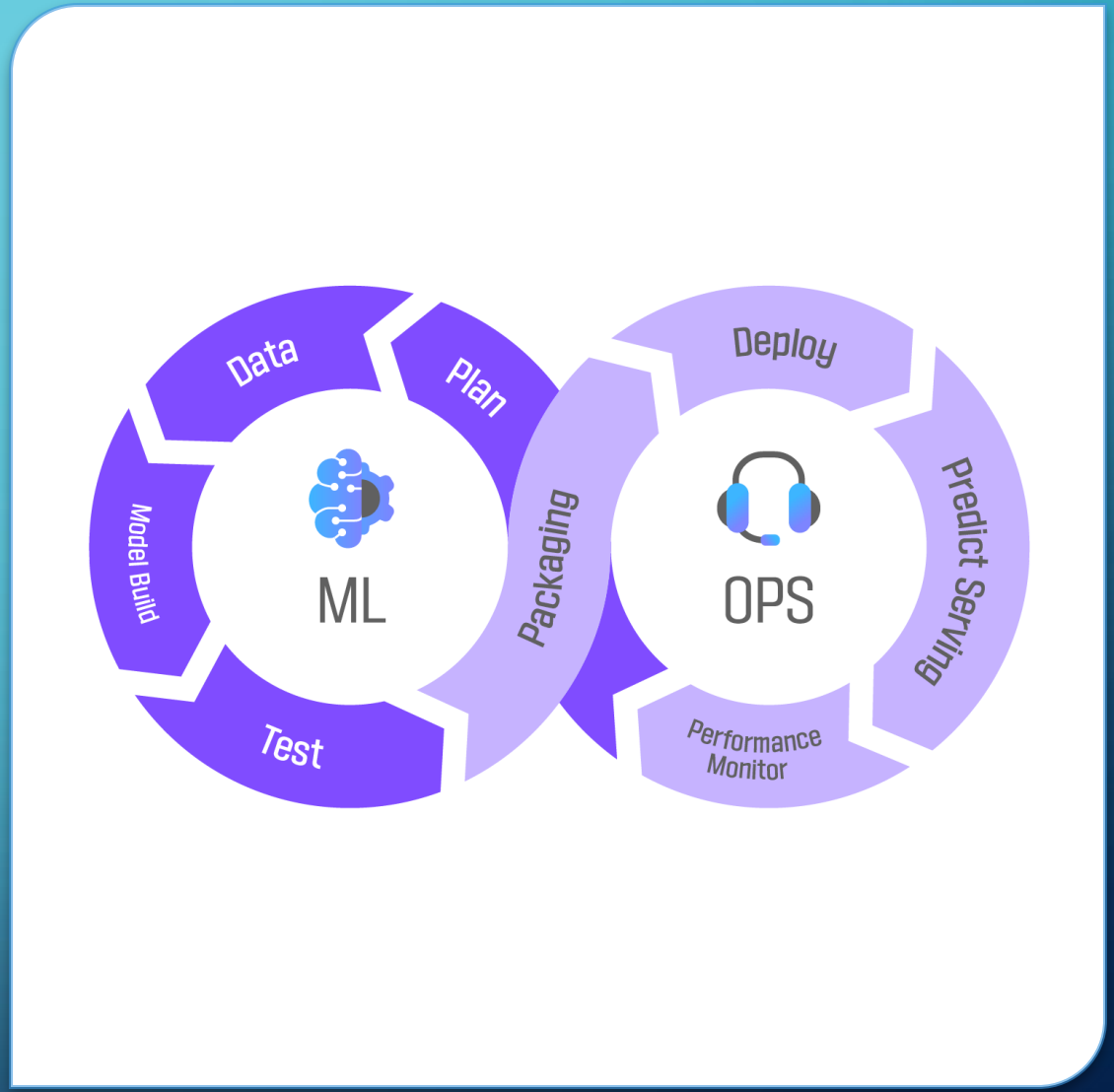
MLOPS : BACKEND

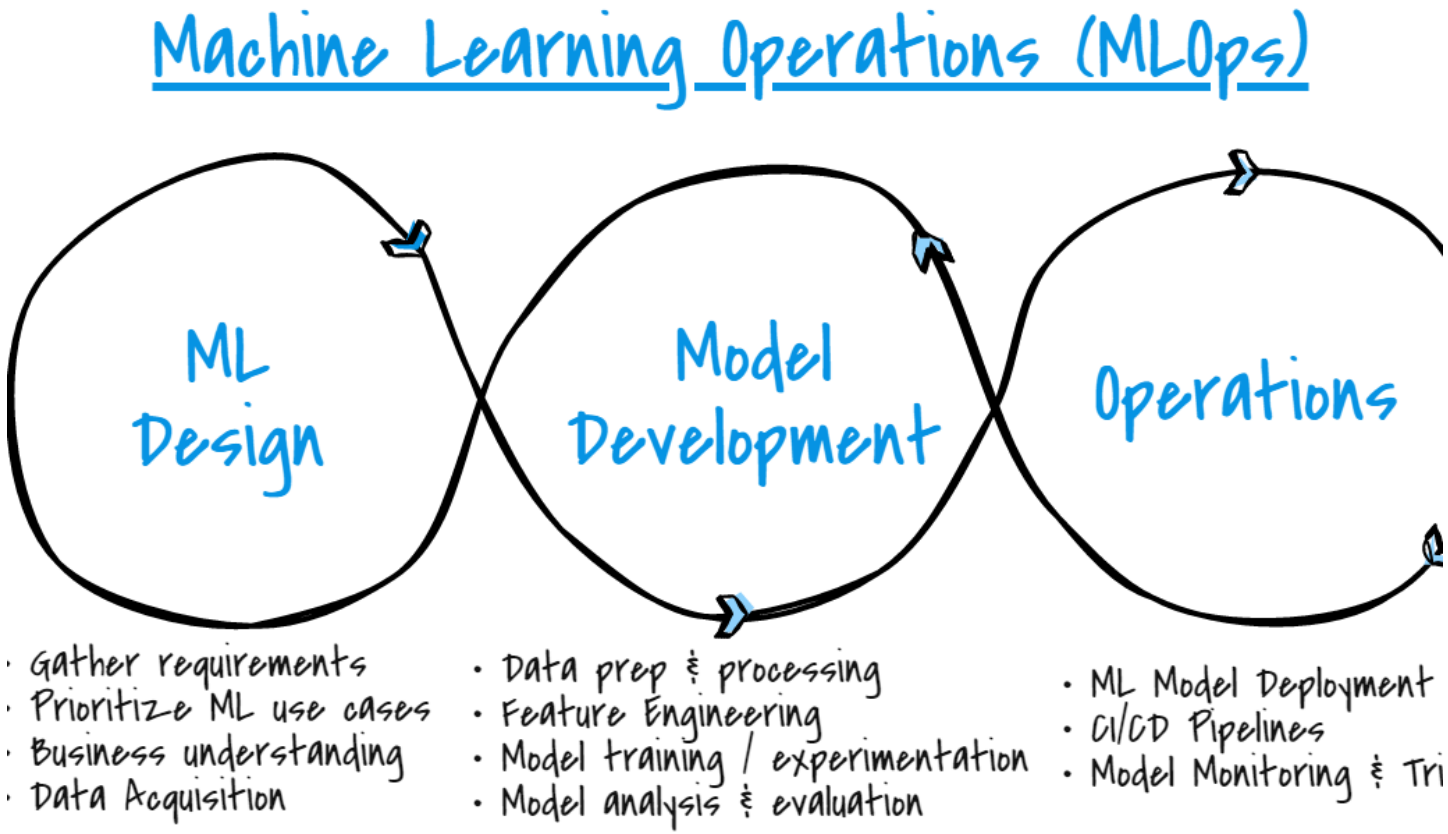
IMPLEMENTING:

- **MLFLOW FOR MONITORING AND TRACKING ML WORKFLOWS**
- **DATA VERSIONING FOR ML RUNS**



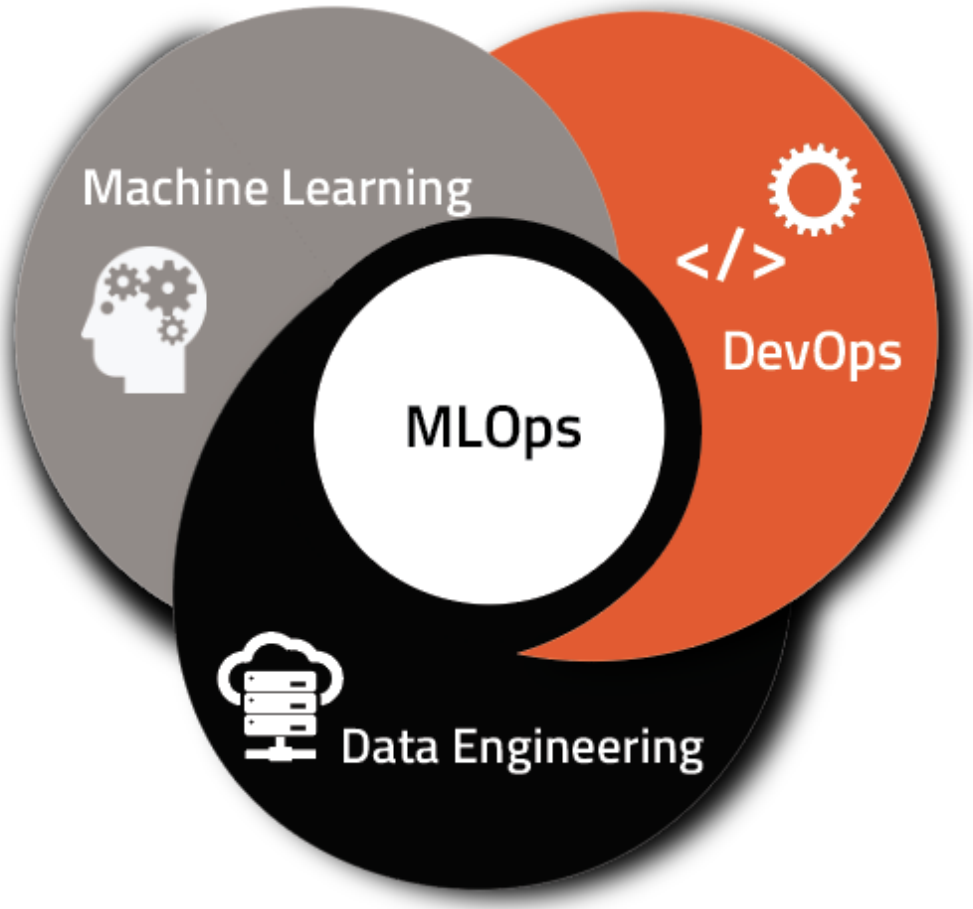
WHAT ARE MLOPS?

MLOps (Machine Learning Operations) is a set of practices aimed at streamlining and automating the lifecycle of machine learning models. It ensures efficient collaboration between data scientists and operations teams, addressing challenges related to model deployment, monitoring, scaling, and reproducibility.



KEY OBJECTIVES OF MLOPS:

- **Automation & Efficiency** – Reduces manual intervention in ML workflows.
- **Model Versioning & Reproducibility** – Ensures consistency in model training.
- **Monitoring & Performance Tracking** – Provides visibility into model performance.
- **Scalability & Deployment** – Facilitates deploying models at scale.
- **Data & Model Governance** – Maintains compliance and auditability.



WHY MLFLOW?

MLFlow is an open-source platform that helps manage the entire machine learning lifecycle, from data ingestion to model deployment. It provides capabilities for:

- **Experiment tracking** – Logging and comparing different PyTorch models.
- **Model versioning** – Managing different iterations of models.
- **Data versioning** – Ensuring reproducibility through DVC and Git.
- **Deployment & Monitoring** – Deploying models and tracking their performance over time.

MLFLOW SETUP AND WORKFLOW COMPONENTS

The MLFlow setup involves multiple components to ensure smooth tracking and reproducibility of ML experiments:

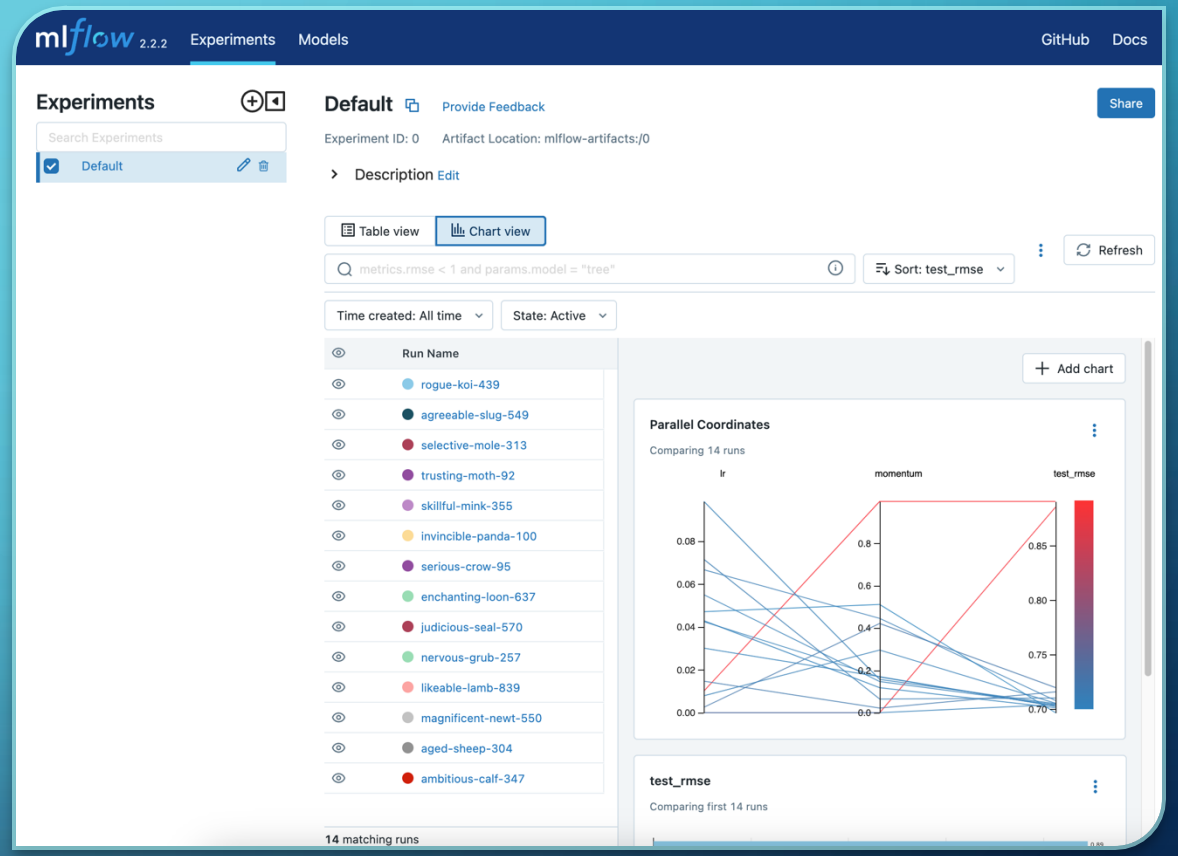
1. Data Ingestion
2. Model Training & Evaluation
3. MLFlow Experiment Tracking
4. Model Versioning

DATA INGESTION

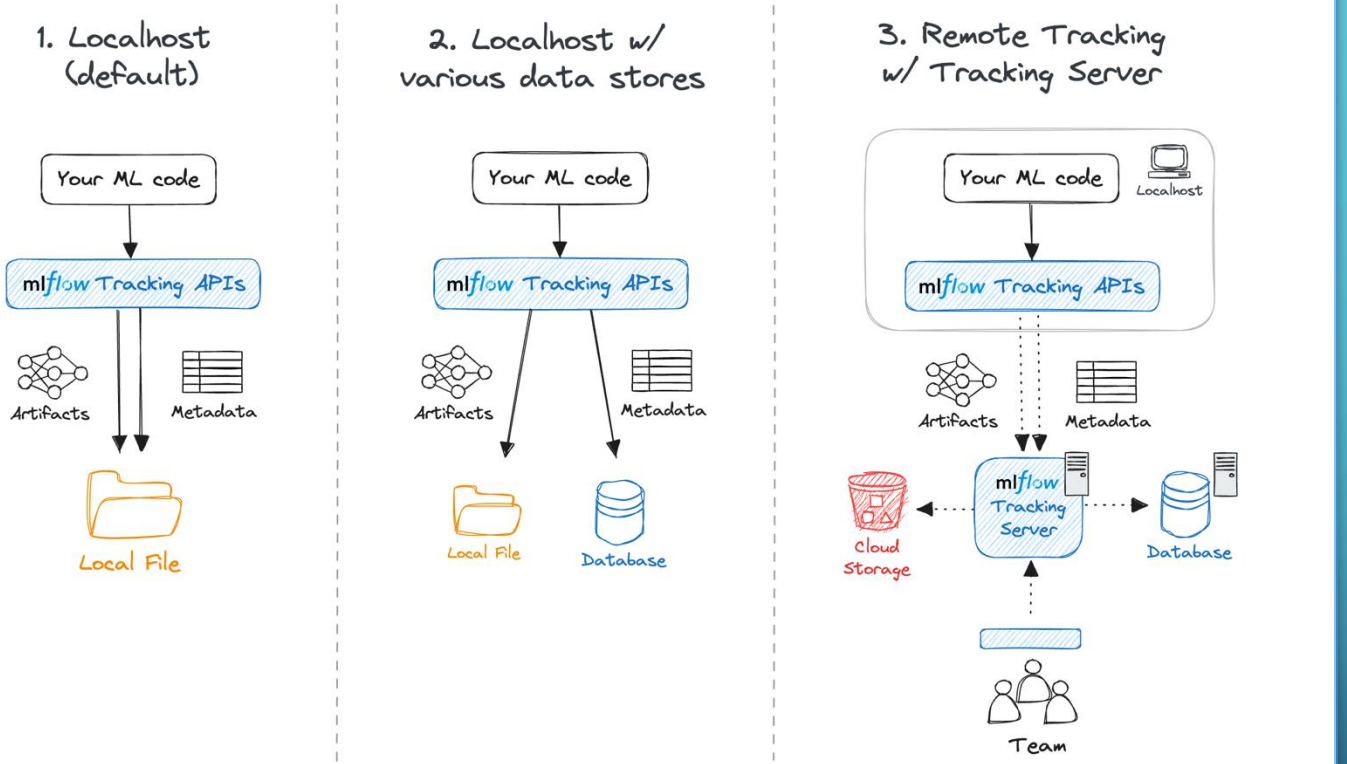
- Data is ingested into the environment through a script that retrieves data from a database (not currently connected, using sample data from MNE).
- The script allows manual replacement of the dataset for testing and validation.
- Ensuring proper data preprocessing and transformation is critical before training.

MODEL TRAINING & EVALUATION

- MLFlow tracks various parameters used during model training:
  - **Hyperparameters**: Batch size, number of epochs, learning rate, optimizer, and loss function.
  - **Performance Metrics**: Accuracy, loss, validation performance, etc.
  - **Comparison of different runs**: Helps identify the best-performing model.
- The Deep Learning team will review the training and test scripts to ensure consistency in data transformation.



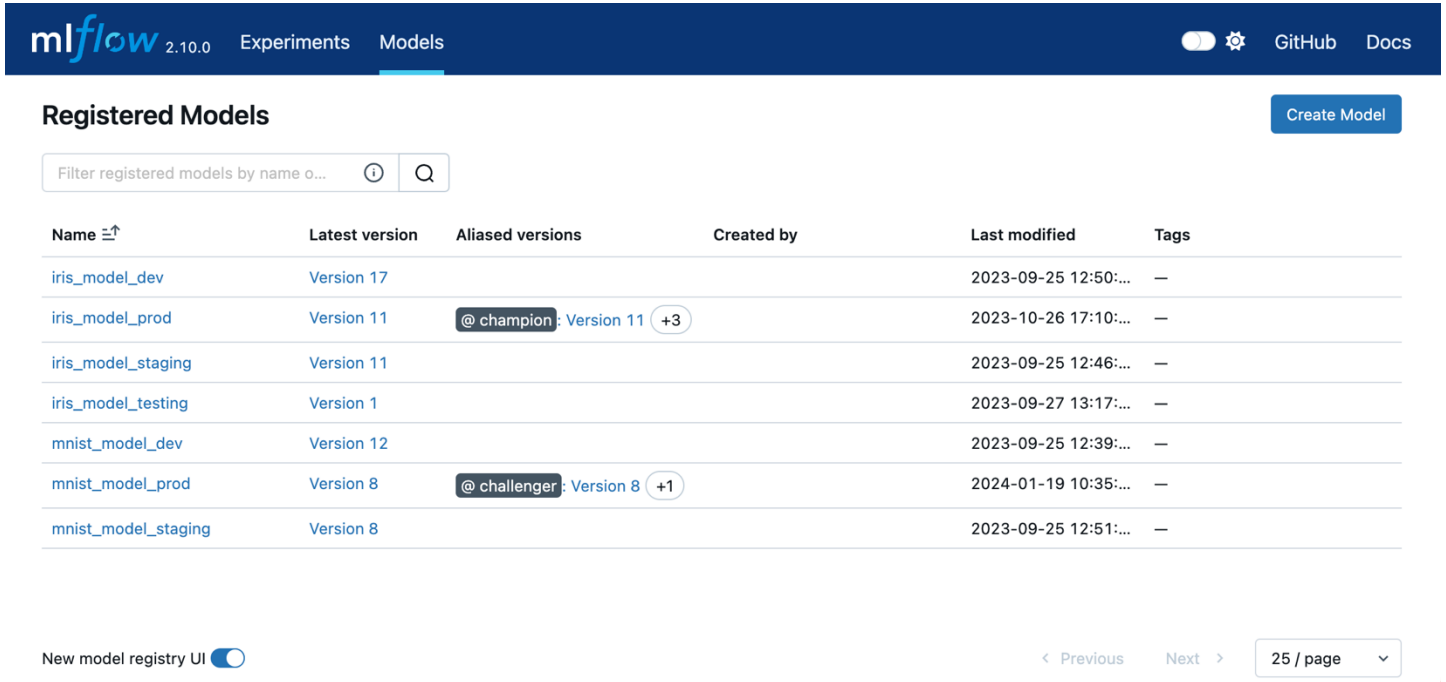
MLFLOW EXPERIMENT TRACKING



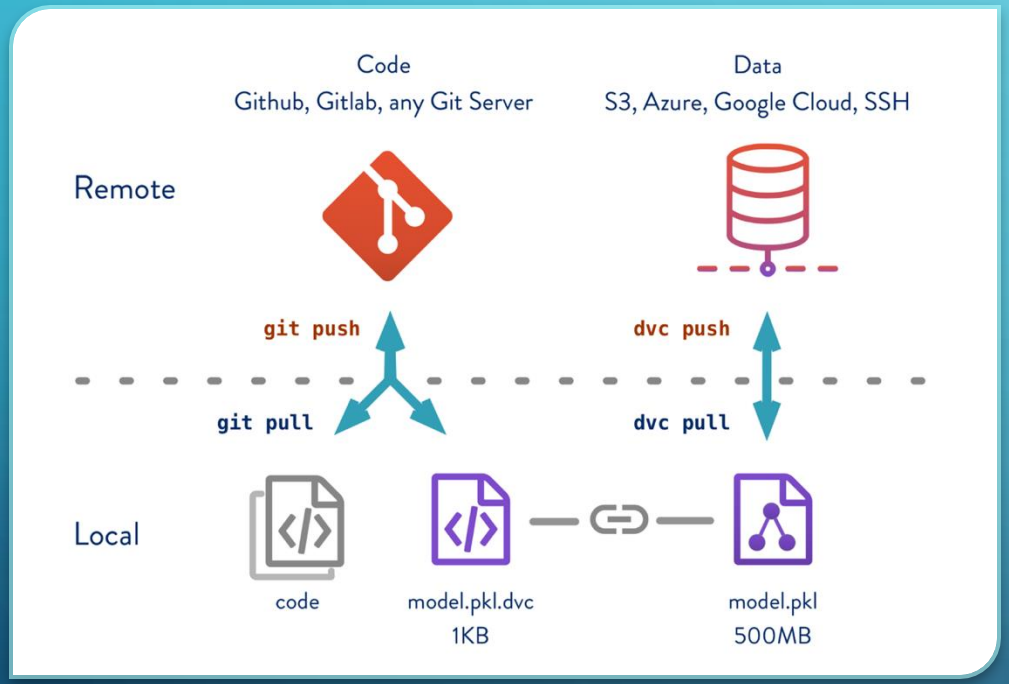
- MLFlow UI provides visualization for all tracked metrics.
- Logs the inputs and outputs of ML runs to ensure reproducibility.
- Saves model artifacts for deployment and further experimentation.

MODEL VERSIONING

- MLFlow tracks and versions models to allow rollbacks and comparisons.
- Each trained model is assigned a unique version, enabling seamless transition from experimentation to production.



DVC & GIT FOR DATA VERSIONING

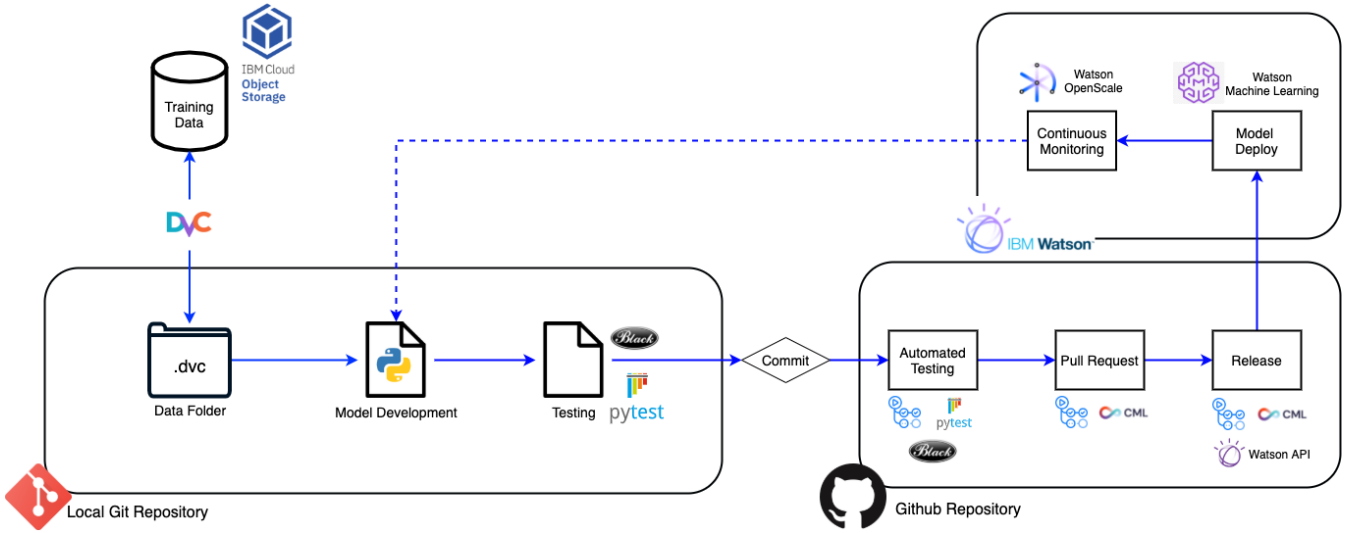


Purpose of DVC:

- **Version Control for Data:** DVC ensures that different versions of datasets are properly tracked, just like code.
- **Reproducibility:** Links data versions with ML experiments to guarantee that models can be reproduced with the same input data.
- **Collaboration:** Enables teams to work efficiently by tracking changes and sharing dataset versions.
- **Storage Optimization:** Keeps large datasets in remote storage while maintaining lightweight *.dvc* metadata files in Git.
- **Integration with MLFlow:** Ensures that both data and model versions are tracked together for a complete ML pipeline history.

HOW DVC WORKS WITH GIT:

1. **Tracking Data Changes:**
    1. When data changes, a Git commit is created.
    2. DVC assigns a unique hash value to track the dataset.
2. **Storage and Accessibility:**
    1. The .dvc file containing dataset metadata is stored in Git.
    2. The actual dataset is stored locally (in the DVC folder and cache) or can also be externally (e.g., Google Drive, cloud storage).
3. **Reproducibility and Collaboration:**
    1. Different team members can pull the exact dataset version for their experiments.
    2. Ensures that models are always trained on the correct dataset.



MLOPS WORKFLOW