Module 0: Getting Started

Welcome to the *Practicum AI: Computing for AI* Course! This course is the second in the *Practicum AI* series. This course can also be taken on its own to familiarize yourself with some of the important tools used in computational science applications.

In this course, you will learn about some of the tools recommended for building, testing, tweaking, and deploying AI models. You will learn about **Jupyter Notebooks, Git, GitHub,** and high-performance computing (**HPC**) environments. These are the key technologies that have become the industry standards for hands-on, applied AI research and development.

In this course we will cover

Module 1: The tools for AI: This module provides a quick overview of the tools we will learn in this course, including why each tool is important for the overall goal of learning to do hands-on, applied AI.

Module 2: Jupyter Notebooks: We will get you started using the widely used notebook technology that powers much of the exploratory analyses that go into *doing* AI. We will get you up and running on one of SCINet's HPC systems, Atlas.

Module 3: git and GitHub.com: This module will introduce you to version control and using git, the industry-standard tool for managing versions of code. The ability to host code online with a site like GitHub.com has fueled the rapid development of AI tools. By learning about these tools, you will be well-equipped to continue developing your own AI applications.

Course Objectives:

By the end of this course:

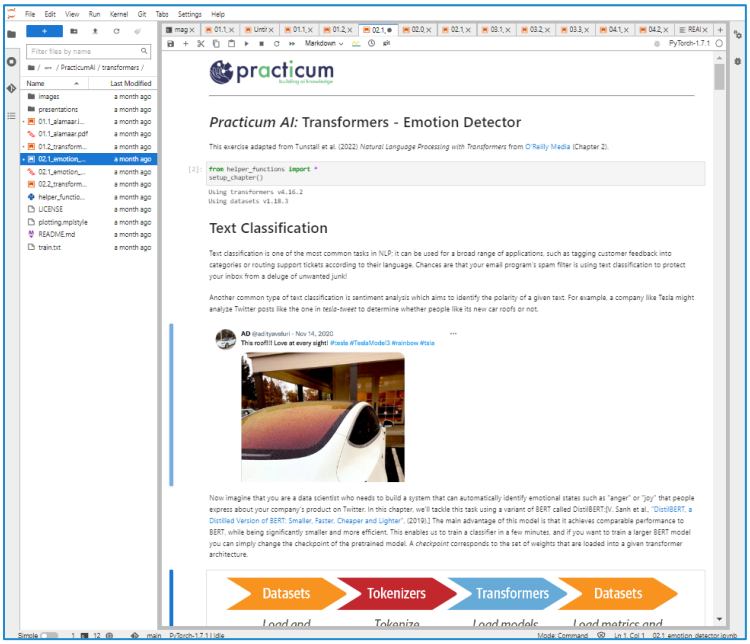
1. Students will be able to describe the key technologies that are needed for hands-on AI applications.
2. Students will be familiar with Jupyter notebooks as the predominant tool for AI data analysis and model development. Students will be able to:
    a. Select the appropriate compute environment and start a Jupyter Notebook.
    b. Explain what Jupyter Notebooks are, and why they are used in data science applications.
    c. Understand what Markdown is and what makes it useful.
    d. Run code blocks and embed images and graphs in Notebooks.
3. Students will recognize the importance of version control in ensuring open, reproducible, and sharable AI code. Students will be able to:
    a. Navigate GitHub, including cloning and creating repositories and adding them to a Jupyter environment.
    b. Use a git workflow to edit a file, add the file, commit the changes, and push changes to a remote repository.

c. Navigate the GitHub discussion board and issue system for getting help and reporting problems.
d. Explain why version control is an important part of AI development.
    i. Understand some ethical considerations concerning version control in software development.
    ii. Understand some ethical considerations concerning code transparency and reproducibility.
e. Understand what SSH keys are, and why they are important.

4. Students will be familiar with the computational demands of AI applications and the need for HPC environments with GPUs for AI model development and deployment.
    a. Understand some ethical considerations concerning HPC use (or lack thereof).

Module 1: The tools for AI

Jupyter Notebooks



AI relies on computer code, and **Jupyter Notebooks** are the standard tool to develop and share AI and other data analysis code. The code is usually in the Python programming language, which we'll cover in the next course, *Practicum AI: Python for AI*! Jupyter Notebooks are interactive, so you can see the results of your code as you change it, which facilitates the iterative processes involved in cleaning data, testing different models, and adjusting hyperparameters (model settings). Notebooks also allow you to add nicely formatted text and images to provide explanations, embed helpful graphics, and overall, enhance the user experience. Notebooks are also portable, so you can use them on your laptop or on a remote server. For these reasons and more, Jupyter Notebooks are a great tool not only for developing AI, but also for teaching you about AI, and that is why we will use Jupyter Notebooks extensively throughout the *Practicum AI* series of courses.

Git and GitHub.com

As noted above, code is at the foundation of AI. Being able to transparently and easily track the development of code, share code with others, and foster reproducibility are important features offered by using a type of software known as version control systems. **Git** is a version control system that allows you to track changes to files over time. This makes it easy to collaborate with others on projects, revert to previous versions of your code, find bugs, test new ideas, and share reproducible results.

GitHub is a website that hosts git repositories (or collections of files). It is a great tool for AI development because it allows you to store, share, and deploy code. GitHub also has many features that support development by allowing you to track your progress on team projects, track and resolve bugs, host web pages, and more. **Note:** Some GitHub features, such as web page hosting, are subject to additional restrictions for official USDA-ARS use.

Leveraging similar technologies, model and dataset repositories allow seamless sharing and versioning of AI models and datasets. By sharing AI models and datasets, the time, effort, compute power, and energy consumed in training these models and developing the datasets can be leveraged for future research. While version control, git, and model repositories may seem like advanced tools, they are such critical parts of AI development that we feel that they are important tools for every AI practitioner to learn. By starting with these tools from the beginning, you will be well-positioned to continue using industry-standard tools as you continue your journey.

Computing Environments

High-Performance Computing (HPC) environments are essential for training large AI models. As we showed in the first course, rapid advances in AI in the 2010s and beyond were fueled by recognition that the graphics processing units (GPUs) originally designed for gaming were ideal for the highly parallel computations used in training neural networks. Subsequent advances in GPUs specifically designed for AI applications have further increased capabilities. While these advances have driven amazing new applications of AI, they also mean that for most applications, specialized, and often expensive, computer hardware is needed for model training and deployment. Ultimately, this means what you can do on a standard laptop or desktop computer is limited when it comes to AI. For this reason, the *Practicum AI* courses are not intended to be run on your own computer.

Access to compute resources with advanced GPUs is one barrier that limits equitable access to AI technologies. Some companies, like Google, provide limited GPU access for free, but due to the hardware costs and power consumption, free tiers are often limited, and costs can quickly add up when doing AI model development and deployment.

HPC systems are generally composed of hundreds, or thousands, of servers all networked together to operate as a single system. Access models vary, but many universities, government agencies, and companies have their own HPC system for their needs. Amazon, Microsoft, Google, and others rent access to portions of their systems, generally referred to as cloud computing. The key commonality among these systems is that they provide access to scalable compute power that far exceeds what could be done on standard personal computers.

Training models on an HPC system can bring tasks that can take weeks (or longer!) down to just minutes (or shorter!!!). Training AI models is an iterative, experimental process requiring adjusting model parameters and trying different things. Keeping training times reasonable enables more experimentation.

The USDA-ARS version of the Practicum AI series is designed to run on Atlas, a SCINet HPC system in Starkville, MS hosted by Mississippi State University (MSU). You will likely see references to HiPerGator, the University of Florida's supercomputer which uses much of the same hardware and software as is used on Atlas. Other Practicum AI content references Google Colab but remember that is not authorized for USDA work.

By the end of this module, you will be familiar with these tools, giving you a solid foundation with which to build and deploy AI models of your own!