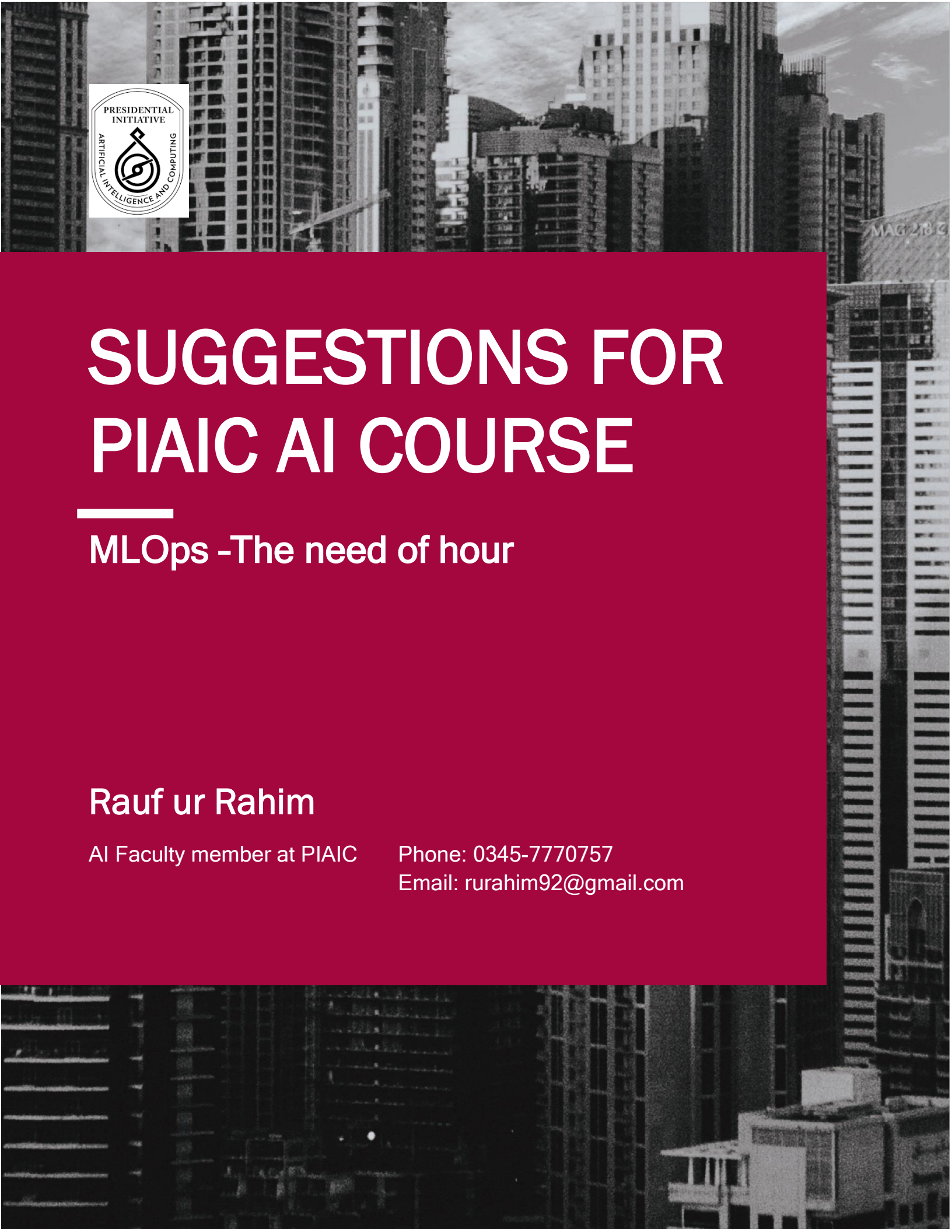


SUGGESTIONS FOR PIAIC AI COURSE

MLOps -The need of hour

Rauf ur Rahim

AI Faculty member at PIAIC

Phone: 0345-7770757
Email: rurahim92@gmail.com

Motivation

This document contains necessary suggestions for AI management related to Machine Learning pipeline and deployment. The gaps are highlighted based on industrial experience and statistics gathered through research so that necessary amendments could be made to update the course outline. This would help the students to excel more in industry.

Introduction

Artificial intelligence (AI) is expanding into standard business processes, resulting in increased revenue and reduced costs. As AI adoption grows, it becomes increasingly important for AI and machine learning (ML) practices to focus on production quality controls. Productionizing ML models introduces challenges that span organizations and processes, involving the integration of new and incumbent technologies.

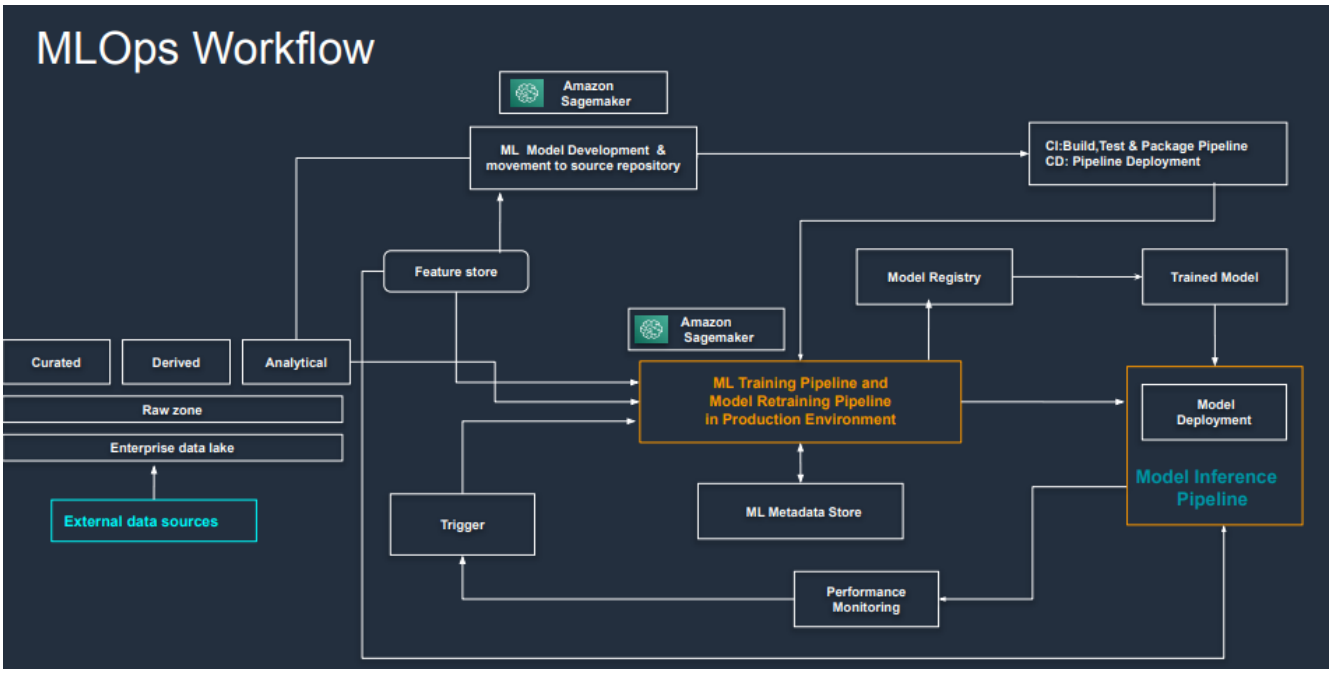
As we know that training a model based on data and tuning it to achieve optimum accuracy in one of the core and challenging work. But when we talk about the industry, unfortunately, it is considered just the 10% of whole work. The other work includes deployment strategies (canary, shadow, A/B testing), re-producing jobs, controlling data versions, tracking model performance online and offline, controlling model drift, model and data lineage, model architect for fast iteration, model degradation w.r.t time and many others.

For this purpose, I want to discuss Machine Learning Operations and pipelines that have become more prominent in 2020 and 2021 and now 2022 and onward it is not possible for industry to ignore it if they want success ML operations in their business.

"Continuous Delivery for Machine Learning (CD4ML) is a
software engineering approach in which a cross-functional
team produces machine learning applications based on code, data,
and models in small and safe increments that
can be reproduced and reliably released at any time,
in short adaptation cycles."

Explanation

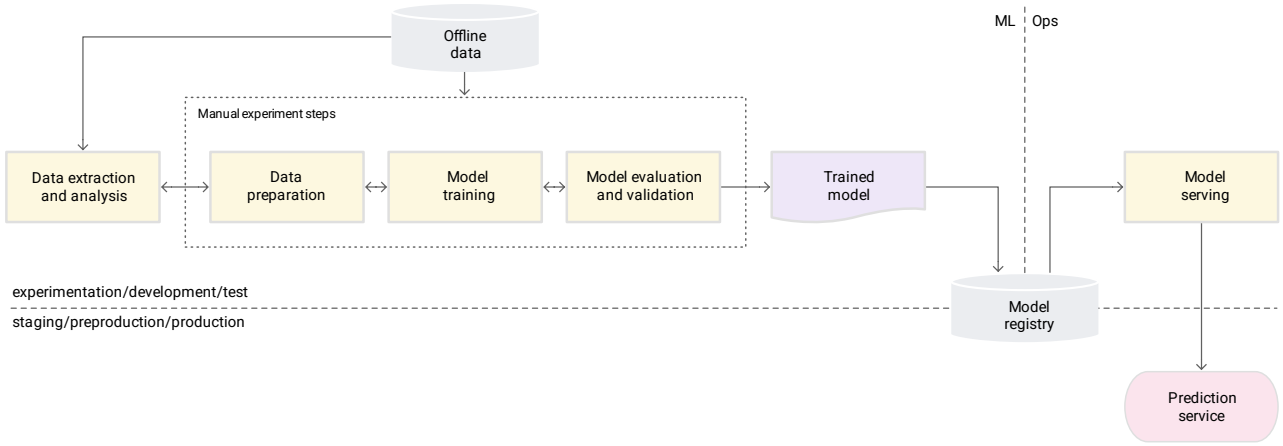
ML and DL systems are impacted by changing data profiles. This is not the case in a traditional IT system. Hence, the model has to be refreshed even if it 'works' currently – leading to more iterations in the pipeline. Hence, you have to monitor models in production and refresh the model by retraining if the model performance falls below a certain criterion (continuous training). The below AWS image depicts some concepts related to reproducing work in offline and online both modes, continuous training with triggering signal, different data version and hyper-parameters tuning etc.



Similarly, Google [9] divides the style of AI productization into three categories

Maturity Level 0: Manual Process

Many teams have data scientists and ML researchers who can build state-of-the-art models, but their process for building and deploying ML models is entirely manual. This is considered the *basic* level of maturity, or level 0. The following diagram shows the workflow of this process.

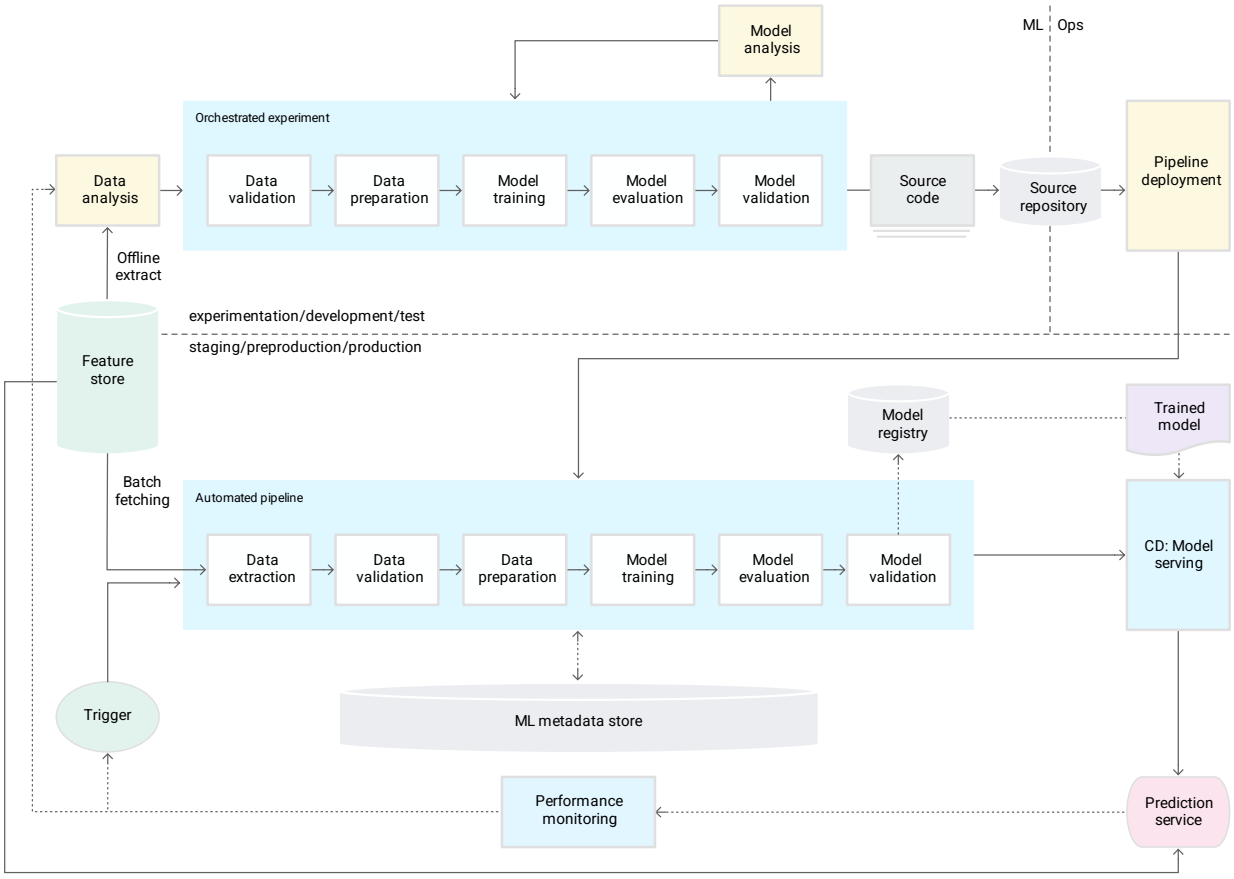


Challenges in this level

MLOps level 0 is common in many businesses that are beginning to apply ML to their use cases. This manual, data-scientist-driven process might be sufficient when models are rarely changed or trained. In practice, models often break when they are deployed in the real world. The models fail to adapt to changes in the dynamics of the environment, or changes in the data that describes the environment

Maturity Level 1: ML Pipeline Automation

The goal of level 1 is to perform continuous training of the model by automating the ML pipeline; this lets you achieve continuous delivery of model prediction service. To automate the process of using new data to retrain models in production, you need to introduce automated data and model validation steps to the pipeline, as well as pipeline triggers and metadata management.
The following figure is a schematic representation of an automated ML pipeline for CT.



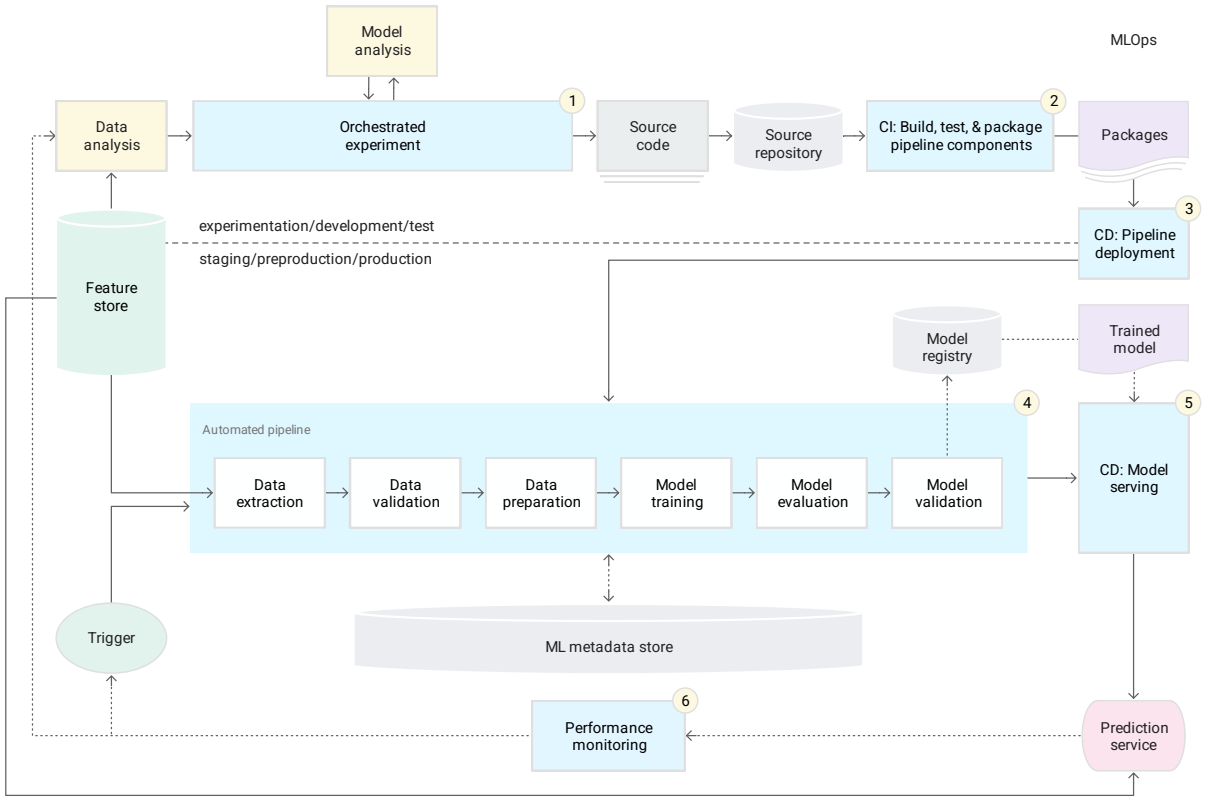
Challenges in this level

Assuming that new implementations of the pipeline aren't frequently deployed and you are managing only a few pipelines, you usually manually test the pipeline and its components. In addition, you manually deploy new pipeline implementations. You also submit the tested source code for the pipeline to the IT team to deploy to the target environment. This setup is suitable when you deploy new models based on new data, rather than based on new ML ideas. However, you need to try new ML ideas and rapidly deploy new implementations of the ML components. If you manage many ML pipelines in production, you need a CI/CD setup to automate the build, test, and deployment of ML pipelines.

Maturity Level 2: CI/CD pipeline automation

For a rapid and reliable update of the pipelines in production, you need a robust automated CI/CD system. This automated CI/CD system lets your data scientists rapidly explore new ideas around feature engineering, model architecture, and hyperparameters. They can implement these ideas and

automatically build, test, and deploy the new pipeline components to the target environment.
The following diagram shows the implementation of the ML pipeline using CI/CD, which has the characteristics of the automated ML pipelines setup plus the automated CI/CD routines.



PIAIC Q4 Course Outline

Now let's look at the PIAIC's current Q4 outline.

1. Introduction to Amazon SageMaker (Week 1)
Chapter 1 of Learn Amazon SageMaker

2. Handling Data Preparation Techniques in the Cloud (Week 2)
Chapter 2 of Learn Amazon SageMaker

3. AutoML with Amazon SageMaker Autopilot (Week 3)
Chapter 3 of Learn Amazon SageMaker

4. Training Machine Learning Models (Week 4)
Chapter 4 of Learn Amazon SageMaker

5. Training Natural Language Processing Models (Week 5)
Chapter 6 of Learn Amazon SageMaker

6. Training Computer Vision Models (Week 6)
Chapter 5 of Learn Amazon SageMaker

7. Extending Machine Learning Services Using Built-In TensorFlow Framework (Week 7)
Chapter 7 Pages 216-238 of Learn Amazon SageMaker

8. Object Detection Models using TensorFlow and SageMaker (Week 8)
Chapter 5 of Hands-On Computer Vision with TensorFlow 2

9. Enhancing and Segmenting Images using TensorFlow and SageMaker (Week 9)
Chapter 6 of Hands-On Computer Vision with TensorFlow 2

10. Training on Complex and Scarce Datasets using TensorFlow and SageMaker (Week 10)
Chapter 7 of Hands-On Computer Vision with TensorFlow 2

11. Video and Recurrent Neural Networks using TensorFlow and SageMaker (Week 10)
Chapter 8 of Hands-On Computer Vision with TensorFlow 2

12. Scaling Your Training Jobs in the Cloud (Week 11)
Chapter 9 of Learn Amazon SageMaker

13. Deploying Machine Learning Models (Week 12)
Chapter 11 of Learn Amazon SageMaker

14. Automating Machine Learning Workflows using CDK and AWS Step Functions (Week 13)
Chapter 12 Pages 414-435 of Learn Amazon SageMakerLabeling

I believe, the PIAIC Q4 unnecessarily focuses on Computer Vision, NLP, Model training and data handling with respect to SageMaker. Most of these core concepts have already been covered in Q2 and Q3 and now there is no as such need to teach again in SageMaker. Instead, there are a lot more important concepts to teach the students that is the AI productization because latest study says almost 87% small to top 100 fortune businesses are not able to properly manage the ML workflows in production.

The questions to arise

By keeping in view, the above outline of PIAIC Q4 the questions arise that to which extend PIAIC student is able:
to work on this whole automation pipeline (CI/CD/CT) [8]
to test different deployment strategies [3], [4]
to run pipelines on auto when a data drift is detected [1]
to monitor the real time ML model performance [6]
to handle the Data Versioning Control (DVC) [5]
to build MLOps workflow with sagemaker, github / gitlab integration [2]
to handle model and data lineage in ML experimentation [7]

Material / Book to Follow

https://docs.aws.amazon.com/sagemaker/latest/dg/sagemaker-dg.pdf
https://www.amazon.com/Practical-MLOps-Operationalizing-Machine-Learning/dp/1098103017

and some additional files are attached with this document to take a look.

References

[1] https://aws.amazon.com/blogs/machine-learning/automate-model-retraining-with-amazon-sagemaker-pipelines-when-drift-is-detected/

[2] https://aws.amazon.com/blogs/machine-learning/build-mlops-workflows-with-amazon-sagemaker-projects-gitlab-and-gitlab-pipelines/

[3] https://aws.amazon.com/blogs/machine-learning/take-advantage-of-advanced-deployment-strategies-using-amazon-sagemaker-deployment-guardrails/

[4] https://aws.amazon.com/blogs/architecture/architecting-for-machine-learning/

[5] https://dvc.org/

[6] https://cml.dev/

[7] https://aws.amazon.com/blogs/machine-learning/model-and-data-lineage-in-machine-learning-experimentation/

[8] https://aws.amazon.com/blogs/machine-learning/building-automating-managing-and-scaling-ml-workflows-using-amazon-sagemaker-pipelines/

[9] https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning