Introduction to AI Ops

Linux Foundation

- ❑   Why Linux? Linux types? How to access Linux env in different system
- ❑   Installation of virtual box, WSL, sandbox for windows user
- ❑   SSH and SSH tools
- ❑   Putty
- ❑   Filezilla
- ❑   WinScp
- ❑   Course introduction
- ❑   Basic Linusx commands
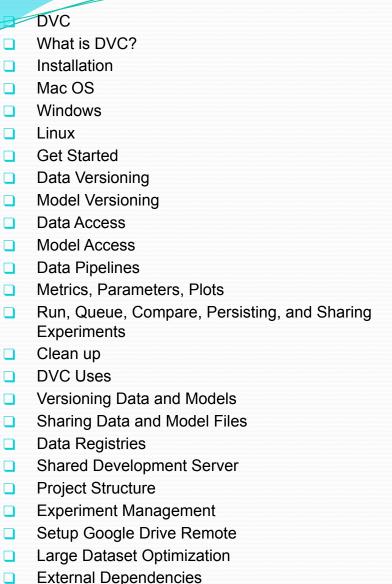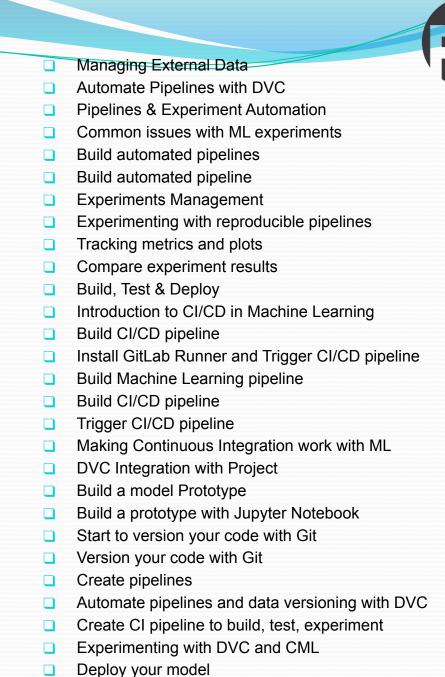- ❑   cronJobs,

GIT Foundation

- What? Why? When? Type? Vendor? Pricing? Industry wise uses of GIT
- Creation of Github/Gitlab/bitbucket account
- Local GitHub UI installation, setup with VSCode and Pycharm
- Local and Remote Repositories installation and configuration
- GIT Repository initialization
- command: git log
- Git Branches
- What is branching in Git and why we need it?
- Master/main branch and user-defined branch
- Checkout and pushing to a branch
- Merging of branches
- Project control and management
- In Remote Repositories

- Initialization of Remote Repositories
- Pushing code to the remote repositories
- Cloning of the remote repositories to local
- PR (Pull Requests)
- Fetch and Pull
- Handling conflict on merging branch
- Forking of repository
- Rebasing
- Resetting and Reverting
- Stashing

Data Version Control (DVC)

- DVC
- What is DVC?
- Installation
- Mac OS
- Windows
- Linux
- Get Started
- Data Versioning
- Model Versioning
- Data Access
- Model Access
- Data Pipelines
- Metrics, Parameters, Plots
- Run, Queue, Compare, Persisting, and Sharing Experiments
- Clean up
- DVC Uses
- Versioning Data and Models
- Sharing Data and Model Files
- Data Registries
- Shared Development Server
- Project Structure
- Experiment Management
- Setup Google Drive Remote
- Large Dataset Optimization
- External Dependencies

- Managing External Data
- Automate Pipelines with DVC
- Pipelines & Experiment Automation
- Common issues with ML experiments
- Build automated pipelines
- Build automated pipeline
- Experiments Management
- Experimenting with reproducible pipelines
- Tracking metrics and plots
- Compare experiment results
- Build, Test & Deploy
- Introduction to CI/CD in Machine Learning
- Build CI/CD pipeline
- Install GitLab Runner and Trigger CI/CD pipeline
- Build Machine Learning pipeline
- Build CI/CD pipeline
- Trigger CI/CD pipeline
- Making Continuous Integration work with ML
- DVC Integration with Project
- Build a model Prototype
- Build a prototype with Jupyter Notebook
- Start to version your code with Git
- Version your code with Git
- Create pipelines
- Automate pipelines and data versioning with DVC
- Create CI pipeline to build, test, experiment
- Experimenting with DVC and CML
- Deploy your model

MLFlow

- What is MLFLow?
- Installation
- MLflow Tracking
- Where Runs Are Recorded
- How Runs and Artifacts are Recorded
- **Scenario 1:** MLFlow on localhost
- **Scenario 2:** MLFlow on localhost with SQLite
- **Scenario 3:** MLFlow on localhost with Tracking Server
- **Scenario 4:** MLFlow with remote Tracking
- Server, backend and artifact stores
- Logging Data to Runs
- Logging Functions
- Launching Multiple Runs in One Program
- Performance Tracking with Metrics
- Visualizing Metrics
- Automatic Logging
- Scikit-learn
- TensorFlow and Keras
- Gluon
- XGBoost
- Pytorch
- MLFLow Tracker
- Organizing Runs in Experiments
- Managing Experiments and Runs with the Tracking Service API
- Tracking UI
- Querying Runs Programmatically
- MLFlow Tracking Servers
- Storage
- Networking
- Logging to a Tracking Server
- MLflow Projects
- Overview
- Specifying Projects
- Running Projects
- Iterating Quickly
- Building Multi Step Workflows
- MLFLow Models
- Storage Format
- Model Signature And Input Example
- Model API
- Built-In Model Flavors
- Model Customization
- Built-In Deployment Tools
- Deployment to Custom Targets
- Model Registry

MLFlow

- ❑ Model Registry Workflows
- ❑ UI Workflow
- ❑ Registering a Model Using the Model Registry
- ❑ API Workflow
- ❑ Adding an MLFlow Model to the Model Registry
- ❑ Fetching an MLIFow Model from the Model Registry
- ❑ Serving an MLFlow Model from Model Registry
- ❑ Adding or Updating an MLFlow Model Descriptions
- ❑ Renaming an MLFlow Model
- ❑ Transitioning an MLFlow Model's Stage
- ❑ Listing and Searching MLFlow Models
- ❑ Archiving an MLFlow Model
- ❑ Deleting MLFlow Models

- ❑ **Kubernetes Foundation**
- ❑ **TFX**
- ❑ **Kubeflow**
  - ✔ What is Kubeflow?
  - ✔ Core Kubeflow components
  - ✔ How to set up Kubeflow on Kubernetes
  - ✔ How to develop basic ML models in Kubeflow Notebooks
  - ✔ How to train and deploy models in Kubeflow
  - ✔ How to use Kubeflow Pipelines
  - ✔ How to use KFServing to deploy models
  - ✔ How to manage logs with Kubeflow Metadata component
  - ✔ Katib Hyper Parameter Tuning
  - ✔ Kubeflow Pipelines to KFServing

GitLab Foundation

- GitLab Triggers
- AWS S3 storage
- GitLab CI/CD Pipelines
- Pipelines definition
- MongoDB cloud Atlas
- Heroku
- Logdata
- Coral for Monitoring

AWS MLOps

- Amazon Sagemaker
- Amazon s3
- AWS Codebuild
- AWS Codecommit
- Sagemaker Training Job
- Sage Maker Endpoint
- Amazon Api Gateway
- Sagemake Model Monitoring
- Cloudwatch Synthetics
- Cloudwatch Alarm

Digital Ocean

- Droplets
- File Transfers
- Gitops
- Jenkins
- Creating Jobs
- Creating pipelines in Jenkins
- Docker Images
- Kubernetes Flow
- Creating Clusters
- Load testing

❖ Live class video recording will be accessible till life time

❖ 1-1 mentorship will be given

❖ Unpaid internships will be given

Thanks PyBron