018</td>
<td>3-5p</td>
<td>TMEC 328</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
<tr>
<td>Intro to O2</td>
<td>12/19/2018</td>
<td>3-5p</td>
<td>TMEC 448</td>
<td>User Training Github</td>
<td>Register Here</td>
</tr>
</tbody>
</table>

**Classes offered:**

**Intro to O2**

O2 for New Users addresses the needs of users who have very little linux experience, and are just getting started with HPC. More time will be devoted to covering linux basics, and the concepts of schedulers and jobs, and data management best practices. The lecture portion of this class is one hour, the second hour will be spent clinic-style with HMS RC staff to address workflow-specific questions and help convert commands to O2 SLURM syntax.

**Intro to Python**

Python is a popular scripting language for scientific computing and available across all computer platforms. The course will introduce you to some of the basics of the Python language as well as some of the nuances involved with its use specific to the O2 environment. The goal is to provide users with a foundational level of familiarity. Topics covered include basic data types and declaration, flow control (if/else), loops, a brief introduction to constructing a script, and a briefer introduction to modules. The course will be taught on O2, but general concepts are easily translatable to desktop and local installations.

**Intro to R/Bioconductor**

Intro to using R and Bioconductor. R is a powerful, open-source, highly adaptable statistical language useful for crunching numbers to datasets
like those produced by next-gen sequencing. This class covers R basics and learning to think like/understand R. Users will learn how to set up personal R libraries on O2, and use O2 R for its high memory allocations and parallelization. Topics include how to install packages, learn about variables, data types, data manipulation, flow control, and functions, perform simple statistical tests, and create a variety of plots. Laptops are encouraged.

Class Files Here

**Intro to MATLAB**

Matlab has become the “language of science” in the past few decades. It is simple to use, yet powerful enough to be productive on large computing infrastructures. If you need: 1) Fast prototyping of research ideas; or 2) avoid spending too much time in coding instead of doing real science by taking advantage of Matlab’s built-in functions; 3) User friendly graphical interface and educational documentation; 4) Simplicity of code; 5) Easy access to GPU computing power; 6) Easy plotting and presentation of data; you will find this introduction course useful. This course will introduce the basics of the MATLAB coding language with O2-scalability and data presentation.

**Intro to Parallel Computing**

This is a short introduction to Parallel Computing that will include an overview of the basic concepts of parallel programming: from running your job in an embarrassingly parallel way to writing simple shared and distributed memory parallelization codes in different languages. The seminar will cover several examples of actual parallel codes however it will not have any “hands on” components. A basic programming experience (of any language, no parallelization) is preferred in order to better follow the topics presented during the seminar.

**Intro to Perl**

Perl is an open source programming language that's flexible, available for nearly all platforms, and is easy to learn. It is well suited for data munging, and processing biological data. Topics covered include variables (scalars, arrays and hashes), numerical and string functions, loops and conditions, regular expressions, and reading and writing files. A brief introduction will be given for subroutines and BioPerl. The class will be focused on running Perl on the O2 cluster, though the fundamentals are applicable for using Perl installations elsewhere.

**Intro to Git and GitHub**

This course introduces Git and GitHub and covers topics including: Getting Started with Git for version control, Using GitHub Desktop effectively, Collaborating with others on GitHub, and Utilizing GitHub Flow for better workflow. No previous exposure is assumed. We hope attendees will leave the class with the knowledge and tools necessary to start integrating Git into their workflows and excited to begin collaborating on GitHub.

**R/Biostatistics**

R/Biostatistics is a multi-part course covering the basics of RNA-seq analysis with R. This biostatistics course covers standard supervised approaches and functional enrichment analyses of a breast cancer RNA-seq dataset. Topics include edgeR for differential analysis, GOSeq for functional enrichment analyses, and KEGG pathway analysis. High-throughput data visualization techniques are emphasized. Each class includes a lecture and R practicum, and registration is for all courses. Laptops are encouraged.

**MathWorks Image Processing and Computer Vision with MATLAB**

This seminar will be particularly valuable for anyone interested in using MATLAB to process, visualize, and quantify imagery. Rather than focus on extracting information from a few homogeneous images, we will introduce a typical real-world challenge, and discuss approaches to managing and exploring collections of widely heterogeneous images. We will also describe approaches to implementing deep learning networks in MATLAB, and will compare and contrast those approaches with more computer vision and machine learning techniques.

In this presentation, we will:

* Explore and manage a range of real-world image sets
* Solve challenging image processing problems with user interfaces
* Classify images by content using machine learning techniques
* Detect, recognize, and track objects and faces in images

**MathWorks Demystifying deep learning: A practical approach in MATLAB**

Are you new to deep learning and want to learn how to use it in your work? Deep learning can achieve state-of-the-art accuracy in many humanlike tasks such as naming objects in a scene or recognizing optimal paths in an environment. The main tasks are to assemble large data sets, create a neural network, to train, visualize, and evaluate different models, using specialized hardware - often requiring unique programming knowledge. These tasks are frequently even more challenging because of the complex theory behind them. In this seminar, we'll demonstrate
new MATLAB features that simplify these tasks and eliminate the low-level programming. In doing so, we’ll decipher practical knowledge of the
domain of deep learning. We’ll build and train neural networks that recognize handwriting, classify food in a scene, and figure out the drivable
area in a city environment.

Along the way, you’ll see MATLAB features that make it easy to:

• Manage extremely large sets of images
• Visualize networks and gain insight into the black box nature of deep networks
• Perform classification and pixel-level semantic segmentation on images
• Import training data sets from networks such as GoogLeNet and ResNet
• Import and use pre-trained models from TensorFlow and Caffe
• Speed up network training with parallel computing on a cluster
• Automate manual effort required to label ground truth
• Automatically convert a model to CUDA to run on GPUs