Tackling Data Version Management in MLOps using DVC

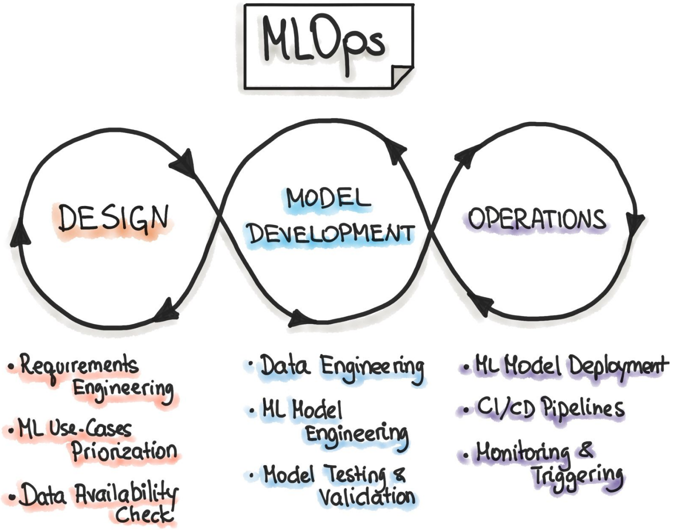
Sampad Kumar Kar
Shourjya Basu

20th November 2023

Introduction to MLOps

- MLOps is a set of practices that combine machine learning (ML) systems development and operations (Ops) to streamline the end-to-end ML lifecycle, from development to deployment and maintenance.
- Key objectives of MLOps:
  - **Collaboration**
  - **Reproducibility**
  - **Automation**
  - **Scaling**

Components of MLOps



Intoduction to Version Control

- Version control is a system that tracks changes to code and other data over time, allowing users to revert to previous versions if necessary.
- Types of versioning in ML projects:
  - **Project Versioning**
  - **Data Versioning**
  - **Model Versioning**
  - **Deployment-based Versioning**

Introduction to Data Versioning

- Data versioning refers to the practice of systematically managing and tracking changes to datasets over time.
- Its primary uses are:
  - **Tracking Changes to Datasets**
  - **Reproducibility**
  - **Collaboration**
  - **Automation in Machine Learning Pipelines**

Research Questions

- What is the current state of data versioning in the marketplace?
- What are the requirements that need to be considered when considering applying data versioning into the project pipeline?
- What kind of approaches or frameworks can be used for building an integrated ML system and to continuously operate it in production?

Data Versioning Tool: DVC

- DVC is an Open-source, Git-based data science library, developed by iterative.ai, which enables users to apply version control to machine development by tracking ML Models and Data sets.
- It provides 3 main functions:
  - **ML Project Version Control**
  - **ML Experiment Management**
  - **Deployment and Collaboration**

Demo Problem

Changing dataset by Modifying Random seed

- Create a simple classification model using a random seed for train-test split.
- Modify the random seed and observe the impact on model performance.
- Track changes to the dataset and model using DVC.

Solution offered using DVC

- Easily reproduce different versions of the dataset and model.
- Compare model performance across different data versions.

DVC Implementation

- **Initialise Git and DVC**
- **Versioning ML artefacts**
- **Storing versioned ML artefacts**
- **Retrieving ML artefacts**
- **Making changes on dataset**
- **Switching between versions**

Conclusion/Key takeaways

- Data version control is essential for reproducible and reliable machine learning.
- DVC provides a powerful and easy-to-use tool for data version control in ML projects.
- Adopting DVC can significantly improve the efficiency and effectiveness of ML development and deployment.

References

- *Tackling Version Management and Reproducibility in MLOps*
  Priscilla Dias Melin
- *On the Co-evolution of ML Pipelines and Source Code - Empirical Study of DVC Projects*
- *Software Engineering for Machine Learning: A Case Study*
- *Creating reproducible data science workflows with DVC*

THANK YOU