|  |  |
| --- | --- |
| **MODULE 7: ML WORKFLOW BEST PRACTICE** | |
| **OBJECTIVES** | * Efficient hyperparameter tuning * Experiment tracking methods and toolkits * CI/CD for ML |
| **METHODS** | Lectures, demos, code-alongs, application exercises, and structured discussion |
| **DURATION** | 5 hours for participants |

|  |  |  |  |
| --- | --- | --- | --- |
| # | SESSION | DURATION | LEARNING OBJECTIVES |
| 7.1 | Efficient hyperparameter tuning for EO data modelling | 1 hour | * Understand the importance of optimizing resource usage in computer vision tasks * Explain common approaches to efficient hyperparameter tuning |
| 7.2 | Experiment Tracking | 3 hours | * Understand the importance of taking a systematized approach to ML * Explain the core elements of experiment tracking * Describe common approaches to experiment tracking including specific toolkits * Demonstrate capacity to train and tune the hyperparameters of a model using an experiment tracking workflow |
| 7.3 | Continuous Integration / Continuous Deployment (CI/CD) with ML | 1 hour | * Understand the benefits of CI/CD for ML workflows * Explain specific aspects of CI and CD for ML workflows |

**OVERVIEW OF EXERCISES**

|  |  |  |
| --- | --- | --- |
| **#** | **Exercise Name** | **Description** |
| Ex 7.1 | **Land-Use-Land-Cover Pipeline** | Use of Google Earth Engine with Rwandan geodata to train a LULC classifier using various algorithms available in scikitlearn in a systematized experiment tracking environment employing WandB |
| Ex 7.2 | **Deep Learning for Crop Yield Estimation** | Use of Google Earth Engine with US geodata to train a CNN in pytorch to predict crop yield Crop yield prediction in a systematized experiment tracking environment employing WandB |

**7.1** **Efficient hyperparameter tuning for EO data modelling**

**7.1 Quiz questions**

|  |  |
| --- | --- |
| 1. What is the purpose of hyperparameter tuning in machine learning models? | 1. To reduce the size of the dataset 2. To optimize the performance of the model 3. To increase the complexity of the model 4. To decrease the accuracy of the model |
| 2. Why is a resource-conscious approach important when tuning hyperparameters in Earth Observation (EO) data modeling? | 1. EO datasets are small and low-dimensional 2. EO datasets are often large, high-dimensional, and multi-spectral, which impacts the training time and resource requirements 3. EO datasets are always easy to process 4. EO datasets do not require any resource management |
| 3. Which hyperparameter tuning method uses a probabilistic model to estimate the performance of different hyperparameter combinations? | 1. Grid Search 2. Random Search 3. Bayesian Optimization 4. Genetic Algorithms |
| 4. How does the Random Search method reduce computational burden compared to grid search? | 1. By evaluating all possible combinations of hyperparameters 2. By using a subset of possible values for each hyperparameter 3. By assigning the same value to all hyperparameters 4. By increasing the range of possible values available for tuning |
| 5. What is the key advantage of Tree-structured Parzen Estimators (TPE) in hyperparameter tuning? | 1. TPE prioritizes combinations with better performance by evolving a population of hyperparameters 2. TPE uses a probabilistic model that iteratively updates its estimation to guide the search process 3. TPE allows you to terminate poorly performing evaluations before they complete 4. TPE efficiently explores the hyperparameter space by leveraging the collective knowledge of the swarm |
| 6. How can early stopping and pruning techniques be beneficial in hyperparameter tuning for EO tasks? | 1. They can be used to increase the size of the dataset 2. They allow termination of poorly performing evaluations before they complete, thereby speeding up the overall tuning process 3. They ensure that all possible combinations of hyperparameters are evaluated 4. They encourage the use of a larger grid and more hyperparameters |

**7.1 Resources:**

* Article: [Comparing Modern Scalable Hyperparameter Tuning Methods](https://towardsdatascience.com/comparing-modern-scalable-hyperparameter-tuning-methods-dfe9661e947f) | by Ayush Chaurasia | Towards Data Science
* Paper: [Practical Bayesian Optimization Of Machine Learning Algorithms](https://arxiv.org/pdf/1206.2944.pdf)

**7.2** **Experiment Tracking**

**7.2 Quiz questions**

|  |  |
| --- | --- |
| 1. What are the three main components that need to be version controlled in a machine learning project? | 1. Source code, data, and hyperparameters 2. Source code, data, and model artifacts 3. Source code, model artifacts, and performance metrics 4. Source code, data, and team communication |
| 2. How does Git Large File Storage (Git LFS) enhance the functionality of Git for managing large files in machine learning projects? | 1. By compressing large files to reduce their size 2. By creating multiple copies of large files for redundancy 3. By replacing large files with lightweight text pointers within your Git repository while storing the actual file contents on a separate server 4. By encrypting large files for secure storage |
| 3. What are the three different types of experiment parameters that should be documented in machine learning? | 1. Hyperparameter Logging, Performance Metrics Logging, and Experiment Metadata Logging 2. Data Version Control, Model Artifacts Management, and Hyperparameter Logging 3. Hyperparameter Logging, Model Artifacts Management, and Data Version Control 4. Source Code Version Control, Performance Metrics Logging, and Experiment Metadata Logging |
| 4. How does experiment parameters documentation contribute to comparability and reproducibility in machine learning workflows? | 1. By providing a consistent format for documenting all changes 2. By logging parameters, you can easily compare different experiments and recreate a specific experiment 3. By reducing the need for manual tracking of changes 4. By providing a platform for team discussion and feedback |
| 5. Which of the following are common types of visualizations used in experiment tracking in machine learning? | 1. Performance curves, confusion matrices, and ROC curves 2. Bar charts, line graphs, and scatter plots 3. Pie charts, flow diagrams, and Gantt charts 4. Histograms, box plots, and violin plots |
| 6. Which of the following features are provided by experiment tracking toolkits to facilitate collaboration among team members? | 1. Web-based dashboard, sharing and permissions, integration with version control systems, comments and annotations, and exporting and reporting 2. Real-time chat, video conferencing, and task assignment 3. Email notifications, document sharing, and calendar scheduling 4. Cloud storage, password protection, and mobile access |

**Resources (articles / tutorials / videos):**

* Article - [10 tips for machine learning experiment tracking and reproducibility: Do it yourself approach without additional tooling](https://developer.ibm.com/blogs/10-diy-tips-for-machine-learning-experiment-tracking-and-reproducibility/) – IBM Developer
* Artcile: [Machine Learning Experiment Tracking with WandB](https://towardsdatascience.com/machine-learning-experiment-tracking-93b796e501b0)
* Article: [WandB — The Best MLOps Platform](https://medium.com/mlearning-ai/wandb-the-best-mlops-platform-bf3aa31b162e)
* MOOC: [Effective MLOps: Model Development with WandB](https://www.wandb.courses/courses/effective-mlops-model-development?utm_source=youtube&utm_medium=video&utm_campaign=course-video-ad)

**7.3 CI/CD with ML**

**7.3 Quiz questions**

|  |  |
| --- | --- |
| 1. What are key aspects of CI in the ML context? | 1. Automated model training and validation, version control for data, code, and models, automated testing, monitoring, and alerting. 2. Automated testing, rollback mechanisms, model monitoring in production, continuous improvement. 3. Automated model training and validation, model deployment automation, rollback and versioning, continuous improvement. 4. Version control for data, code, and models, model deployment automation, model |
| 2. What are key aspects of CD in the ML context? | 1. Model deployment automation, rollback and versioning, model monitoring in production, continuous improvement. 2. Automated model training and validation, version control for data, code, and models, automated testing, monitoring, and alerting. 3. Automated model training and validation, model deployment automation, rollback and versioning, continuous improvement. 4. Version control for data, code, and models, model deployment automation, model monitoring in production, continuous improvement. |
| 3. How does CI/CD support the unique requirements of ML workflows? | 1. By automating the build, testing, and deployment processes. 2. By facilitating collaboration, reproducibility, and model quality. 3. Both a and b. 4. Neither a nor b. |
| 4. Why should CD practices in ML include rollback mechanisms and model versioning? | 1. To ensure that you can easily revert to a previous version of a model if needed. 2. To maintain an up-to-date model serving infrastructure and reduce manual intervention. 3. To track metrics like accuracy, latency, and resource utilization. 4. To incorporate feedback from production systems, users, or new data. |

**Resources (articles / tutorials / videos):**

* Organization: [MLOPS.org](https://ml-ops.org/)
* MOOC: [Effective MLOps: Model Development with WandB](https://www.wandb.courses/courses/effective-mlops-model-development?utm_source=youtube&utm_medium=video&utm_campaign=course-video-ad)