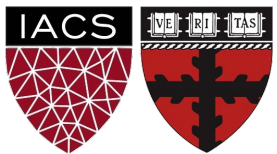
Operationalizing Models

From Notebook to App
Deep Learning Operations (DLOps)

CS109B Data Science 2

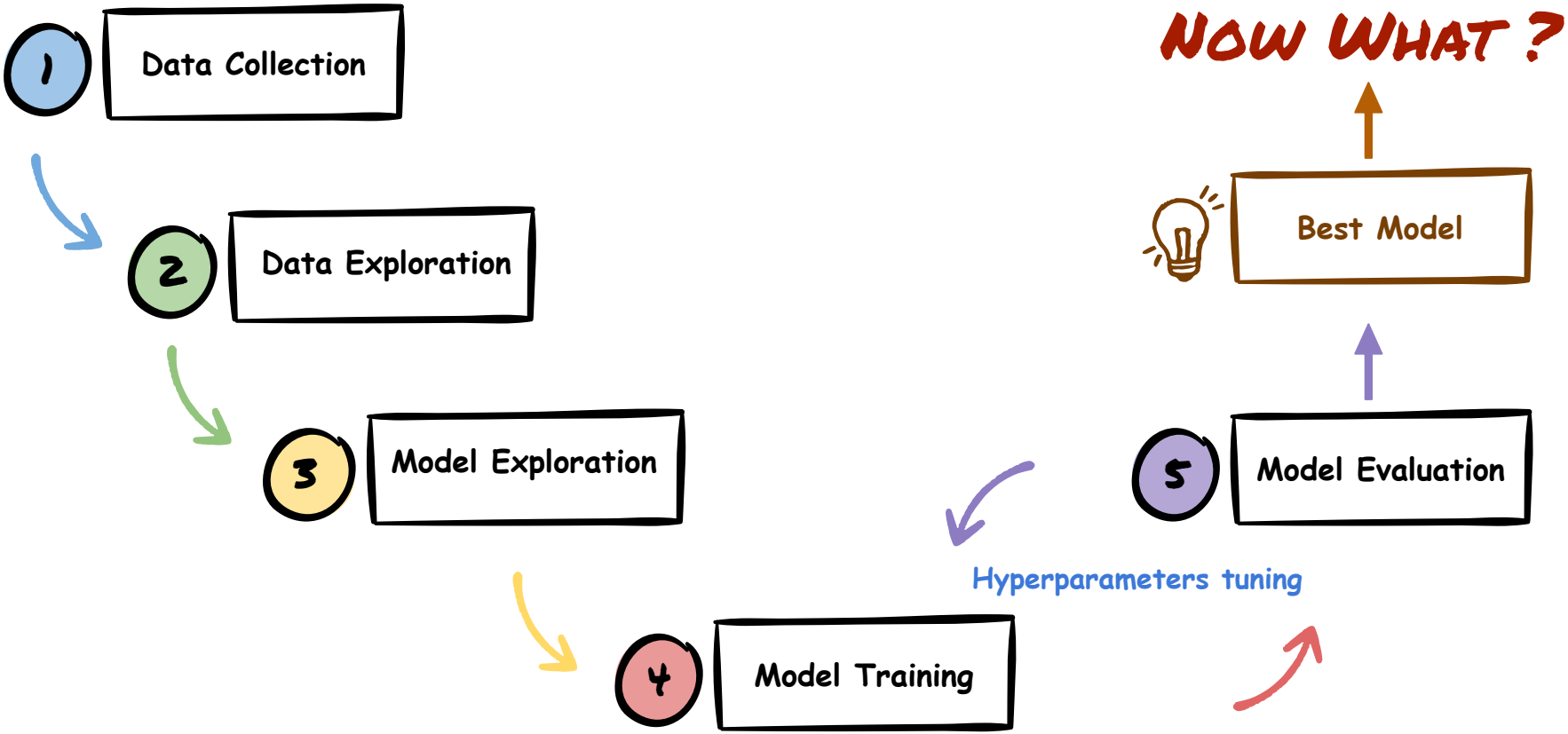
Shivas Jayaram, Pavlos Protopapas



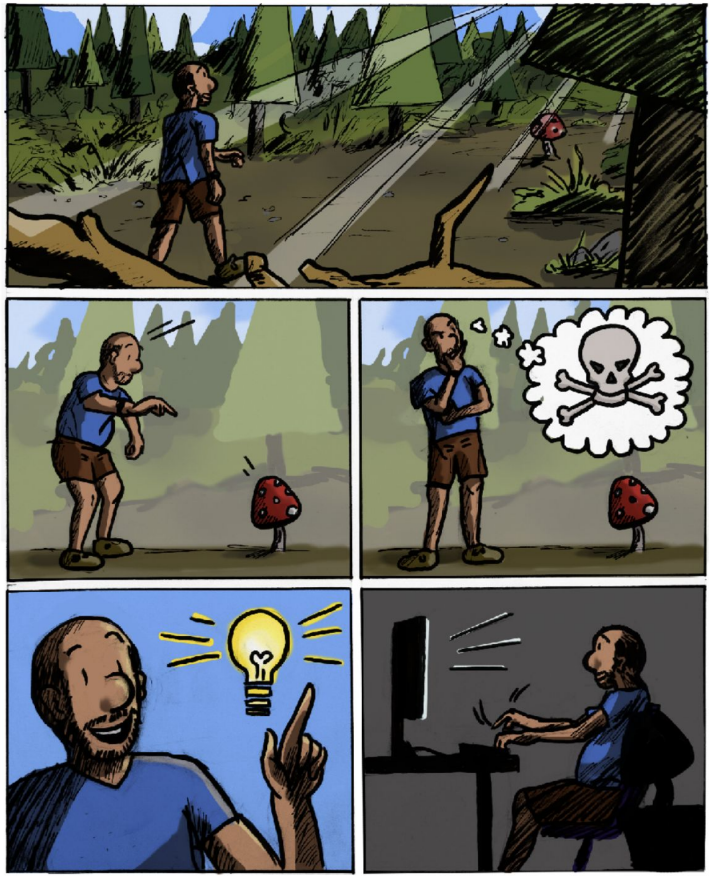
Overview

- Summary of Deep Learning Flow
- Building a “Mushroom Finder” App
- Challenges in operationalizing models
- Intro to Deep Learning Operations (DLOps)

Deep Learning Flow



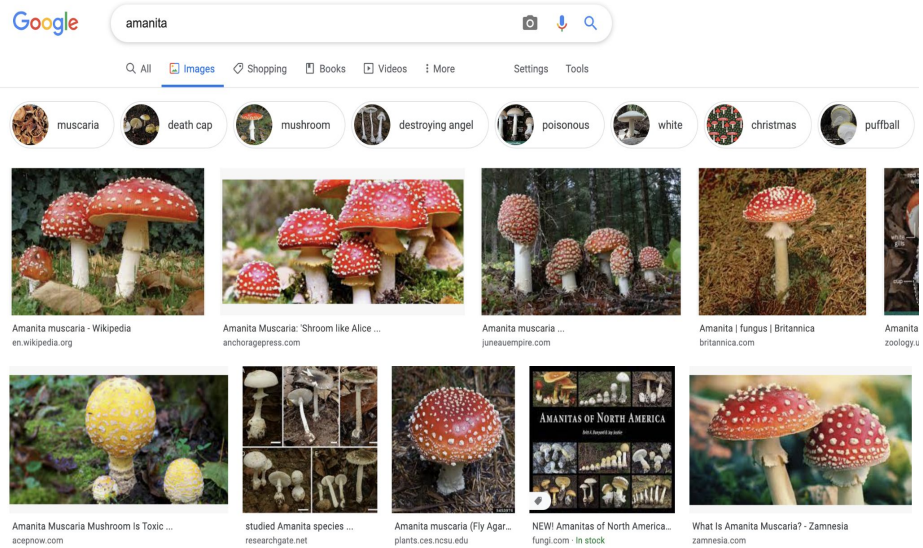
Use Case: Finding Mushrooms 🍄



Credit: Nikolas Protopapas

Data Collection

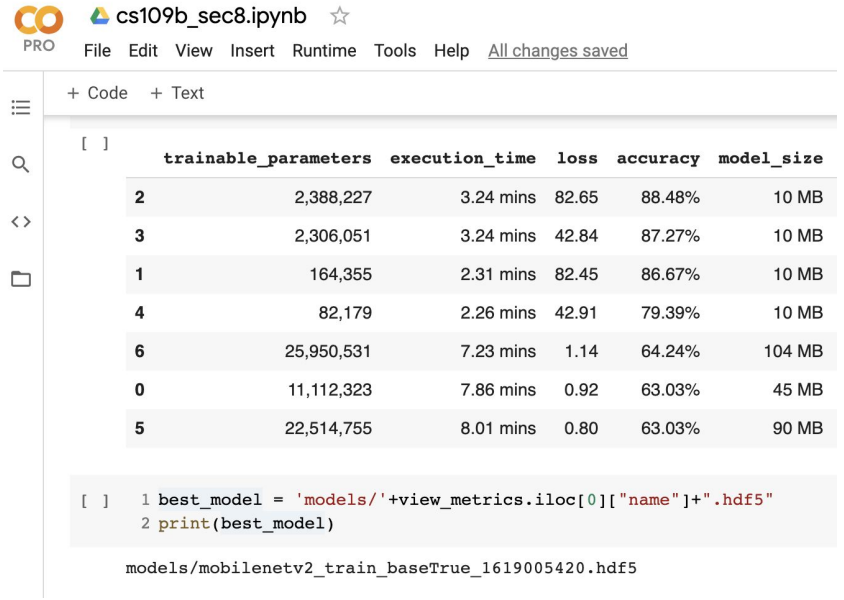
- Collect images from Google
- For our example we'll download images for mushrooms oyster, crimini, **amanita (Poisonous)**
- **Code walkthrough...**



Python Script

Model Exploration / Training / Evaluation

- Identify our problem task
- Try various model architectures
- Transfer Learning
- Hyperparameters tuning
- **Code walkthrough...**



Colab

Build 🍄 Mushroom Finder App

Build 🍄 Mushroom Finder App

- We want to build an app to take a photo of a mushroom and it helps us identify the type of mushroom
- How do we build the app?

Expose our Model

FastAPI is a modern, fast (high-performance), web framework for building APIs with Python. https://fastapi.tiangolo.com/

A REST API is way of serving model prediction calls into a HTTP request

Expose our Model

https://mushroom.tunnelto.dev/api/predict_file

```
{
    "prediction_label":"amanita",
    "accuracy":93.54
}
```

https://mushroom.tunnelto.dev/api/predict_file

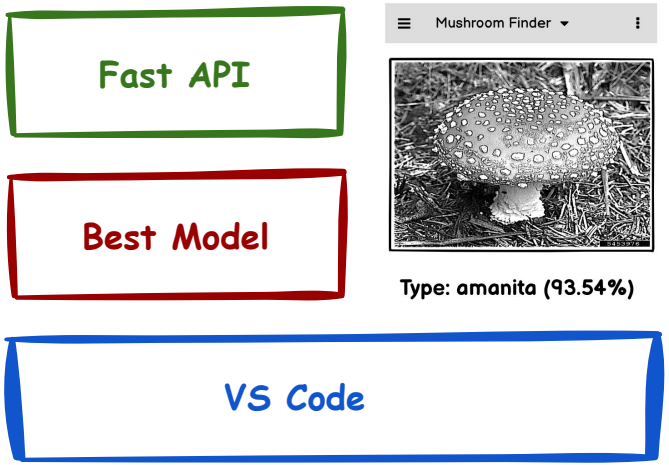
```
{
    "prediction_label":"crimini",
    "accuracy":80.72
}
```

https://mushroom.tunnelto.dev/api/predict_file

```
{
    "prediction_label":"oyster",
    "accuracy":98.25
}
```

Build 🍄 Mushroom Finder App

- Save our best model
- Build an API (application programming interface) using FastAPI package
- Build a frontend app using HTML & javascript
- **Code & Demo walkthrough...**



Deploy Mushroom Finder App

Fast API

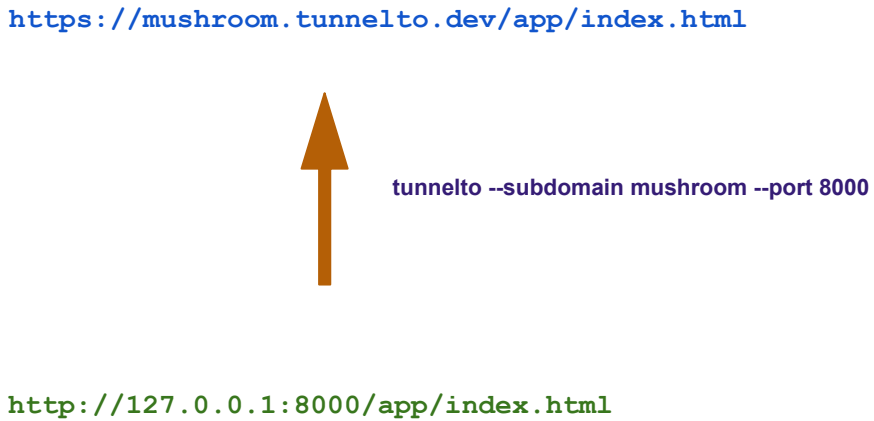
Best Model

VS Code

http://127.0.0.1:8000/app/index.html

Deploy 🍄 Mushroom Finder App

- Ideally we will want to deploy our app to a server
- But for simplicity we will expose our app from our local machine to the public internet using **tunnelto.dev**

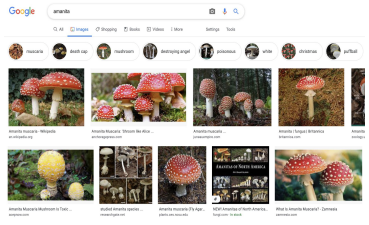


tunnelto.dev is utility to expose your local web server to the internet with a public URL. https://tunnelto.dev/

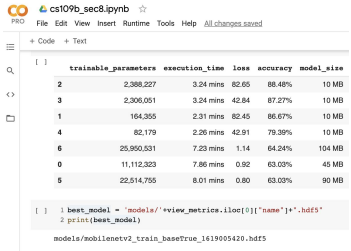
Deploy 🍄 Mushroom Finder App

- You can go to https://mushroom.tunnelto.dev/app/index.html
- Try out the 🍄 app!

Putting it all together



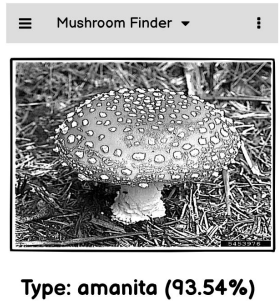
Python Script



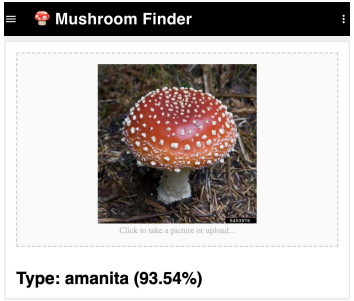
Colab

Fast API

Best Model



VS Code



Putting it all together

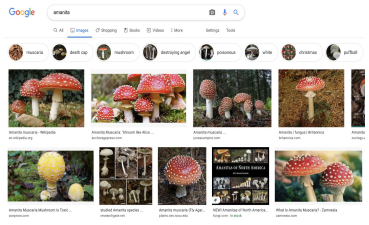
Data Collection

Data Exploration

Model Exploration

Model Training

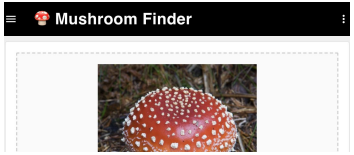
Model Evaluation



Python Script

Colab

OPERATIONALIZING MODEL !



Fast API

Best Model

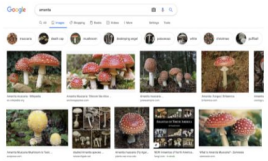
Type: amanita (93.54%)

VS Code

Challenges

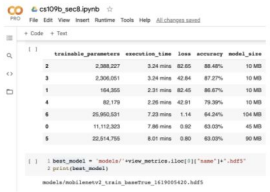
Development

Deployment



Python Script

Python: pipenv
Chromium: Mac install
OS: Mac



Colab

Python: Colab provided env
OS: Linux

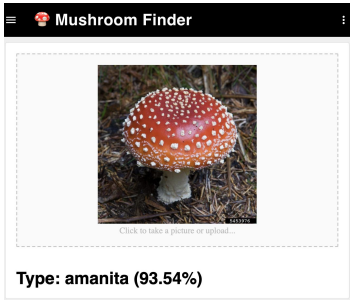
Fast API

Best Model

Type: amanita (93.54%)

VS Code

Python: pipenv
OS: Mac



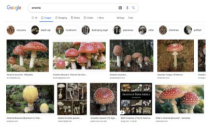
Python: pipenv
OS: Linux

Server

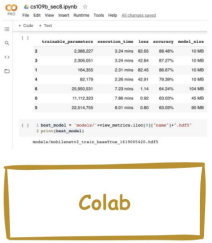
One developer
Using a Macbook

Challenges - Multiple Developers

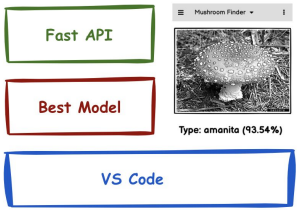
Development



Python Script



Colab

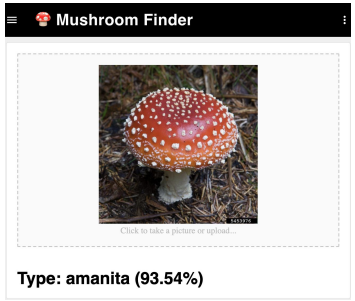


Python: pipenv
Chromium: Mac install, Windows install
OS: Mac, Windows

Python: Colab provided env
OS: Linux

Python: pipenv
OS: Mac, Windows

Deployment



Python: pipenv
OS: Linux

Server

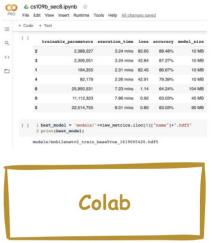
Multiple developers, Using Mac and Windows OS

Challenges - Multiple Developers + Automation

Development

Python Script

Python: pipenv
Chromium: Mac install, Windows install
OS: Mac, Windows



Colab

Python: Colab provided env
OS: Linux

Fast API

Best Model

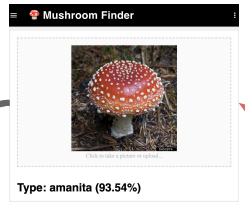
Type: amanita (93.54%)

VS Code

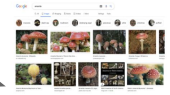
Python: pipenv
OS: Mac, Windows

Multiple developers, Using Mac and Windows OS

Deployment



Model Training



Python Script

Python: pipenv
OS: Linux

Server

Challenges - Multiple Developers + Automation

- OS specific installations required
- How to collaborate code?
- How to share Datasets & Models?
- Need for multi GPUs or training for more than 12 hours
- Automate data collection / model training
- New team member onboarding
- “It works on my machine” ¯_(ツ)_/¯

DLOps

- **Development Operations (DevOps):**

  DevOps is a practice that brings together software development (Dev) and operations (Ops) to streamline this process for better productivity and shorten development life cycle

- **Deep Learning Operations (DLOps):**

  DLOps is a practice that brings together deep learning model development, application development, and operations together to streamline the interaction between the three and simplify the deep learning life cycle

** These concepts will be taught in more detail in the fall in **AC295 - Advanced Practical Data Science**

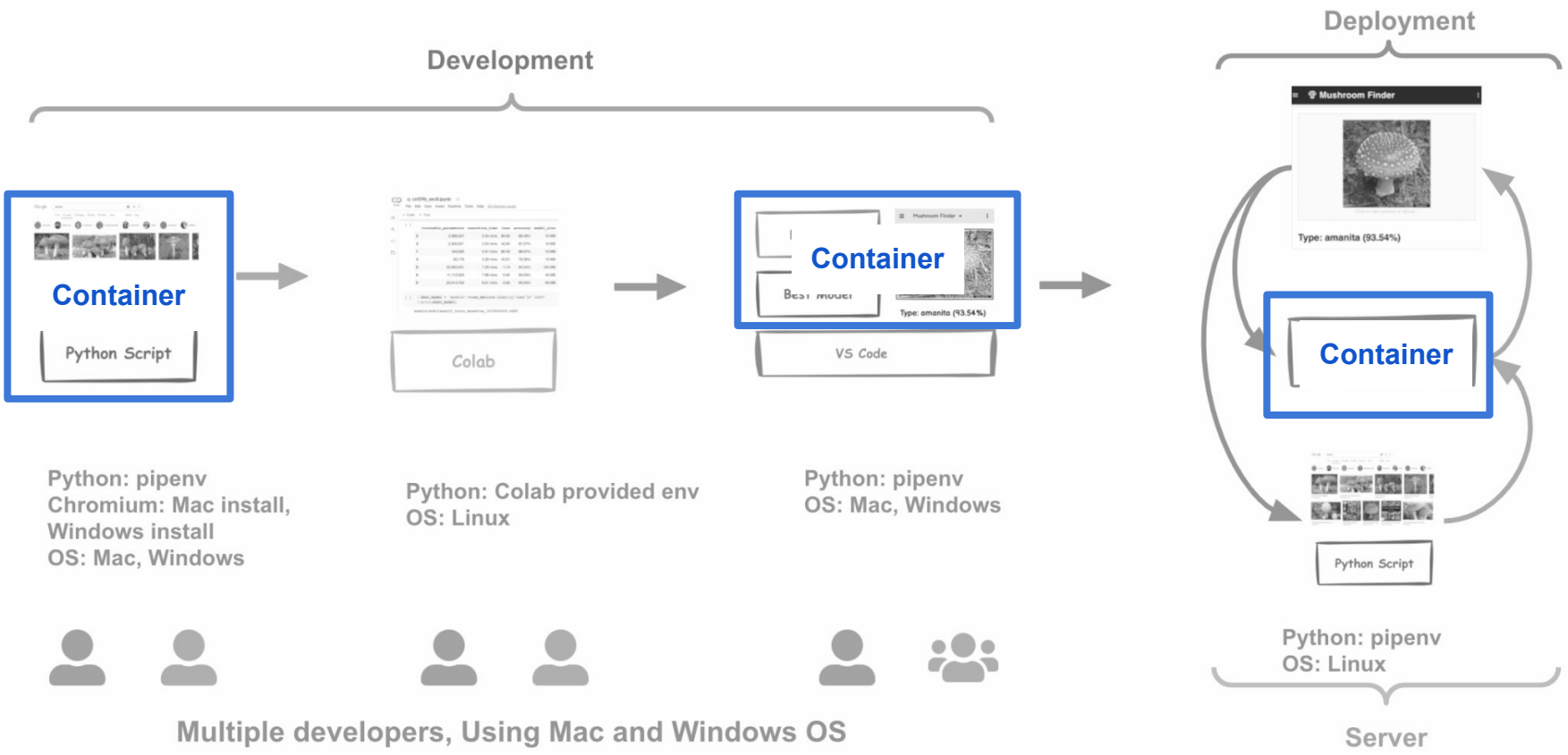
DLOps

- **Deep Learning**
  - Data collection & exploration
  - Model exploration & selection
  - Model training & evaluation
  - Model distillation & quantization
- **Development**
  - APIs / Model serving
  - Web & mobile apps
  - Edge device apps
  - Services for automation

- **Operations**
  - Provisioning and managing deployment servers
  - Provisioning and managing on-demand GPU servers
  - Maintain 100% uptime of app / apis
  - CI/CD: Continuous Integration / Continuous Deployment
  - Continuous data collection / model training
  - Model monitoring

** These concepts will be taught in more detail in the fall in **AC295 - Advanced Practical Data Science**

DLOps - Containers

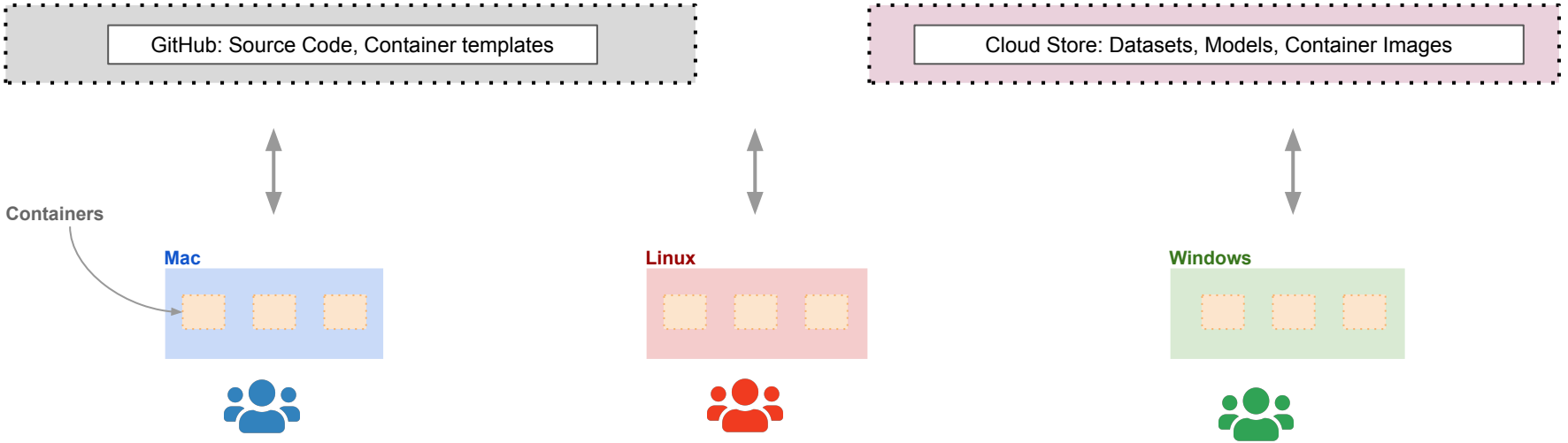


DLOps - Containers

- Containers are fully packaged software with all dependencies included
- Containers can be used for development, training, and deployment
- Containers are extremely portable and lightweight
- Development teams can easily share containers
- **Docker** is a tool designed to make it easier to create, deploy, and run applications by using containers

** These concepts will be taught in more detail in the fall in **AC295 - Advanced Practical Data Science**

DLOps - Common Store

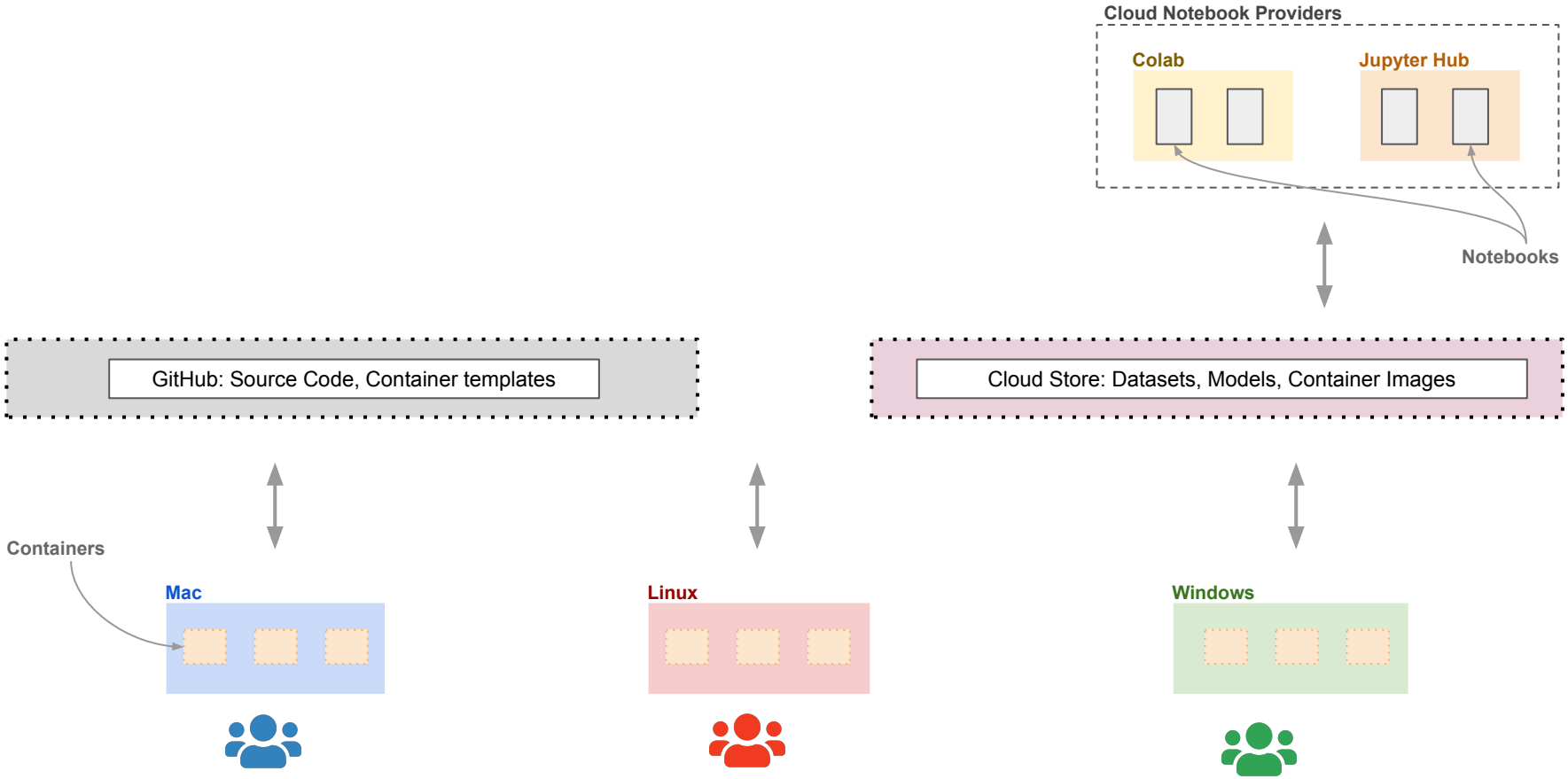


DLOps - Common Store

- **GitHub**
  - GitHub is a code hosting platform for version control and collaboration
- **Cloud Store**
  - Any file hosting platform to share datasets & models. E.g: Google Cloud Store, Amazon S3 buckets
  - A Container registry to manage containers. E.g: Docker Hub, Google Container Registry

** These concepts will be taught in more detail in the fall in **AC295 - Advanced Practical Data Science**

DLOps - Notebooks for Prototyping

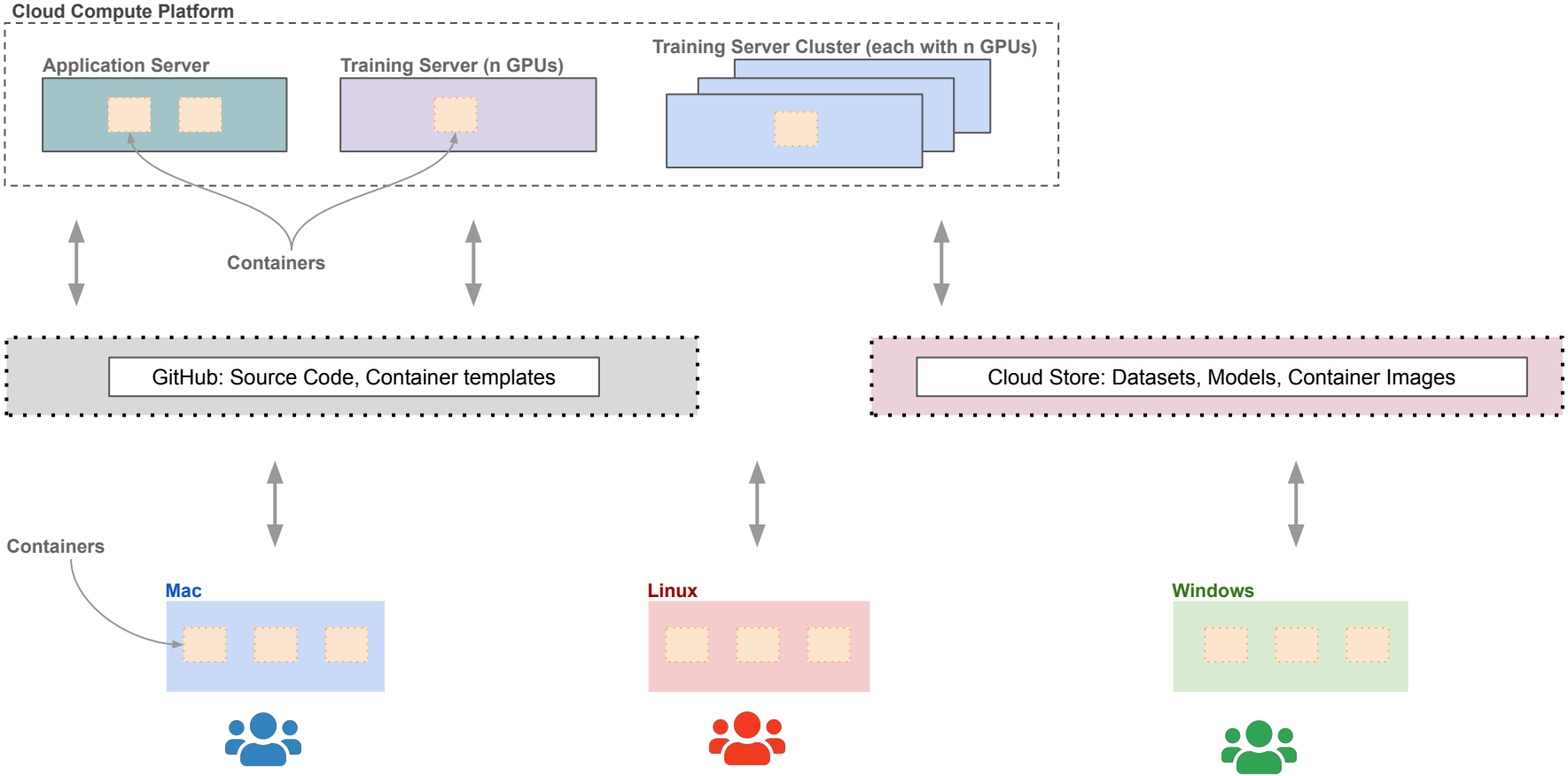


DLOps - Notebooks for Prototyping

- **Colab**
  - Colab is a hosted notebook service offered by Google
  - Access to free GPU
  - Pro version gives you better GPUs and more training time
- **Jupyter Hub**
  - Hosted notebooks with access to GPU
  - You would need to keep track of shutting down your instance as cost can add up
- **Integration between Cloud Store and Notebooks**
  - Sharing datasets & models

** These concepts will be taught in more detail in the fall in **AC295 - Advanced Practical Data Science**

DLOps - Cloud Compute Platform

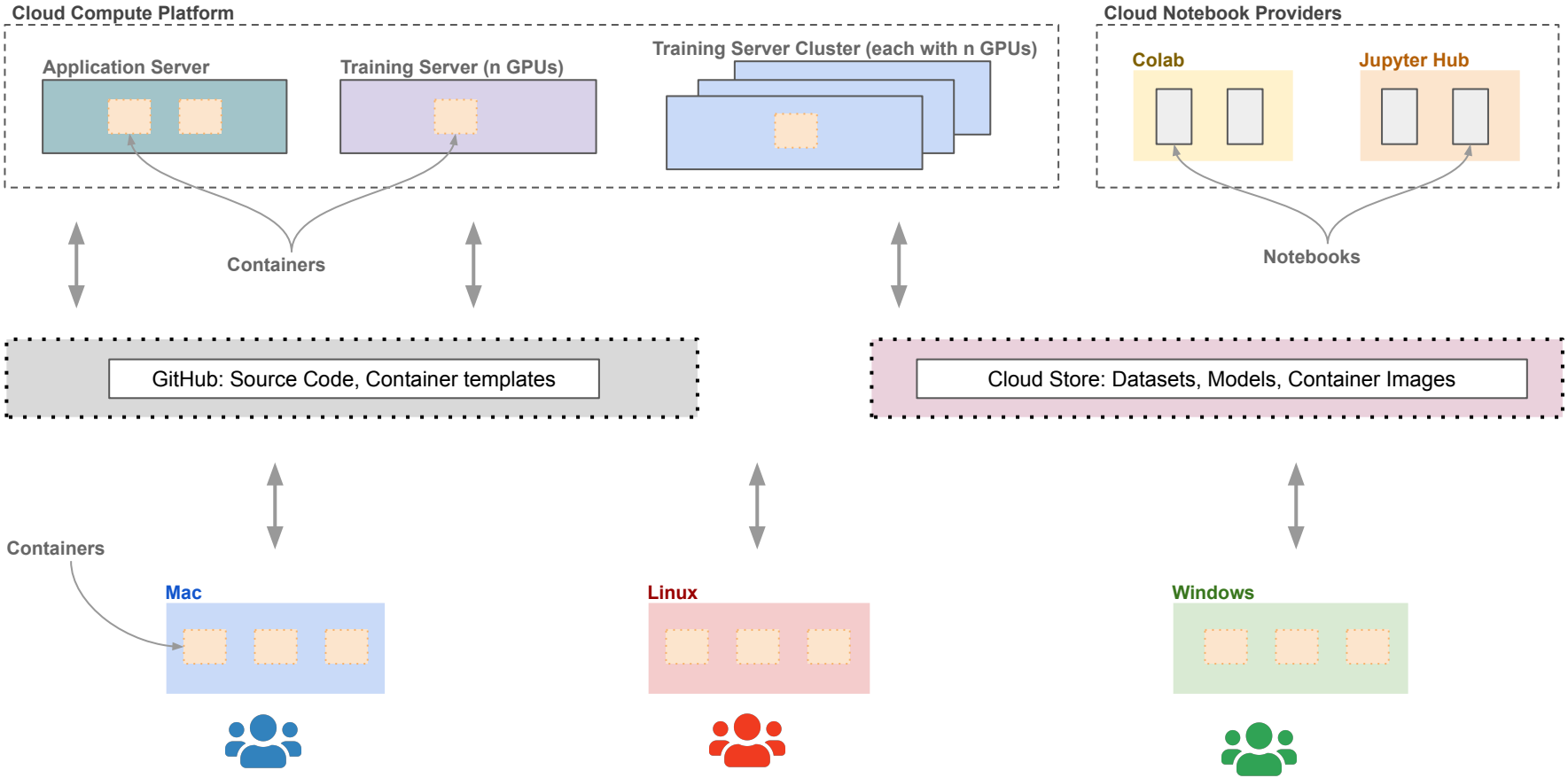


DLOps - Cloud Compute Platform

- **Deployment Server**
  - Server that host app, APIs, databases etc
  - These machines need not have GPU since most models can be deployed for inference with CPU
  - Cheaper CPU servers are available from all could providers
- **Training Server**
  - Servers with GPUs for model training
  - GPU servers are expensive
  - On demand only
- **Training Server Cluster**
  - Server clusters for multi node / multi GPU training
  - These are for very large model training
  - On demand only

** These concepts will be taught in more detail in the fall in **AC295 - Advanced Practical Data Science**

DLOps - The Complete Landscape



Revist Challenges

- ✔OS specific installations required
- ✔How to collaborate code?
- ✔How to share Datasets & Models
- ✔Need for multi GPUs or training for more than 12 hours
- ✔Automate data collection / model training
- ✔New team member onboarding
- ✔“It works on my machine” ¯_(ツ)_/¯

DLOps - Tech Stack

- **Deep Learning**
  - TensorFlow, PyTorch,Apache MXNet, MS Cognitive Toolkit, Tensorflow Lite, TFJS
  - Colab, JupyterHub
  - Amazon Sagemaker, Google AI Platform
  - KubeFlow
  - Dask
- **Development**
  - FastAPI, Tensorflow Serving, TorchServe
  - React, Angular, Vue
  - Xcode, SwiftUI, Android Studio
  - VS Code
- **Operations**
  - GCP (Google Cloud Platform)
  - GitHub
  - Docker
  - Kubernetes
  - Ansible (Infrastructure deployment)
  - Jenkins (CI/CD)
  - Nginx (Web server)

** Some of these will be taught in more detail in the fall in **AC295 - Advanced Practical Data Science**

What Next?

If these topics interest you definitely check out
AC295 - Advanced Practical Data Science in the fall

Thank You

Questions?