datacamp

COURSE GLOSSARY

MLOps Concepts

API (Application Programming Interface): A defined set of inputs, outputs, and communication rules that allow different software components or services to interact

CI/CD pipeline: An automated workflow combining continuous integration (frequent automated testing and merging of changes) and continuous deployment (automated release of validated changes) to accelerate safe delivery

Concept drift: A change in the underlying relationship between input features and the target variable so that past learned patterns no longer hold

Container: A lightweight, portable package that bundles an application and all its dependencies so it runs consistently across different environments

Data drift: A change over time in the distribution or characteristics of input features that can affect a model's predictive accuracy

Data pipeline: An automated workflow that extracts data from sources, transforms or cleans it, and loads it into storage or systems where it can be used for analysis or model training

Data quality: A measure of how well data serves its intended purpose, commonly evaluated by dimensions such as accuracy, completeness, consistency, and timeliness

DevOps: A set of practices and tools that integrate software development and IT operations to enable faster, more reliable, and automated software delivery

ETL (Extract, Transform, Load): A common data ingestion pattern that extracts raw data from sources, transforms it into the required format, and loads it into a target database or warehouse

Experiment tracking: The systematic logging of model training runs, including data versions, code, hyperparameters, metrics, and artifacts, to compare, reproduce, and share results

Feature engineering: The process of selecting, creating, and transforming raw data into meaningful input variables (features) that improve a model's predictive performance

Feature store: A centralized repository for storing, discovering, and serving reusable features across teams and projects to ensure consistency and faster development

Feedback loop: The process of collecting ground-truth outcomes for model predictions, comparing them to model outputs, and using those results to evaluate performance and guide improvements or retraining

Inferencing: The process of using a deployed machine learning model to generate predictions or decisions on new, unseen input data

Machine learning lifecycle: The end-to-end, iterative process that takes an ML project from problem design and data preparation through model development, deployment, and ongoing maintenance

Microservices architecture: An approach that structures an application as a collection of small, independently deployable services that communicate over APIs for better scalability and fault isolation

MLOps maturity: A characterization of how advanced an organization's MLOps practices are—assessed by the level of automation, collaboration, traceability, and monitoring—often described in distinct maturity levels.

MLOps: The set of practices, tools, and cultural principles for designing, deploying, and maintaining machine learning systems in production continuously, reliably, and efficiently

Monitoring: The ongoing observation of model and system behavior using statistical checks (data and prediction distributions, performance metrics) and computational checks (latency, throughput, resource usage) to detect degradation or failures

Retraining: The process of updating a model by training it on new data (or a combination of new and historical data) to restore or improve performance in response to drift or changing requirements

Runtime environment: The software and hardware context in which code executes (e.g., OS, language and library versions, and compute resources) that can affect program behavior