Training Course on CI/CD for Machine Learning Pipelines

Professional Development Training Course Outline

Category	Data Science	**Duration**	10 Days
Base Fee	$3,000 USD	**Delivery Mode**	Online / On-site Hybrid

Course Introduction

In the rapidly evolving landscape of Artificial Intelligence and Machine Learning, the efficiency and reliability of deploying machine learning models to production are paramount. Traditional software development CI/CD practices, while foundational, require significant adaptation to address the unique challenges of ML workflows, including data versioning, model reproducibility, and continuous model retraining. This comprehensive training course delves into the critical principles and practical applications of **CI/CD for Machine Learning Pipelines**, equipping participants with the essential skills to automate, streamline, and scale their MLOps initiatives. We will explore how **Continuous Integration**, **Continuous Delivery**, and **Continuous Deployment** paradigms transform the entire ML lifecycle, from data ingestion to model monitoring, ensuring robust, high-performing, and consistently updated ML systems.

Training Course on CI/CD for Machine Learning Pipelines: Automating ML Workflow Integration and Delivery emphasizes a hands-on, practical approach to building **automated ML pipelines**, integrating best-in-class tools and methodologies. Participants will gain deep insights into orchestrating complex ML workflows, implementing rigorous testing strategies for both code and data, and deploying models with confidence. By mastering the integration of **DevOps principles** with **Machine Learning Operations (MLOps)**, attendees will learn to reduce manual errors, accelerate model iteration cycles, and significantly improve the overall stability and scalability of their predictive solutions. This course is a vital step for any organization aiming to achieve truly agile and production-ready machine learning.

Programme Curriculum

Training Course on CI/CD for Machine Learning Pipelines: Automating ML Workflow Integration and Delivery

Introduction

In the rapidly evolving landscape of Artificial Intelligence and Machine Learning, the efficiency and reliability of deploying machine learning models to production are paramount. Traditional software development CI/CD practices, while foundational, require significant adaptation to address the unique challenges of ML workflows, including data versioning, model reproducibility, and continuous model retraining. This comprehensive training course delves into the critical principles and practical applications of **CI/CD for Machine Learning Pipelines**, equipping participants with the essential skills to automate, streamline, and scale their MLOps initiatives. We will explore how **Continuous Integration**, **Continuous Delivery**, and **Continuous Deployment** paradigms transform the entire ML lifecycle, from data ingestion to model monitoring, ensuring robust, high-performing, and consistently updated ML systems.

Training Course on CI/CD for Machine Learning Pipelines: Automating ML Workflow Integration and Delivery emphasizes a hands-on, practical approach to building **automated ML pipelines**, integrating best-in-class tools and methodologies. Participants will gain deep insights into orchestrating complex ML workflows, implementing rigorous testing strategies for both code and data, and deploying models with confidence. By mastering the integration of **DevOps principles** with **Machine Learning Operations (MLOps)**, attendees will learn to reduce manual errors, accelerate model iteration cycles, and significantly improve the overall stability and scalability of their predictive solutions. This course is a vital step for any organization aiming to achieve truly agile and production-ready machine learning.

Course Duration

10 days

Course Objectives

1. Master the core concepts of CI/CD and their specialized application in Machine Learning Operations (MLOps).
2. Design and implement robust automated ML pipelines for data ingestion, feature engineering, model training, and evaluation.
3. Utilize version control systems (e.g., Git, DVC) effectively for managing code, data, and model artifacts in reproducible ML workflows.
4. Implement Continuous Integration (CI) practices for ML code, ensuring frequent integration and automated testing.
5. Develop Continuous Delivery (CD) strategies for ML models, enabling rapid and reliable model deployments to production environments.
6. Explore Continuous Deployment (CDP) for fully automated model releases, leveraging advanced deployment patterns like canary releases and A/B testing.
7. Integrate and leverage popular MLOps tools and platforms (e.g., MLflow, Kubeflow, Airflow, Jenkins, GitLab CI/CD, GitHub Actions) for pipeline orchestration.
8. Establish comprehensive automated testing frameworks for ML models, including data validation, model performance validation, and bias detection.
9. Implement monitoring and alerting solutions for deployed ML models, detecting model drift, data drift, and performance degradation.
10. Understand and apply Infrastructure as Code (IaC) principles to provision and manage ML infrastructure for scalable pipelines.
11. Optimize hyperparameter tuning and experimentation through automated CI/CD processes, accelerating model improvement.
12. Ensure model reproducibility and auditability across the entire ML lifecycle using proper versioning and lineage tracking.

13. Implement security best practices within CI/CD pipelines for ML, safeguarding data and models in production.

Organizational Benefits

- Significantly reduce the time-to-market for new ML models and updates.
- Minimize deployment risks and production issues through automated testing and continuous monitoring.
- Foster seamless collaboration between data scientists, ML engineers, and operations teams.
- Ensure consistent and verifiable model outcomes, crucial for auditing and debugging.
- Automate repetitive tasks, leading to lower manual effort and fewer errors.
- Enable rapid iteration and testing of new ideas and model architectures.
- Build robust and scalable ML systems capable of handling increasing data volumes and model complexity.
- Identify and resolve model performance degradation or data quality issues early.
- Implement better data versioning and lineage tracking for compliance and data quality.

Target Audience

1. Machine Learning Engineers
2. Data Scientists
3. DevOps Engineers
4. AI/ML Architects
5. Software Engineers
6. Data Engineers
7. Technical Leads
8. Anyone involved in the lifecycle of deploying and maintaining machine learning models in production

Course Outline

Module 1: Introduction to CI/CD and MLOps

- Understanding the MLOps Landscape: Evolution from traditional software development to MLOps.
- Core Principles of CI/CD: Continuous Integration, Continuous Delivery, Continuous Deployment.
- Unique Challenges in ML Pipelines: Data versioning, model drift, reproducibility, computational resources.
- Benefits of CI/CD for ML: Faster iterations, reliability, scalability, collaboration.
- **Case Study:** How a FinTech company reduced model deployment time by 70% using a nascent MLOps CI/CD pipeline.

Module 2: Version Control for ML Projects

- Git for Code Versioning: Branching strategies, pull requests, collaborative development.
- Data Version Control (DVC): Tracking large datasets and linking them to code commits.
- Model Versioning: Managing different iterations of trained models and their metadata.
- Artifact Management: Storing and retrieving preprocessed data, feature sets, and model binaries.
- **Case Study:** A retail giant ensuring data consistency across multiple ML teams using DVC and Git.

Module 3: Setting Up Your CI/CD Environment

- Choosing CI/CD Tools: Jenkins, GitLab CI/CD, GitHub Actions, Azure DevOps, CircleCI.
- Containerization with Docker: Packaging ML models and their dependencies for consistent environments.
- Orchestration with Kubernetes (Introduction): Managing containerized workloads for scalable deployments.
- Cloud Platforms for MLOps: AWS SageMaker, Google Vertex AI, Azure ML.
- **Case Study:** A healthcare startup leveraging GitHub Actions and Docker to automate their ML model build process.

Module 4: Building Automated ML Pipelines

- Pipeline Orchestration Tools: Apache Airflow, Kubeflow Pipelines, MLflow Projects.
- Defining Pipeline Stages: Data ingestion, data validation, feature engineering, model training, evaluation.
- Parameterization and Configuration Management: Making pipelines flexible and reusable.
- Dependency Management: Ensuring all required libraries and environments are consistent.
- **Case Study:** An e-commerce platform building an automated recommendation engine retraining pipeline using Airflow.

Module 5: Continuous Integration for Machine Learning

- Automated Code Testing: Unit tests, integration tests for ML code components.
- Data Validation in CI: Schema checks, range checks, anomaly detection on incoming data.
- Pre-commit Hooks and Linting: Enforcing code quality and standards.
- CI Triggers: Event-based triggers (e.g., code push, new data arrival).
- **Case Study:** A logistics company using GitLab CI/CD to run automated tests on their fraud detection model's data preprocessing logic.

Module 6: Model Training and Experiment Tracking

- Automated Model Training: Triggering retraining based on data updates or schedule.
- Hyperparameter Tuning Automation: Integrating tools for efficient hyperparameter search (e.g., Optuna, Ray Tune).
- Experiment Tracking with MLflow: Logging parameters, metrics, and artifacts for reproducibility.
- Model Registry: Centralizing trained models for easy access and deployment.
- **Case Study:** A marketing analytics firm optimizing campaign performance by automating hyperparameter tuning with MLflow tracking.

Module 7: Model Evaluation and Validation in CI/CD

- Automated Model Evaluation Metrics: Accuracy, precision, recall, F1-score, AUC.
- Baseline Model Comparison: Ensuring new models outperform existing ones.
- Bias and Fairness Testing: Detecting and mitigating biases in model predictions.
- Data Drift and Concept Drift Detection: Monitoring shifts in data distribution and model performance.
- **Case Study:** A credit scoring agency implementing automated fairness checks in their CI/CD pipeline to ensure ethical model deployments.

Module 8: Continuous Delivery for Machine Learning Models

- Model Packaging and Export: Preparing models for deployment (e.g., ONNX, TensorFlow SavedModel, Pickle).
- Creating Deployable Artifacts: Containerizing models with prediction APIs.
- Release Management: Versioning and managing model releases.
- Automated Deployment to Staging Environments: Testing models in a production-like setting.
- **Case Study:** A media company automating the packaging and delivery of their content recommendation models to a staging environment.

Module 9: Advanced Deployment Strategies

- Blue/Green Deployments: Minimizing downtime during model updates.
- Canary Releases: Gradually rolling out new model versions to a subset of users.
- A/B Testing for Models: Comparing different model versions in production.
- Shadow Deployments: Running a new model alongside the old without impacting users.
- **Case Study:** An online streaming service using canary deployments to safely introduce new personalized video recommendation models.

Module 10: Model Monitoring and Alerting

- Real-time Model Performance Monitoring: Tracking latency, throughput, error rates.
- Drift Detection in Production: Identifying data drift and concept drift in live environments.
- Setting Up Alerting Systems: Notifying teams of performance degradation or anomalies.
- Logging and Observability for ML: Centralized logging, distributed tracing.
- **Case Study:** A manufacturing company implementing real-time monitoring to detect anomalies in their predictive maintenance models, preventing costly downtime.

Module 11: Retraining and Feedback Loops

- Automated Retraining Triggers: Based on performance degradation, new data availability, or schedule.
- Continuous Training (CT): Integrating retraining into the CI/CD pipeline.
- Human-in-the-Loop Feedback: Incorporating expert feedback into model improvement.
- Rollback Strategies: Reverting to previous stable model versions.
- **Case Study:** A social media platform implementing continuous retraining for their spam detection model based on new user feedback and reported content.

Module 12: Infrastructure as Code (IaC) for MLOps

- Terraform for Infrastructure Provisioning: Managing cloud resources for ML pipelines.
- CloudFormation (AWS) / ARM Templates (Azure) / Deployment Manager (GCP): Platform-specific IaC.
- Managing ML Compute Resources: GPUs, TPUs, distributed training clusters.
- Cost Optimization for ML Infrastructure: Rightsizing and auto-scaling.
- **Case Study:** A pharmaceutical company using Terraform to provision a reproducible and scalable environment for their drug discovery ML models.

Module 13: Security and Compliance in ML CI/CD

- Secure Coding Practices for ML: OWASP Top 10 for ML.
- Data Privacy and Anonymization in Pipelines: GDPR, HIPAA considerations.
- Access Control and Authentication: Securing access to data and models.

- Vulnerability Scanning and Dependency Management: Identifying and addressing security risks.
- **Case Study:** A financial institution implementing strict access controls and regular security audits within their CI/CD pipeline for compliance.

Module 14: Team Collaboration and MLOps Best Practices

- Establishing MLOps Culture: Bridging the gap between data science, engineering, and operations.
- Cross-functional Team Structures: Data Scientists, ML Engineers, DevOps Engineers.
- Documentation and Knowledge Sharing: Maintaining clear records of models and pipelines.
- Scalability Considerations for MLOps Teams: Managing growth and complexity.
- **Case Study:** A large tech company fostering a collaborative MLOps culture leading to faster innovation cycles and fewer production incidents.

Module 15: Advanced Topics and Future Trends in MLOps

- MLOps Platforms Deep Dive: Comparing and contrasting popular commercial and open-source platforms.
- Explainable AI (XAI) in CI/CD: Integrating interpretability into the pipeline.
-

Upcoming Scheduled Sessions

Start Date	End Date	Location	Price
23 Nov 2026	04 Dec 2026	Online	$2,000
23 Nov 2026	04 Dec 2026	Physical	$3,000
23 Nov 2026	04 Dec 2026	Physical	$6,000
23 Nov 2026	04 Dec 2026	Physical	$6,000
23 Nov 2026	04 Dec 2026	Physical	$12,000
23 Nov 2026	04 Dec 2026	Physical	$9,000
23 Nov 2026	04 Dec 2026	Physical	$12,000
23 Nov 2026	04 Dec 2026	Physical	$12,000

Ready to enroll or request a customized program? Book online or contact us:

Register Online Now