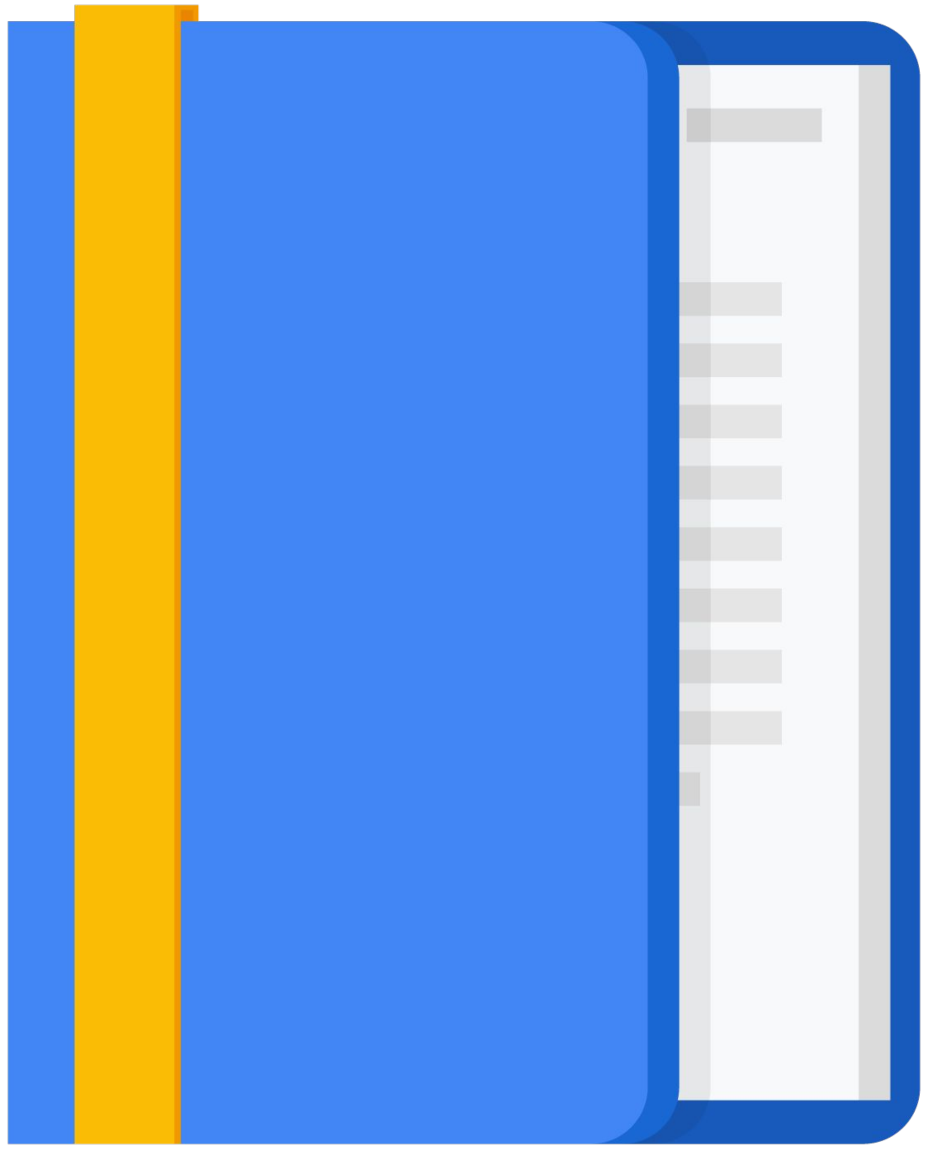
Why and When to Employ MLOps

Agenda

Data Scientists’ Pain Points

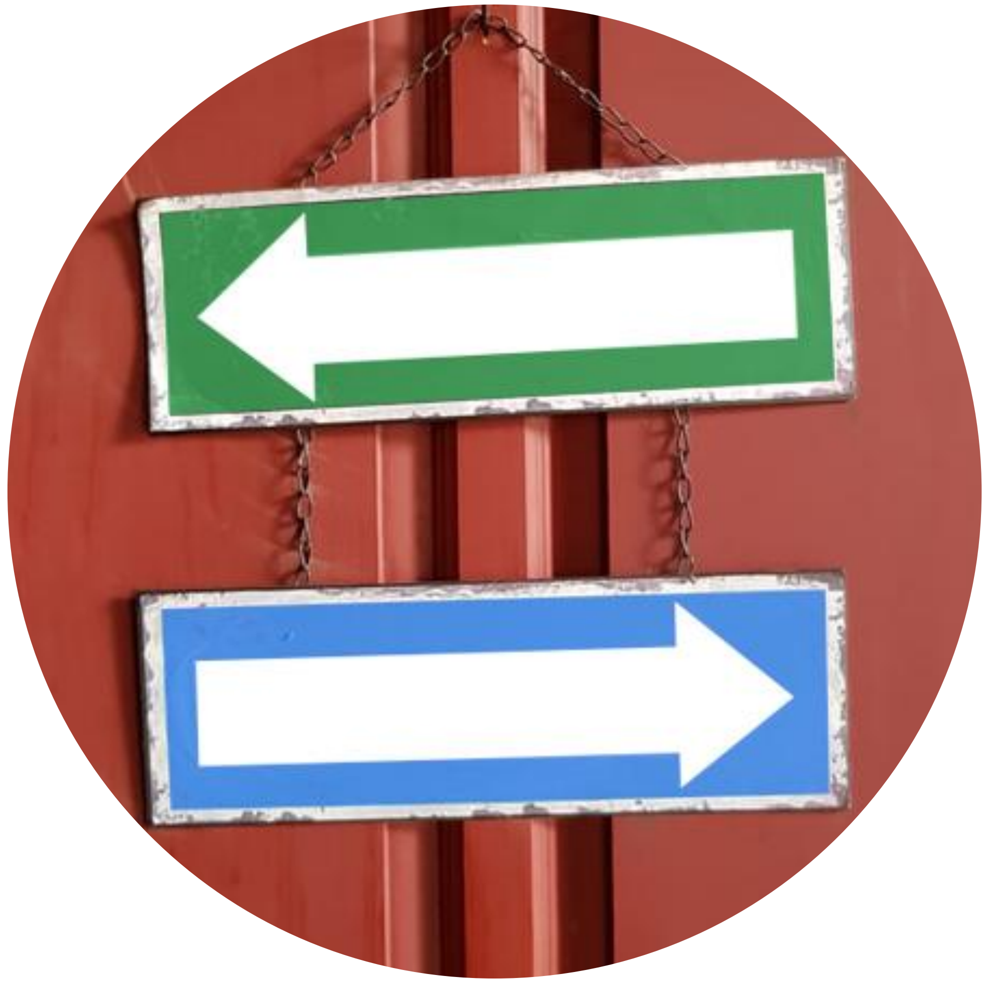
The concept of DevOps in ML

Machine Learning Lifecycle



It’s hard to track of …

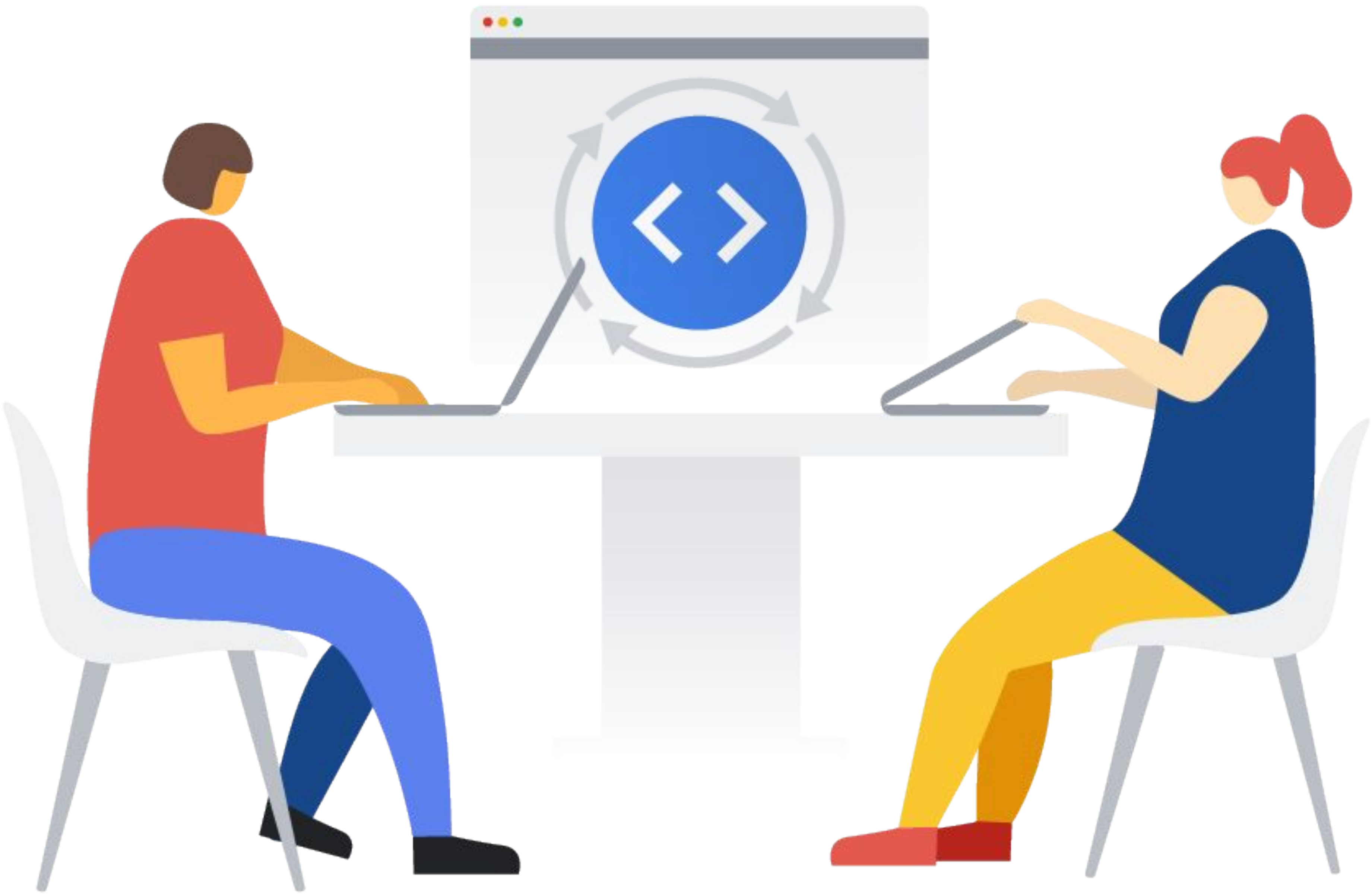
Model Versions



Metrics &
Pre-processing

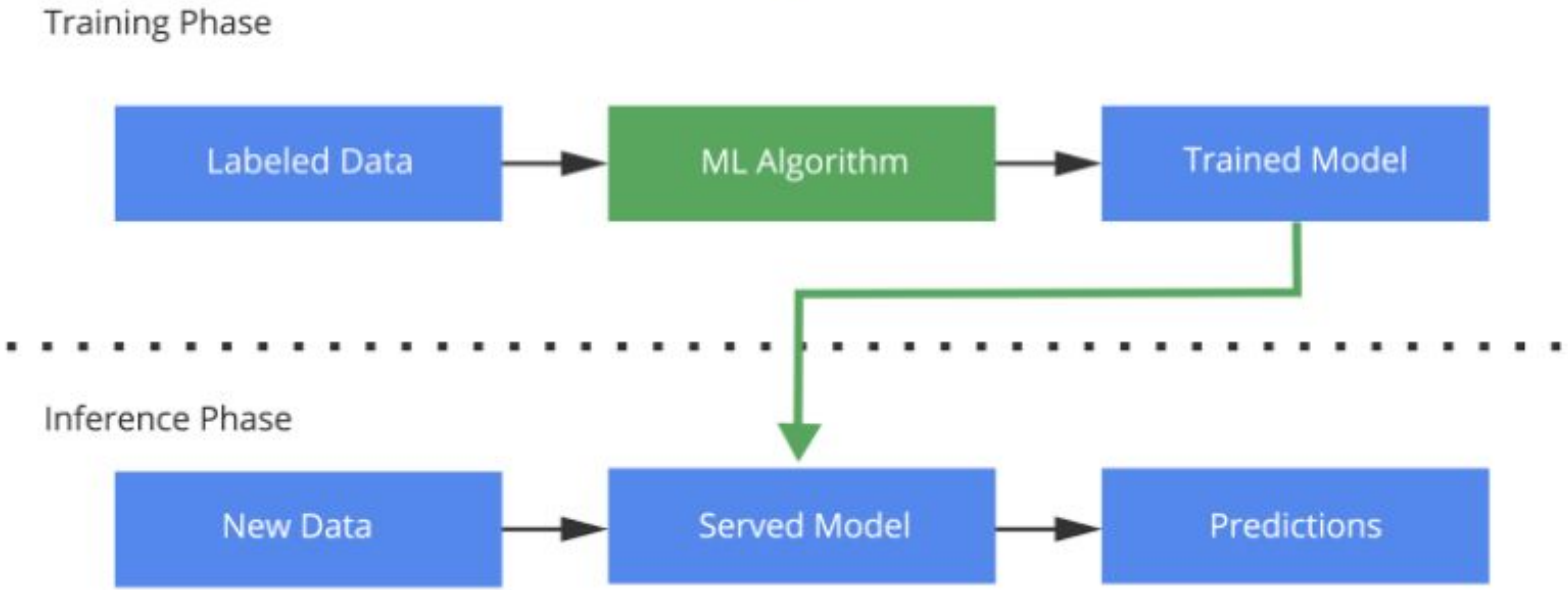
Best Hyper-Parameters

It’s hard to share models & reproduce results...



...deploy in production

The root of the problem: 2 phases = 2 environments

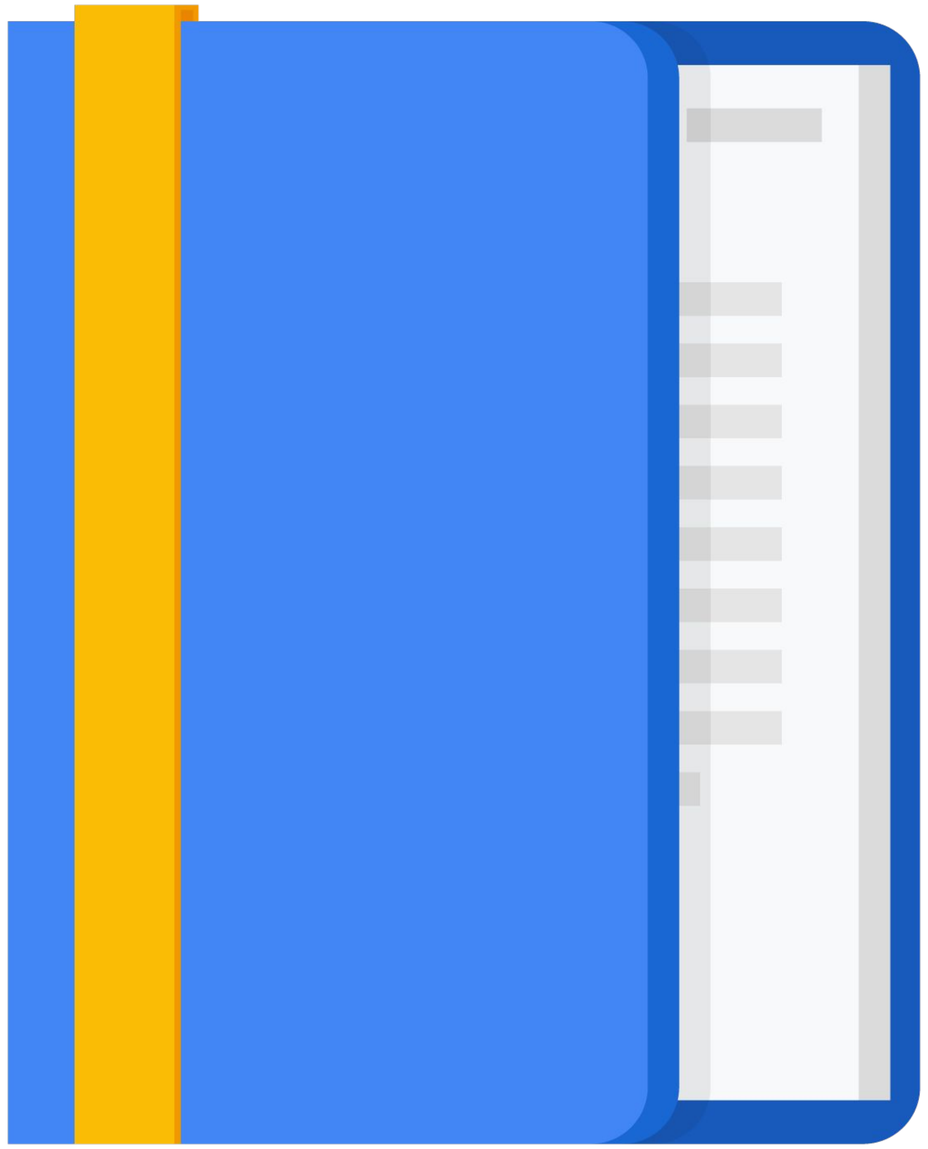


Agenda

Data Scientists' Pain Points

The concept of DevOps in ML

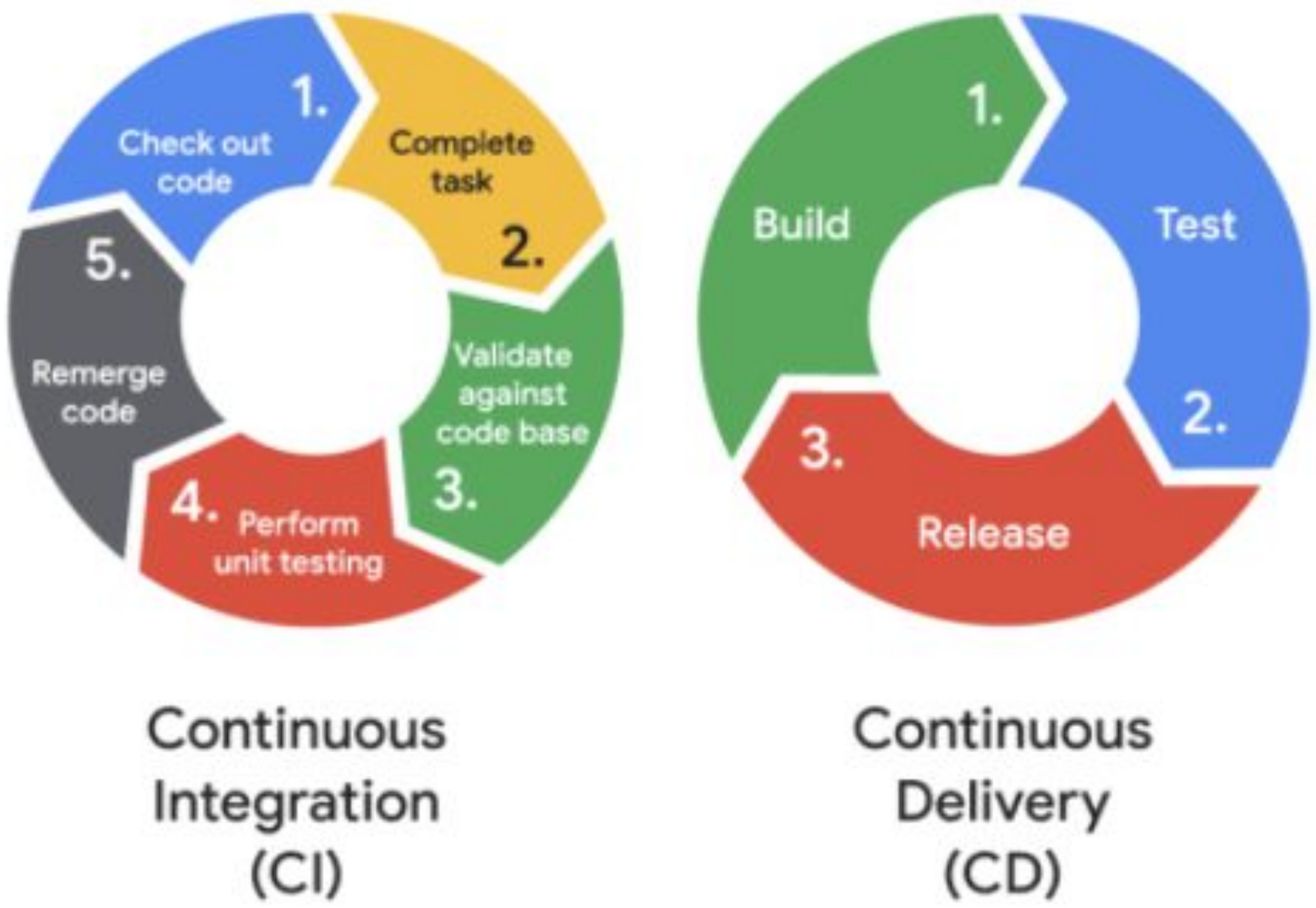
Machine Learning Lifecycle



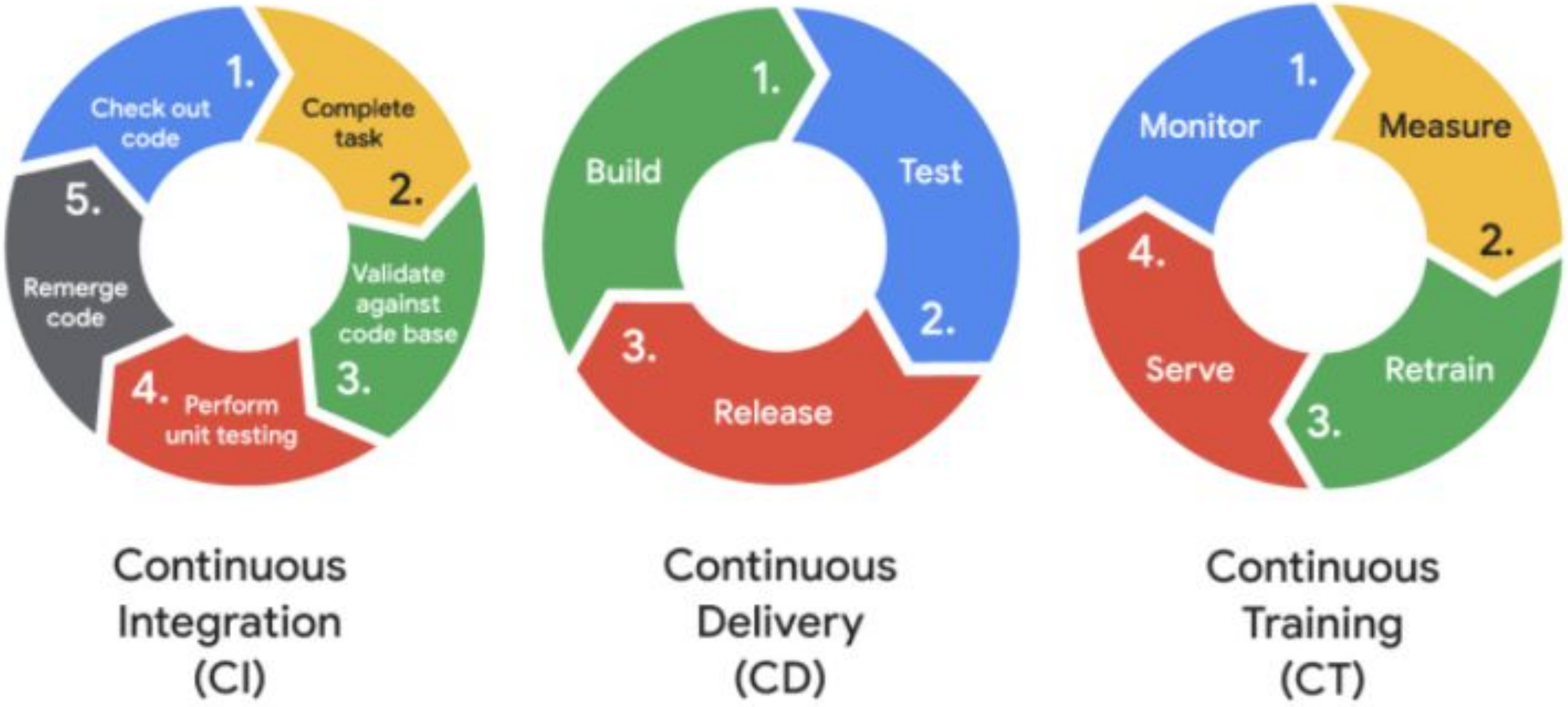
MLOps is a lifecycle management discipline for machine learning

Some DevOps concepts translate directly to MLOps

Some DevOps concepts translate directly to MLOps



Some DevOps concepts translate directly to MLOps



But MLOps differs from DevOps in important ways

	DevOps	MLOps
1	Test and validate code and components	Also test and validate data, data schemas, and models

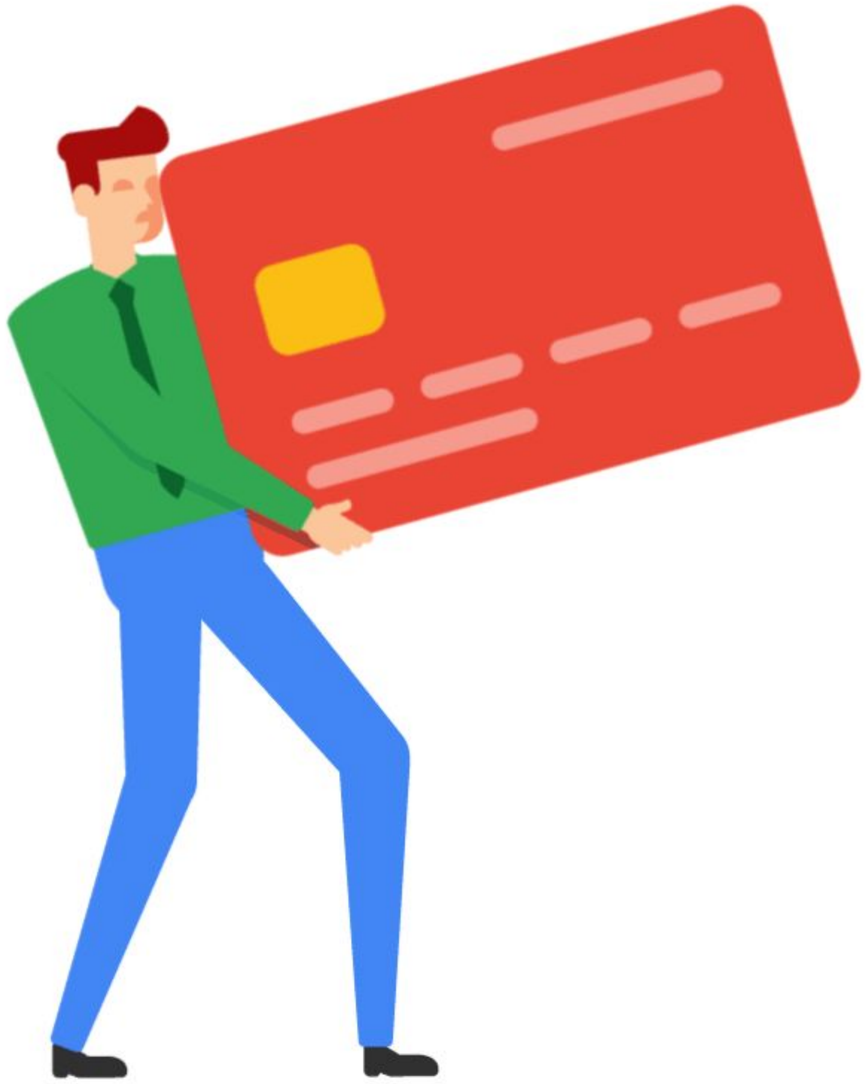
But MLOps differs from DevOps in important ways

	DevOps	MLOps
1	Test and validate code and components	Also test and validate data, data schemas, and models
2	Focus on a single software package or service	Also consider the whole system: the ML training pipeline

But MLOps differs from DevOps in important ways

	DevOps	MLOps
1	Test and validate code and components	Also test and validate data, data schemas, and models
2	Focus on a single software package or service	Also consider the whole system: the ML training pipeline
3	Deploy code and move to the next task	Constantly monitor, retrain, and serve the model

Machine learning is the high-interest credit card of technical debt



Credit Card Bill

Total to Pay	255 days

ML systems can easily build up technical debt

Multi-functional teams

ML systems can easily build up technical debt

Multi-functional teams

Experimental nature

ML systems can easily build up technical debt

Multi-functional teams

Experimental nature

Testing complexity

ML systems can easily build up technical debt

Multi-functional teams

Experimental nature

Testing complexity

Deployment complexity

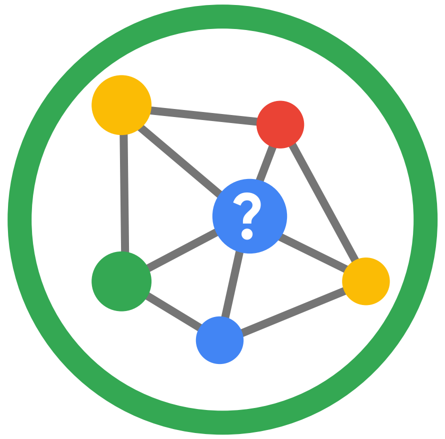
ML systems can easily build up technical debt

Multi-functional teams

Experimental nature

Testing complexity

Deployment complexity



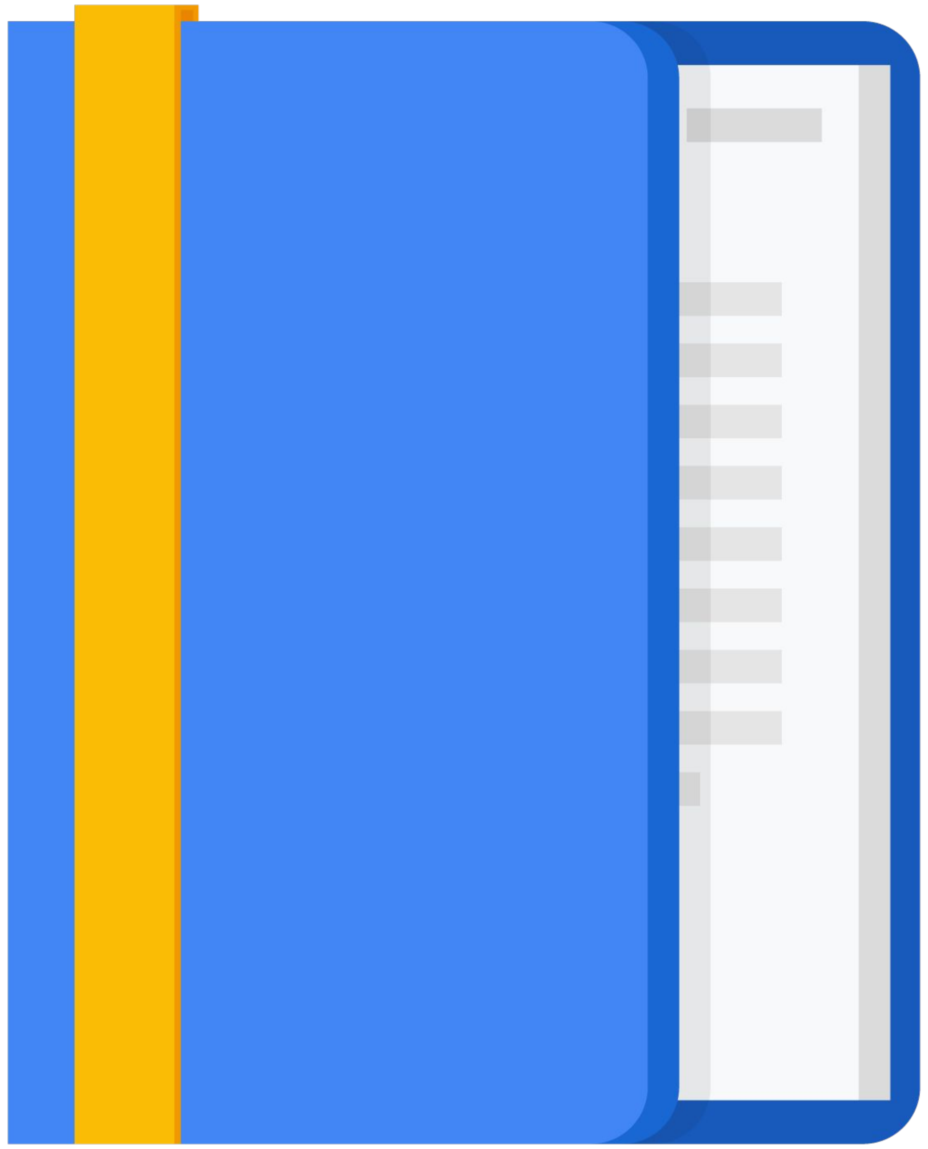
Model decay

Agenda

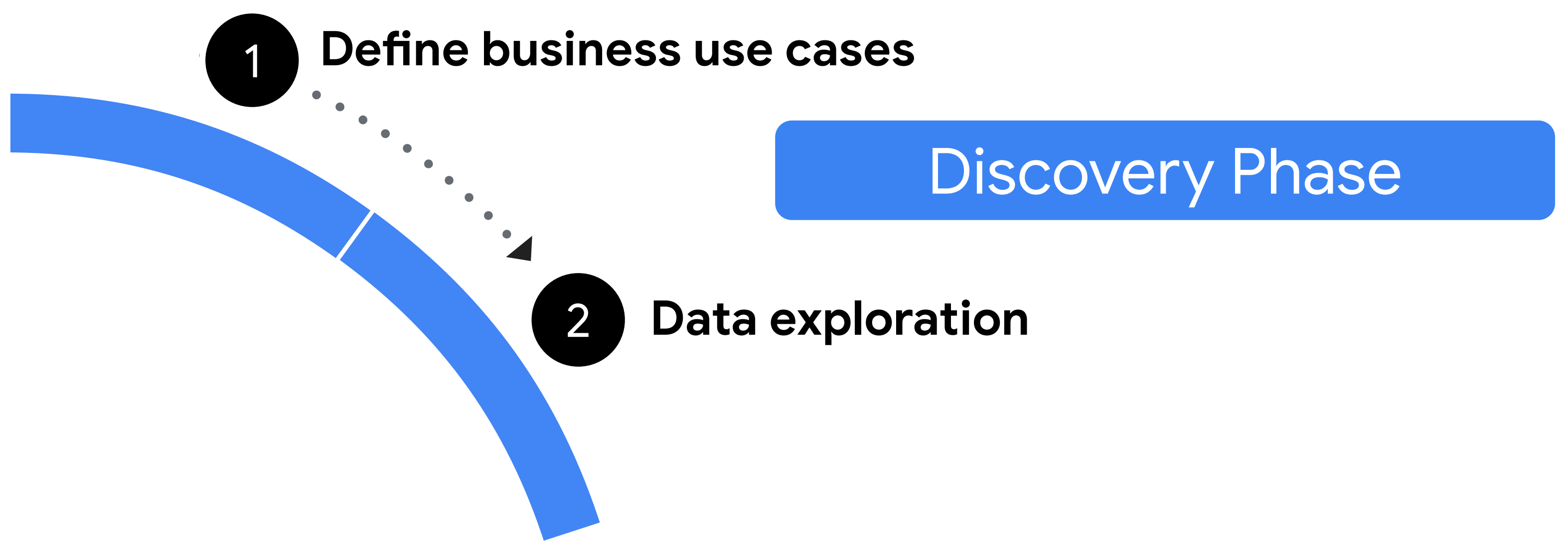
Data Scientists' Pain Points

The concept of DevOps in ML

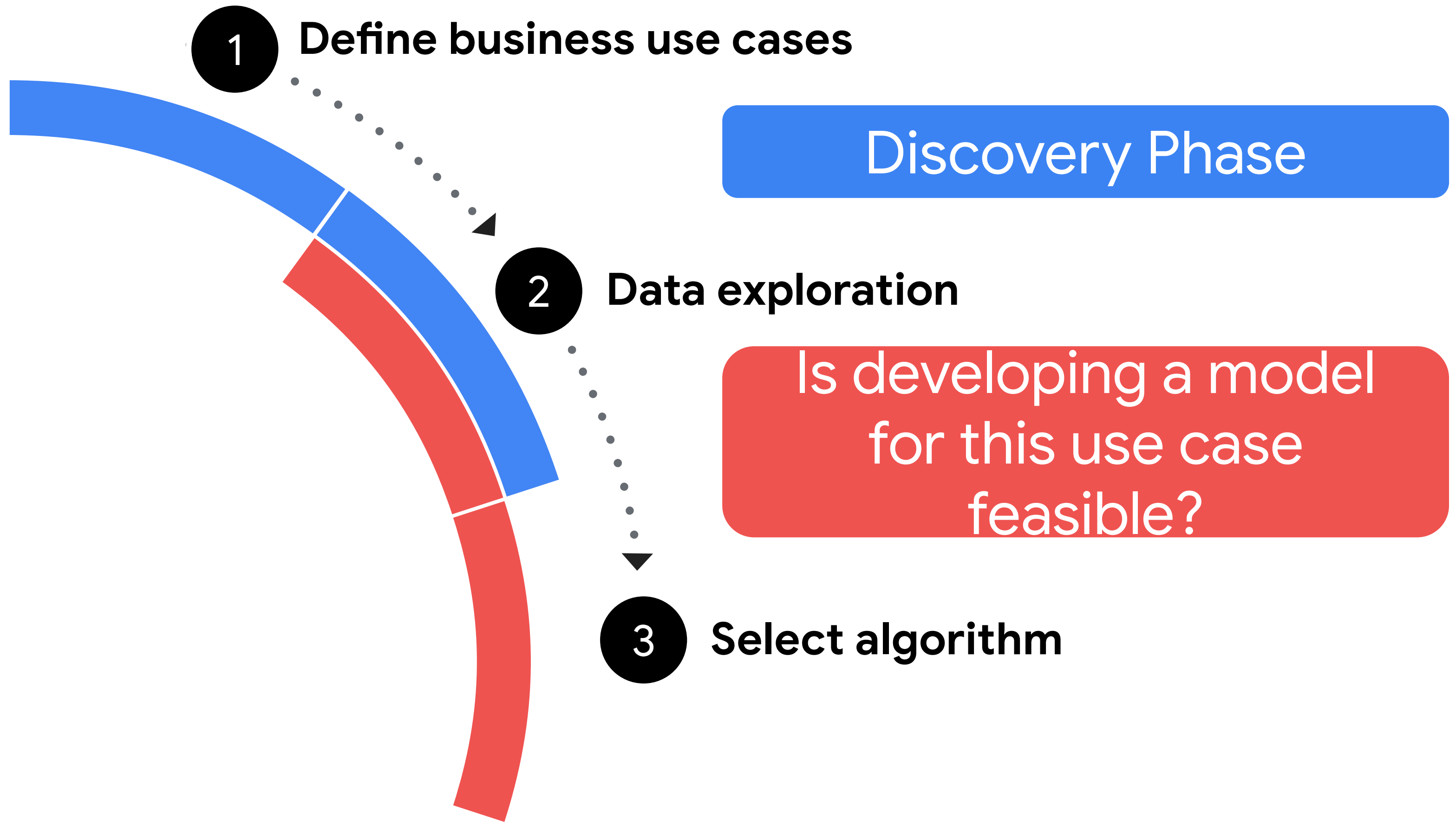
Machine Learning Lifecycle



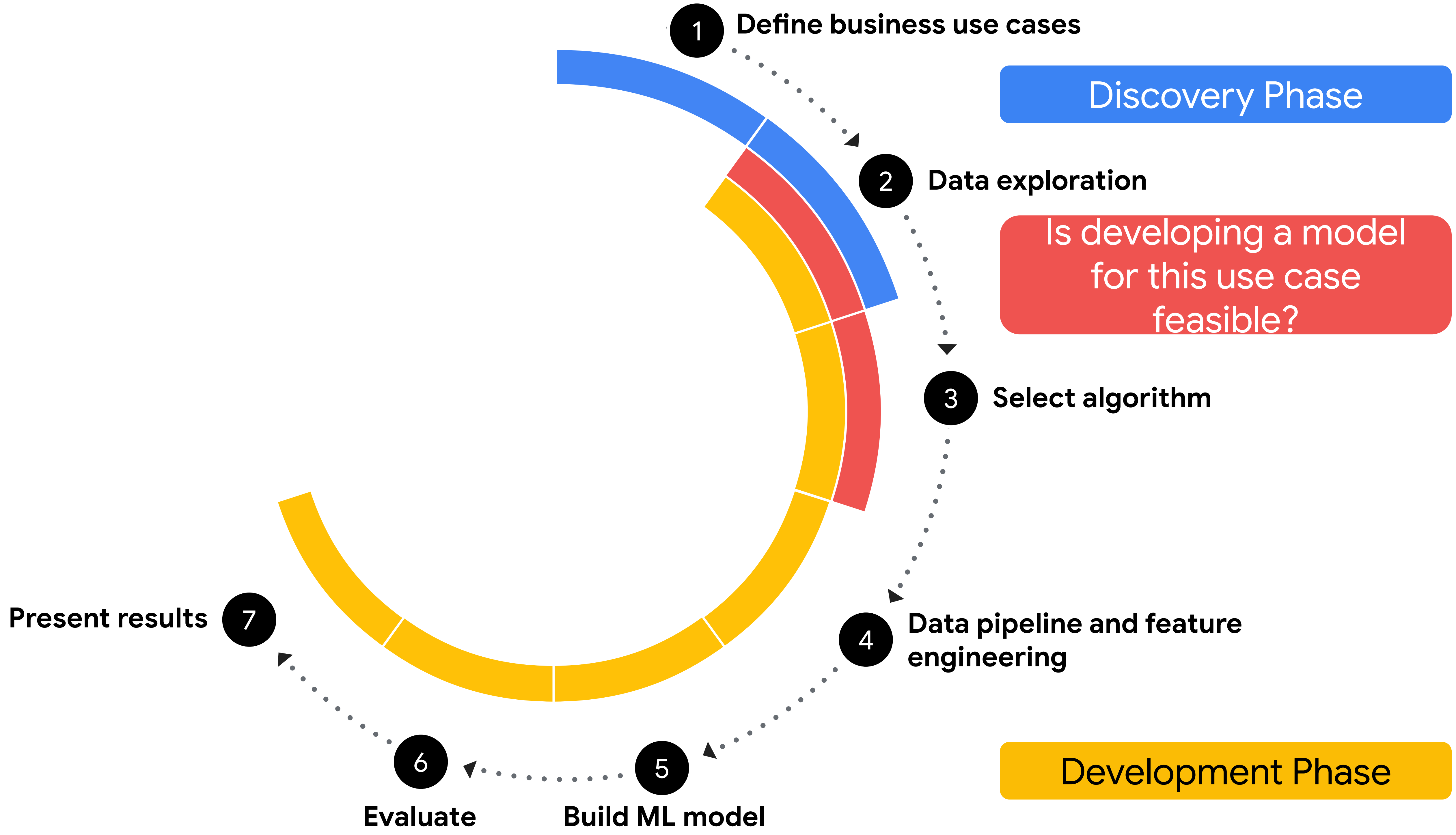
Phases of a machine learning project



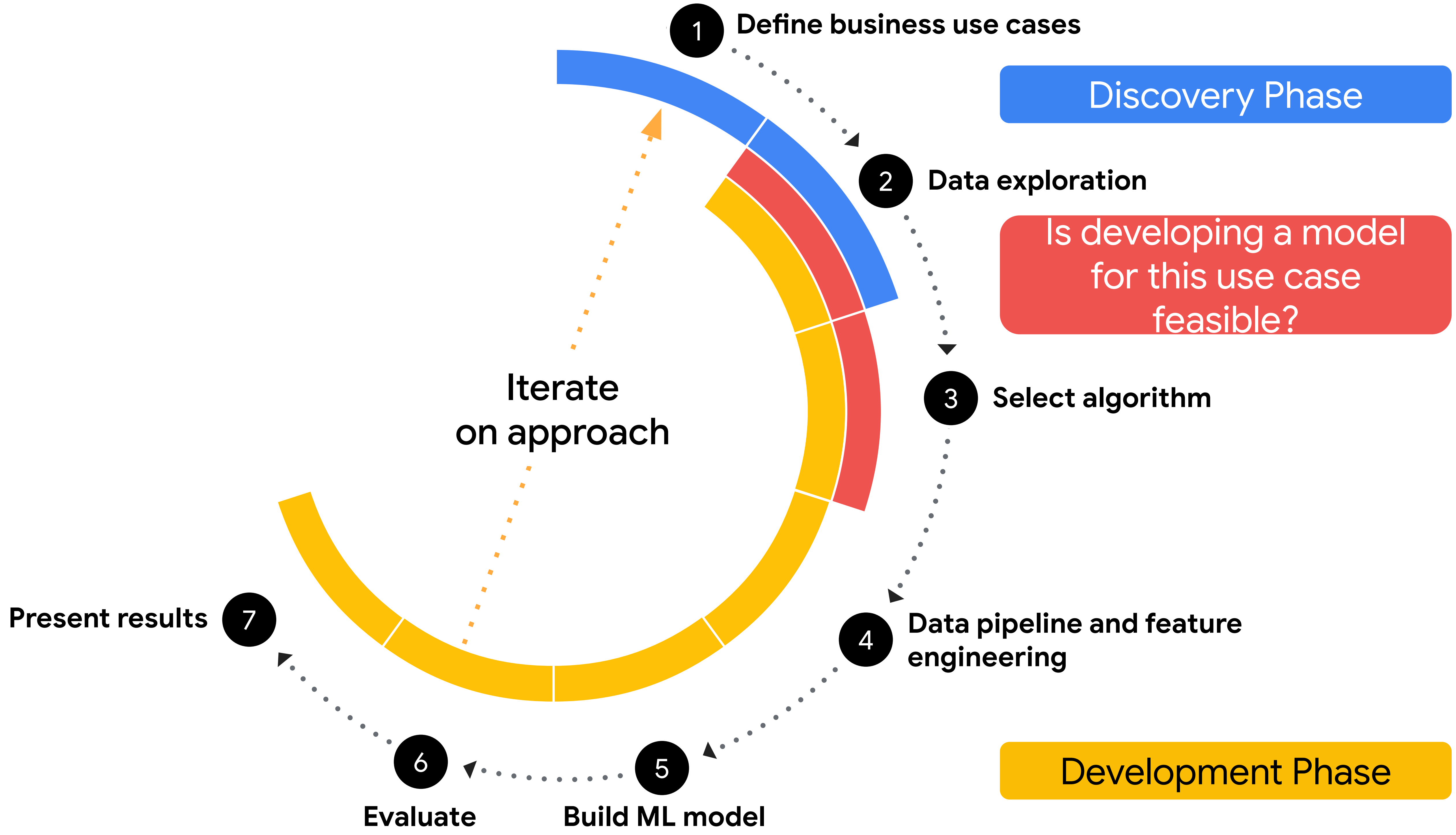
Phases of a machine learning project



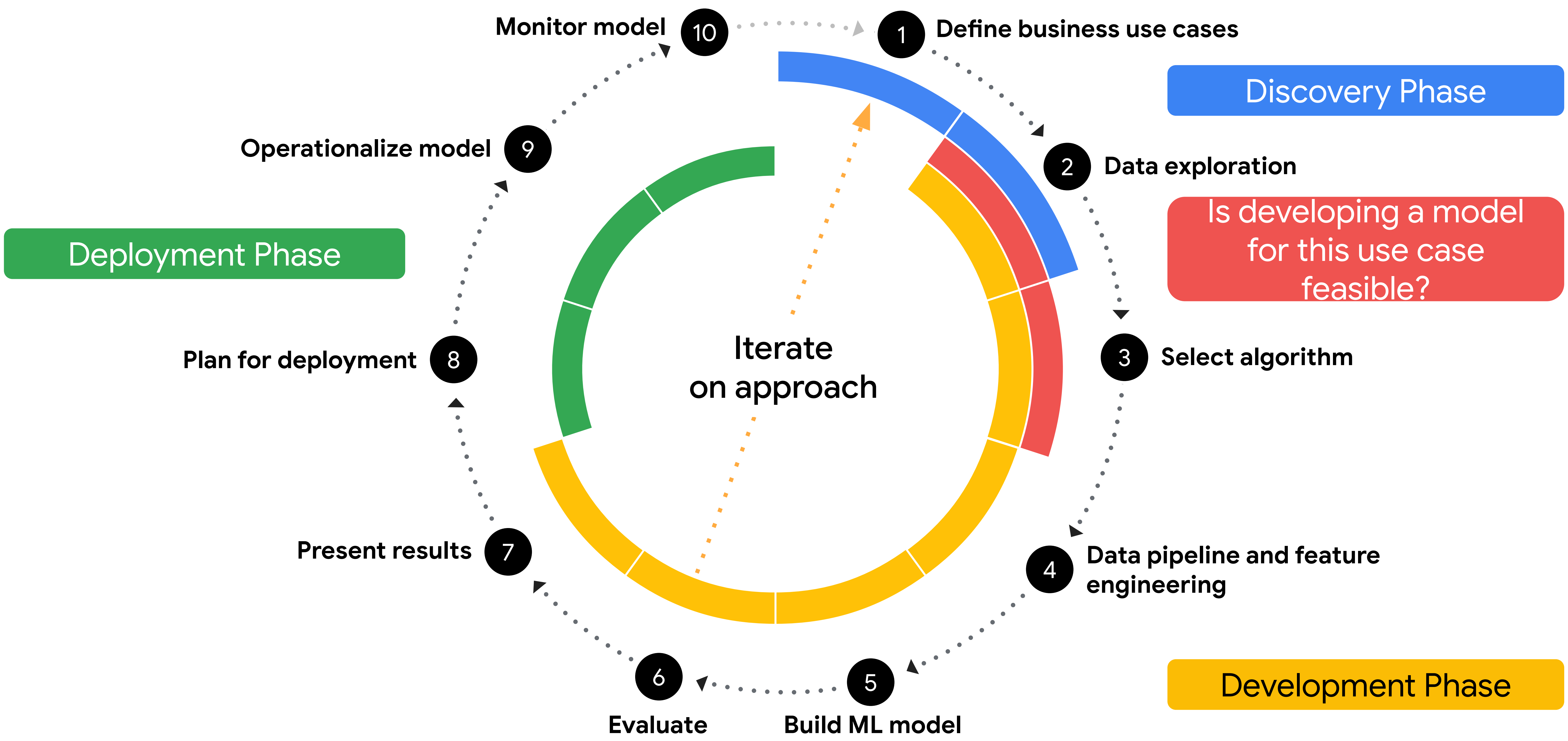
Phases of a machine learning project



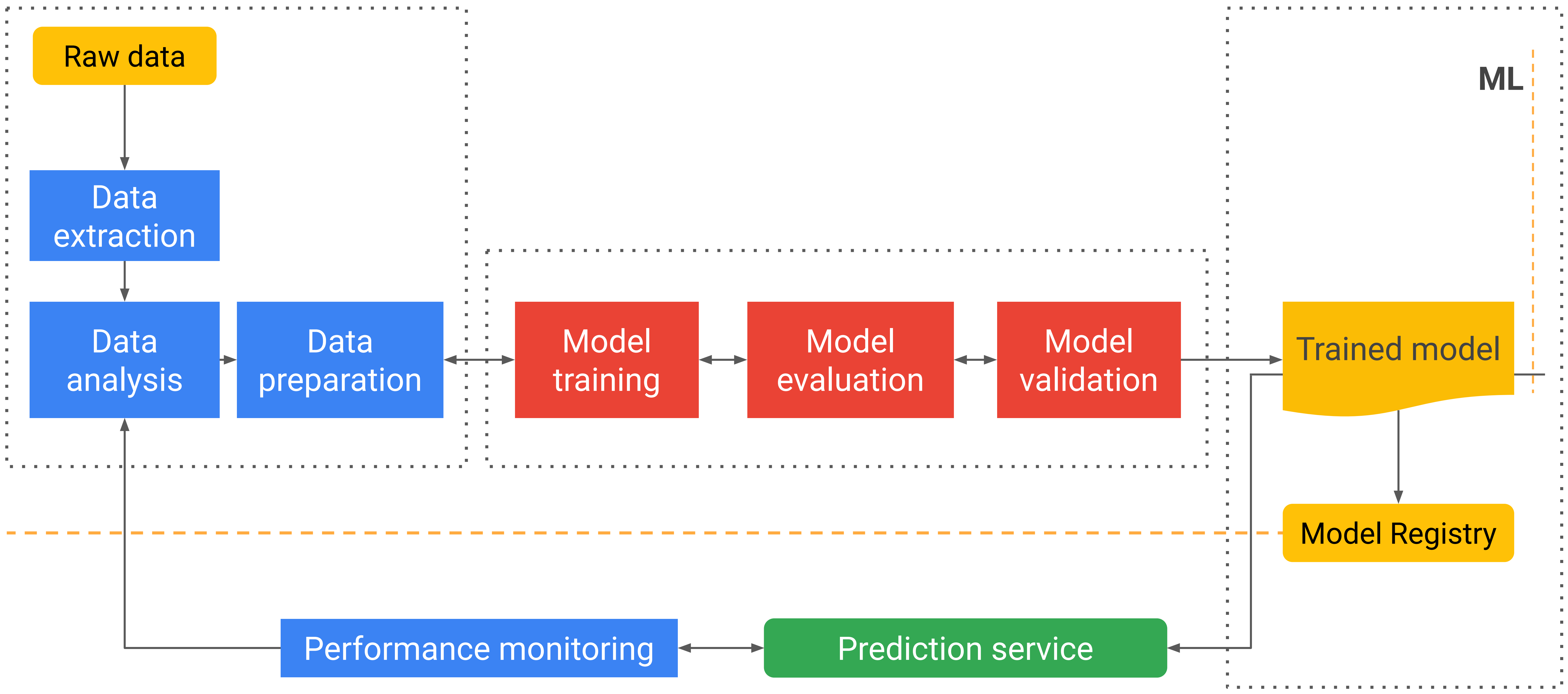
Phases of a machine learning project



Phases of a machine learning project



An ML pipeline contains well-defined processes



The level of automation defines the maturity of the ML process

Level 0	Level 1	Level 2
Build and deploy manually (e.g., local training in JuypyterLab, scp the trained model into custom serving node)	Containerized training and Cloud serving (e.g., Vertex Training and Vertex Prediction)	Automate the whole process into ML pipeline from data prep to model serving (e.g., Vertex Pipeline)

cloud.google.com