riskified tech;

Multiple ML Models For Multiple Clients

Steps For Scaling Up

March 2022

Agenda

01 Riskified

02 Single Model vs Multi Models Tradeoff

03 The ML Pipeline Solution

04 Riskified's Trainer-as-a-Service : From Theory To Practice

About me

Ori Peri

- ML Engineer @Riskified
- MSc in SW & Information Systems Engineering @BGU
- BSc in Computer Science @BGU

ori.peri@riskified.com

Riskified by the Numbers

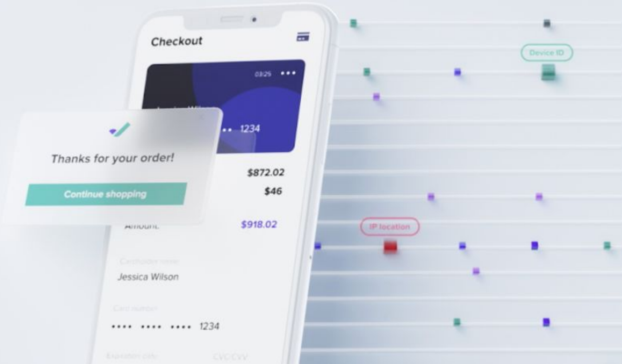
750+
Global team, nearly 50% in **engineering & analytics**

180+
Countries across the globe

50+
Publicly held companies among our clients

$89B
Online volume (GMV) reviewed in 2021

98%+
Client retention* for the past 2 years



ticketmaster

macy's

PRADA

wish

wayfair

lastminute.com

REVOLVE

FINISH LINE

*Annual dollar retention

As of March 2022

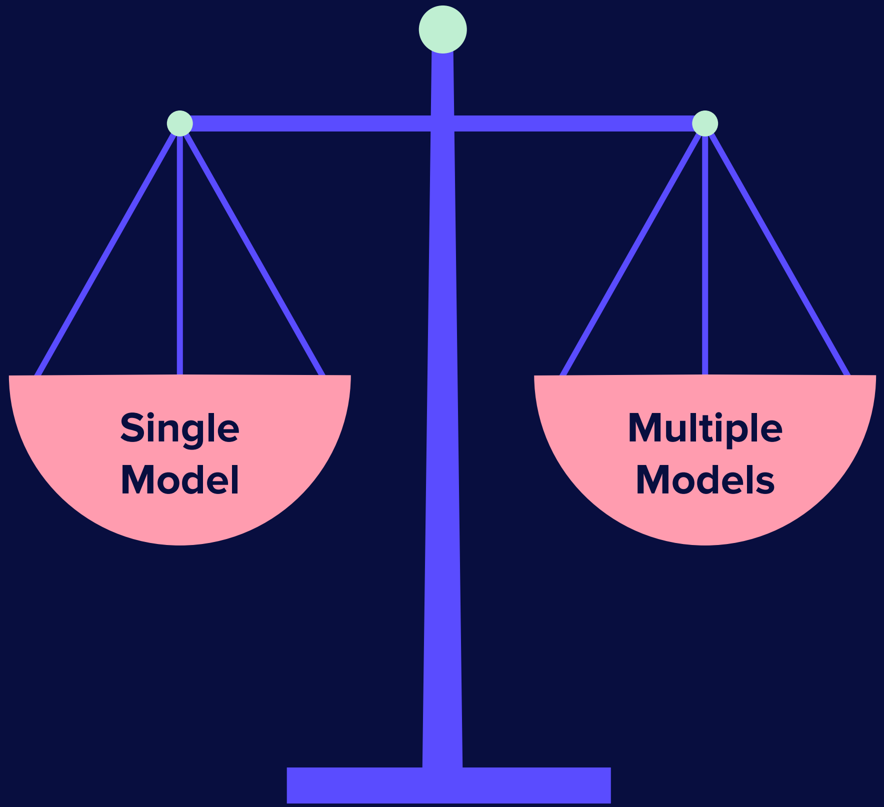
Training in Riskified

14 Days Manual Pipeline

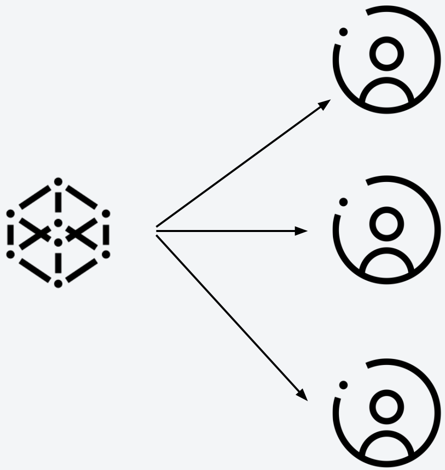
25+ Production Models

>1 Year To Retrain All Models

The Tradeoff



Single Model



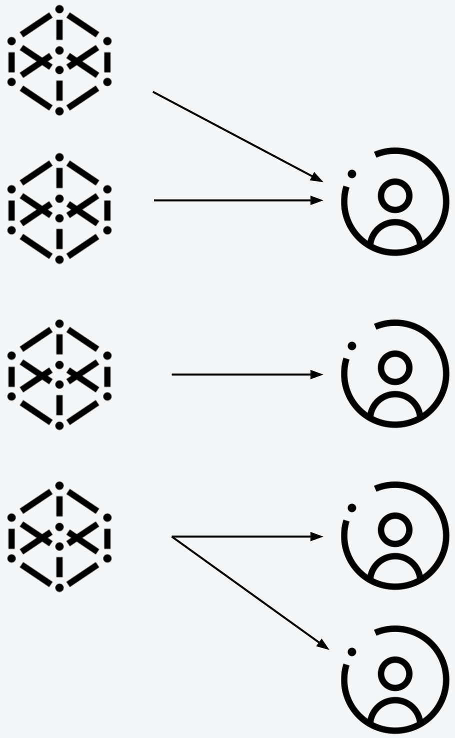
Pros:

- Small engineering effort
- Larger dataset
- Cold start handling

Cons:

- Lower performance
- Differences in data distribution
- Monitoring different KPIs

Multiple Models



Pros:

- Better performance
- Fitting customers' KPI

Cons:

- Costs
- Cold start
- Small customers
- Heavy engineering effort

The ML Pipeline Solution

MLOps Levels

Quick Overview

Level 0 - No MLOps

- Manual training, validation
- 0 tracking of training and performance

Level 1 - Automated Training & Deployment

- Auto ML pipeline
- Centralized tracking
- Easy to reproduce

Level 2 - Full MLOps

- Rapid exploration of features, models
- CI/CD/CT

The ML Pipeline's BBs

Data Validation

Data Preparation

Model Training

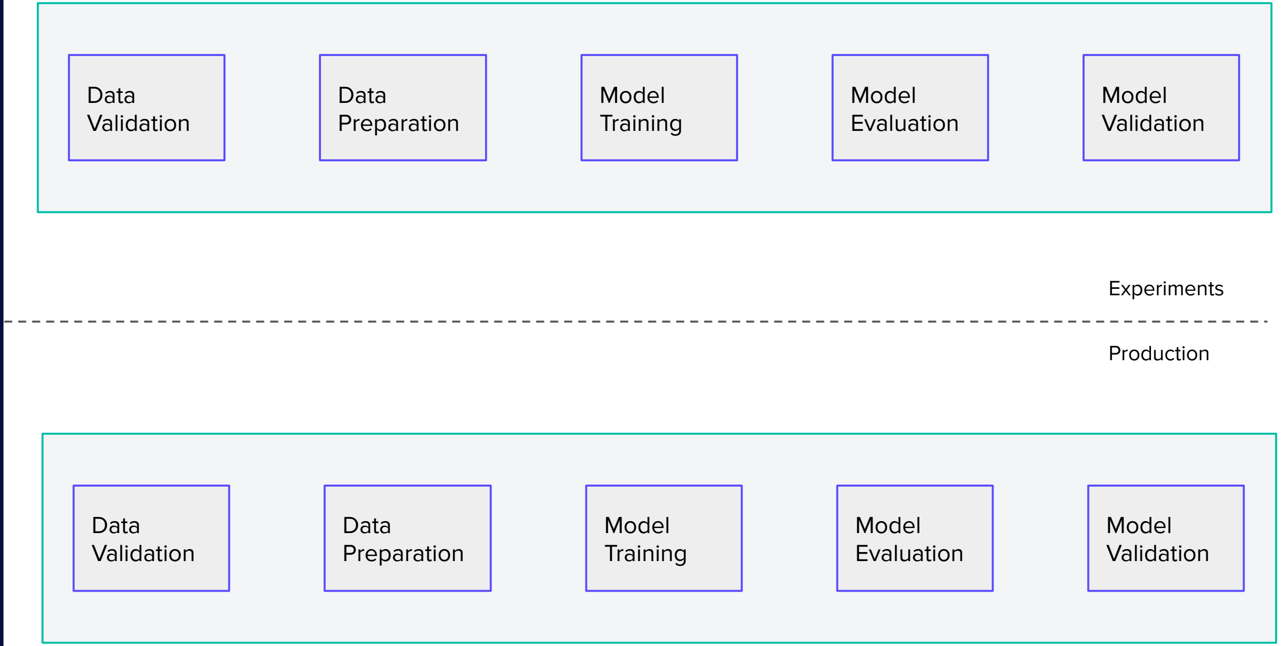
Model Evaluation

Model Validation

- Containerised
  & Composable BBs

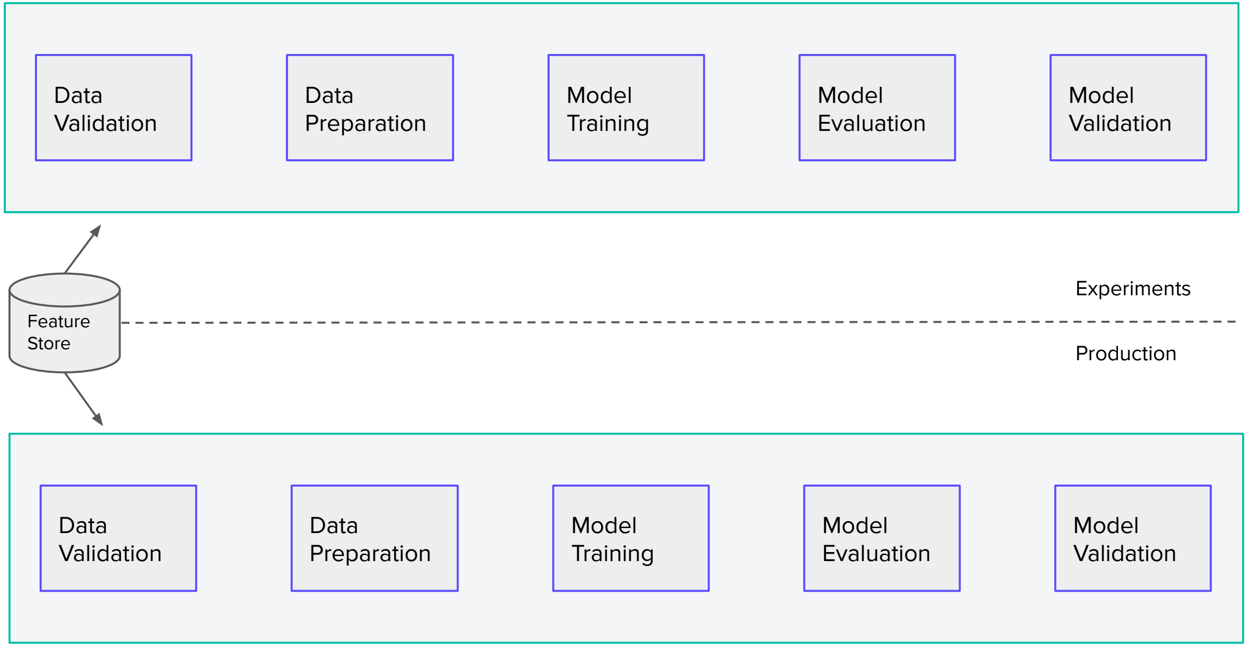
The ML Pipeline

- Same pipeline is used for research (offline) and production (online)



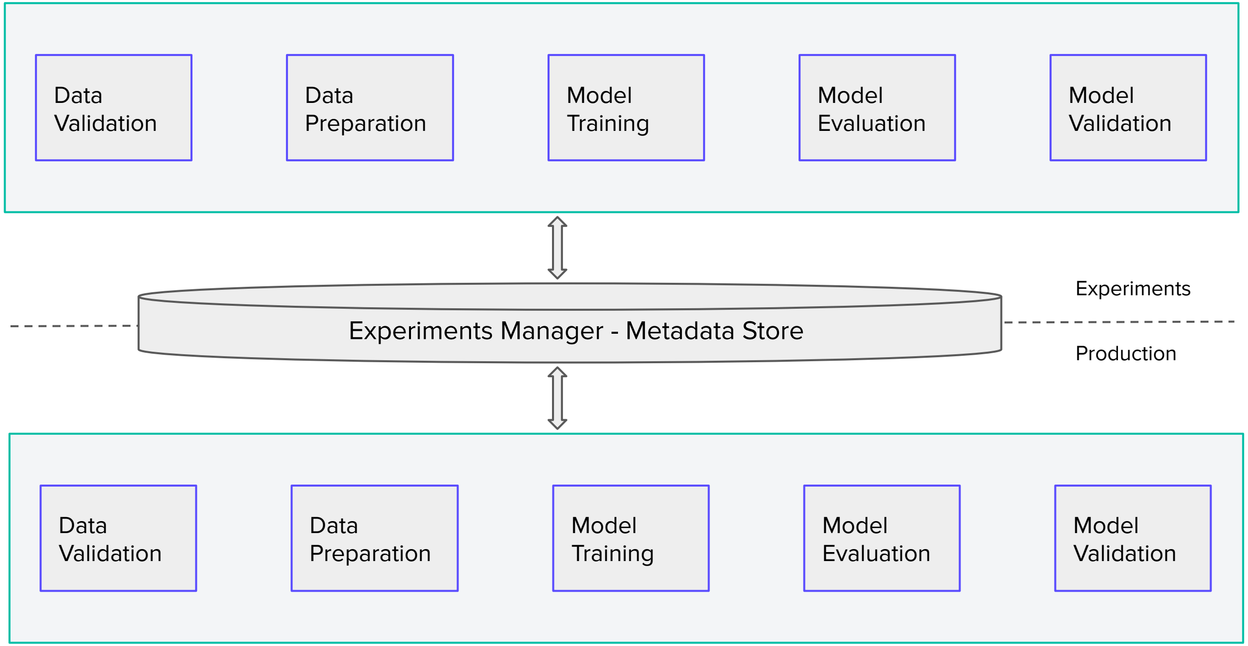
Features Store

- Features definition standardization
- Offline & Online consistency



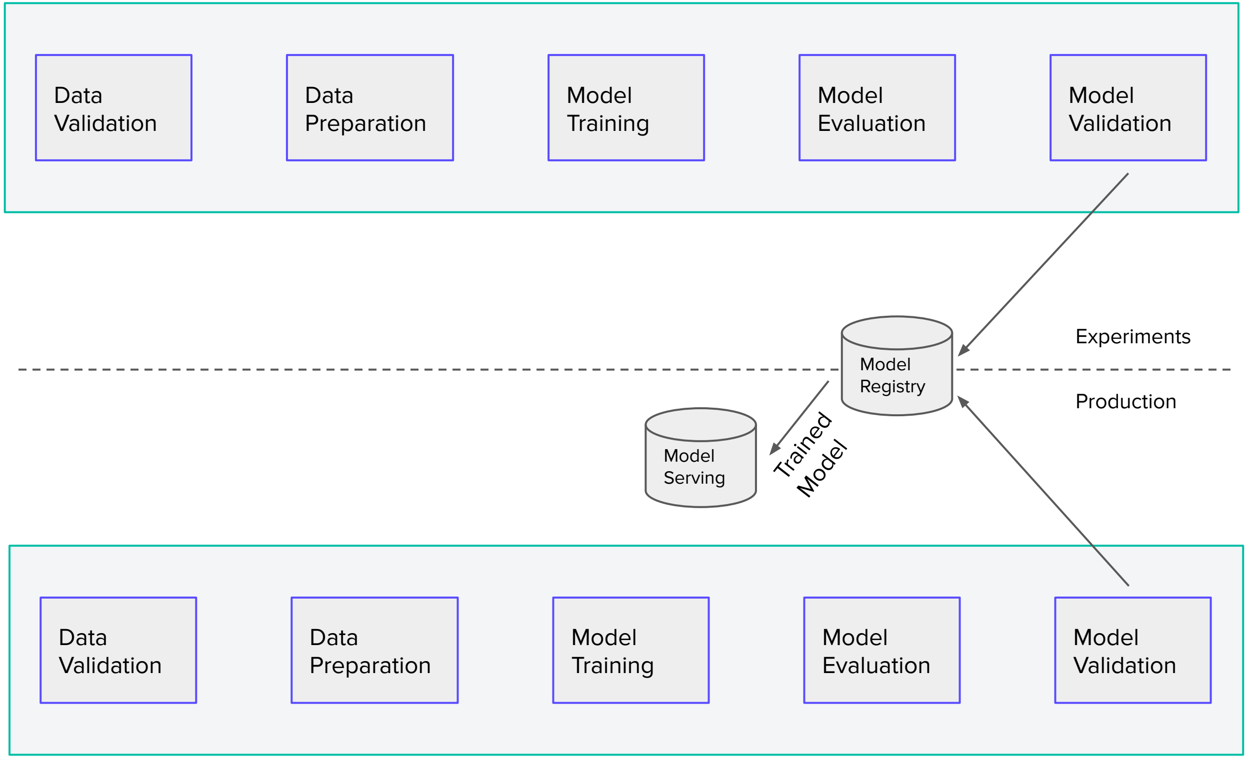
Experiments Manager

- Record execution info & artifacts
- Visibility of training & model plots
- Enable easy comparisons



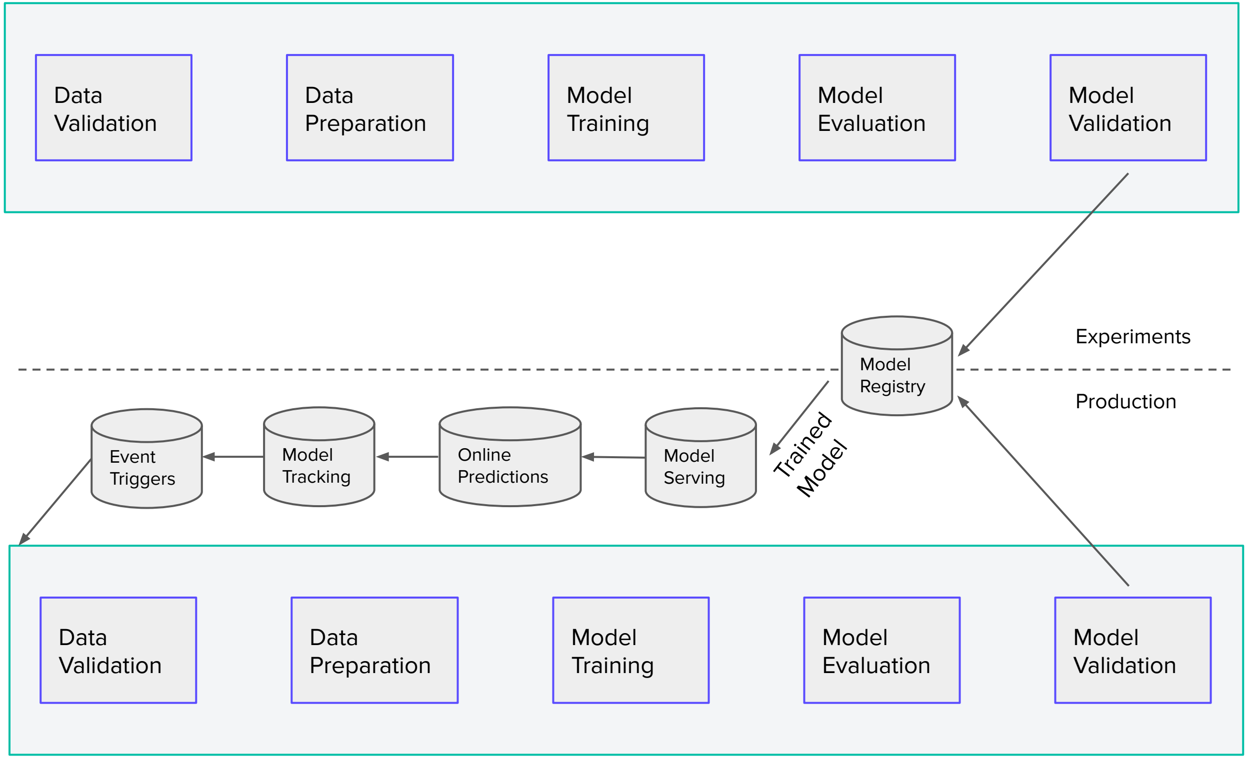
Model Registry

- Manage customer's SOTA models & versions
- Retrain & online serving handoff



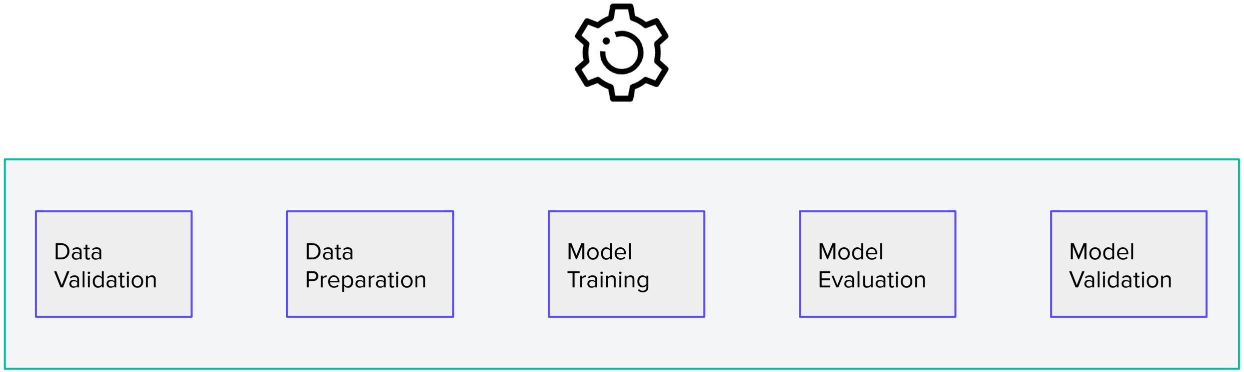
Event Triggers

- Data drift - features distributions
- Model performance degradation
- New data - trigger after X amount of new data/time passed



Workflow Orchestrator

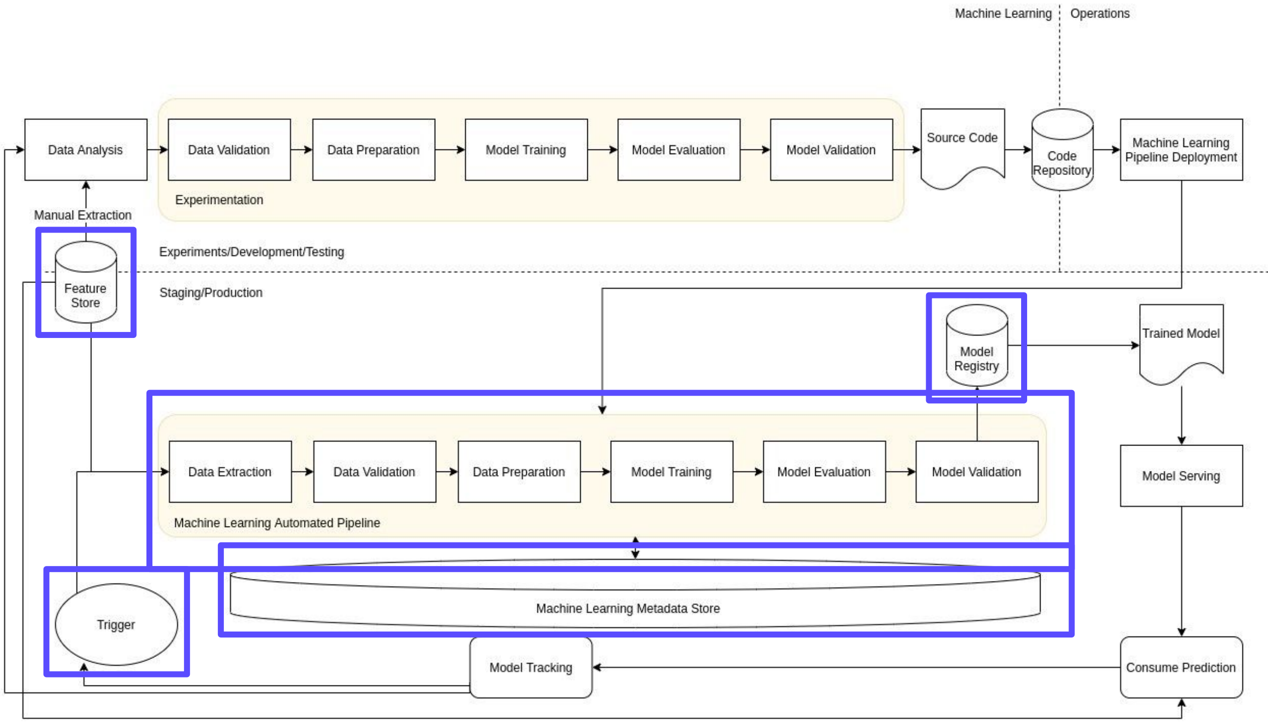
- Reproducibility
- Debug & rerun failures
- Execution info - dataset, features, code version, etc.



The ML Pipeline - All Together

Model Life Cycle:

- Model Monitoring -> Retrain Decision -> Data Preparation -> Train -> Inference -> Validation -> Deploy



Source - https://towardsdatascience.com/mlops-level-1-continuous-training

Riskified's

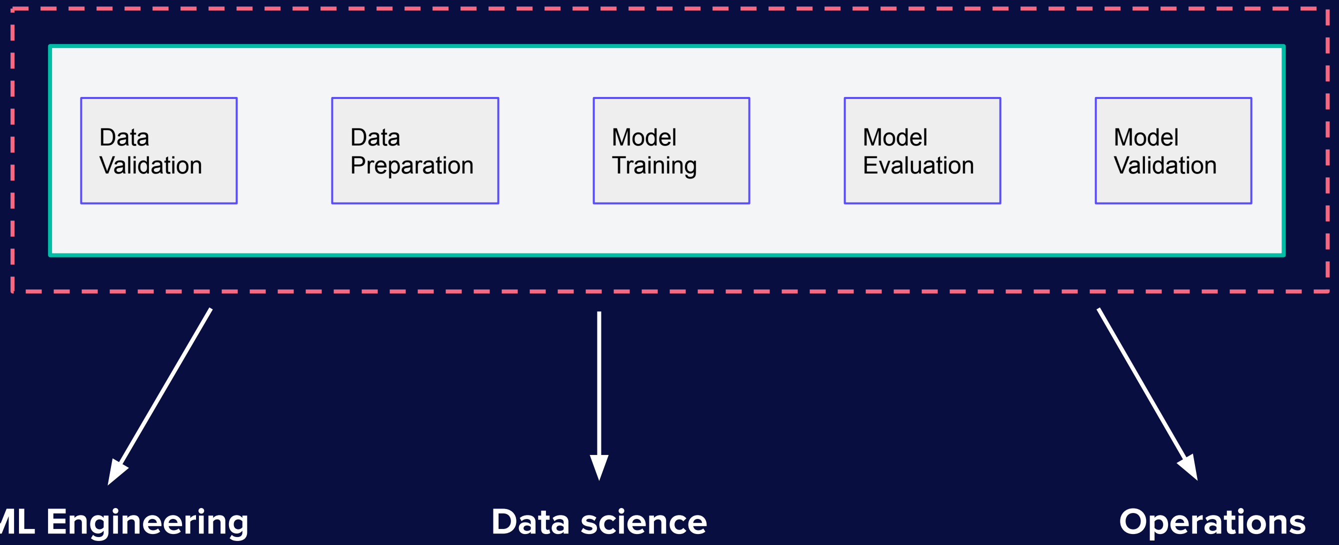
Trainer-as-a-Service :

From Theory To Practice

Operations

Riskified’s Trainer-as-a-Service

Goals ➔ Difficulties ➔ Solutions



Riskified's Trainer-as-a-Service

Goals

01 Support Multiple Code Languages

02 Reduce Researchers SW Effort

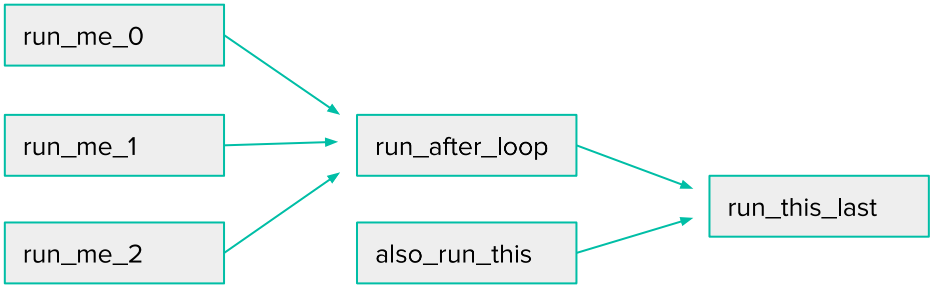
03 On-Demand BB Replacement

04 Allow Parallel Development On The Same Pipeline

Riskified’s Trainer-as-a-Service

Pipeline Orchestrator:

- Airflow workflow is a DAG above k8s consists of multiple tasks



Riskified's Trainer-as-a-Service

Goals

01 Support Multiple Code Languages

02 Reduce Researchers SW Effort

03 On-Demand BB Replacement

04 Allow Parallel Development On The Same Pipeline

Riskified’s Trainer-as-a-Service

- Each task is a containerized code running within a given Docker image
- Enable R/Python/Scala components

run_me_0

Riskified's Trainer-as-a-Service

Goals

01 Support Multiple Code Languages

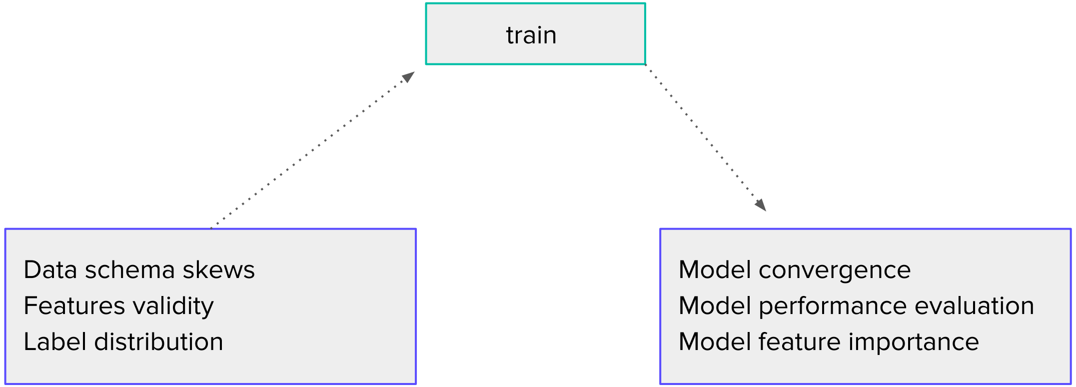
02 Reduce Researchers SW Effort

03 On-Demand BB Replacement

04 Allow Parallel Development On The Same Pipeline

Riskified's Trainer-as-a-Service

- Tasks are testable with a defined API
- Analytical + SW tests in pipeline
- Reduce dev effort for data scientists



Riskified's Trainer-as-a-Service

Goals

01 Support Multiple Code Languages

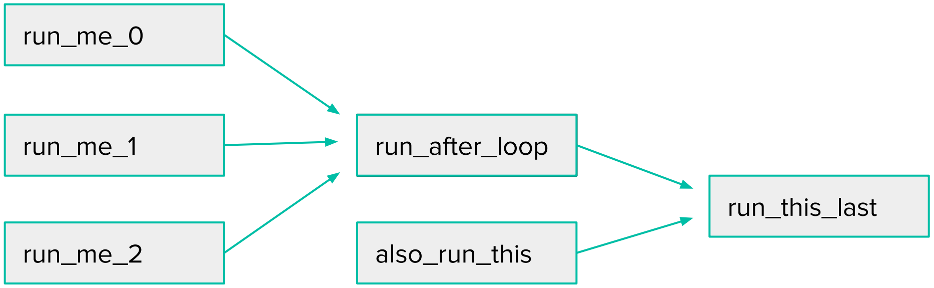
02 Reduce Researchers SW Effort

03 On-Demand BB Replacement

04 Allow Parallel Development On The Same Pipeline

Riskified's Trainer-as-a-Service

- Framework for easy insertion of new code components into pipeline
- Facilitate images replacement, integration and testing



Riskified’s Trainer-as-a-Service

Goals

01 Support Multiple Code Languages

02 Reduce Researchers SW Effort

03 On-Demand BB Replacement

04 Allow Parallel Development On The Same Pipeline

Riskified's Trainer-as-a-Service

The Goal:

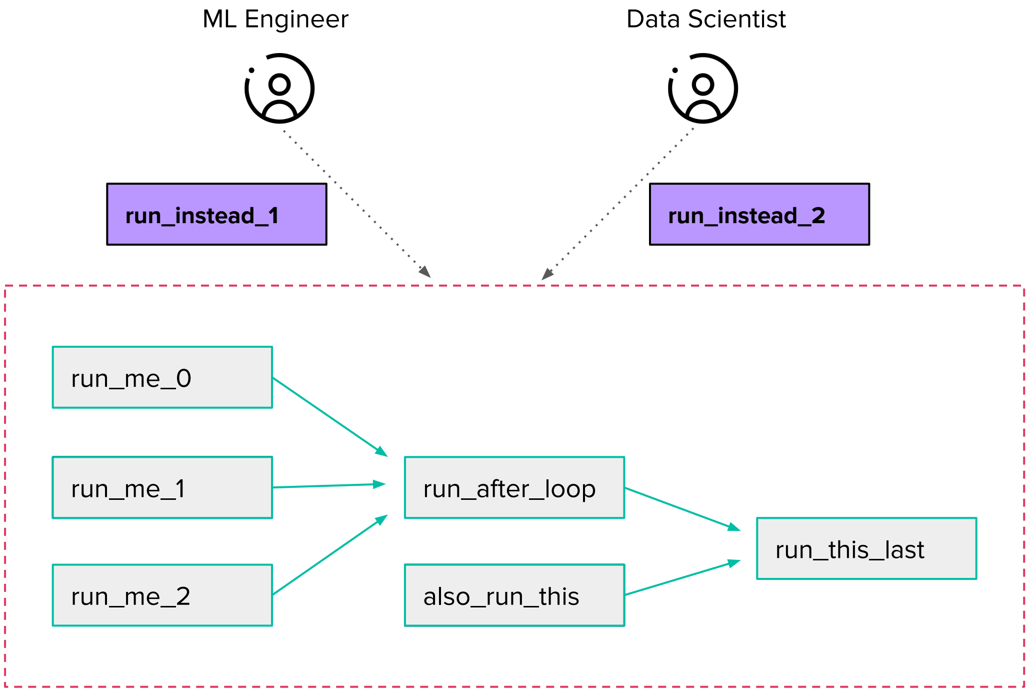
Enable parallel bug fixing + features development while maintaining code versions

The problem:

Multiple developers are using the same pipeline for testing

The solution:

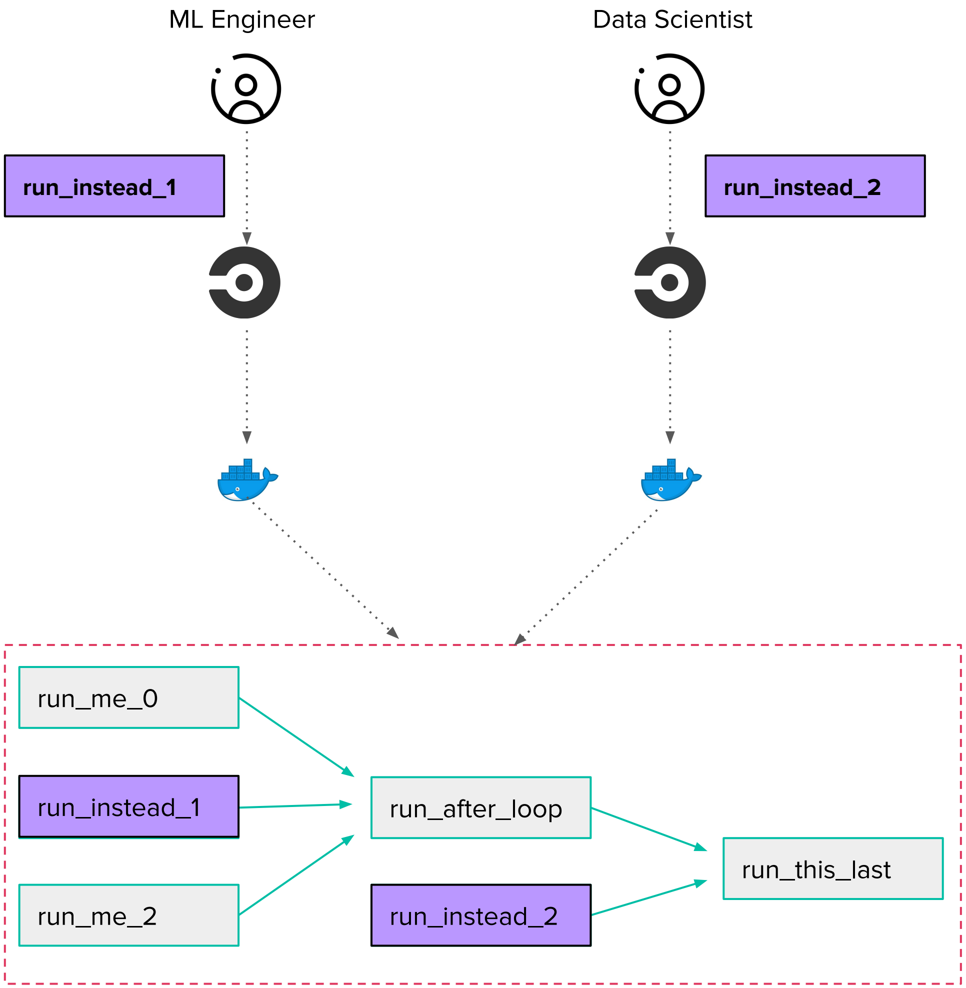
Keep isolated environments within the same pipeline



Riskified's Trainer-as-a-Service

The How:

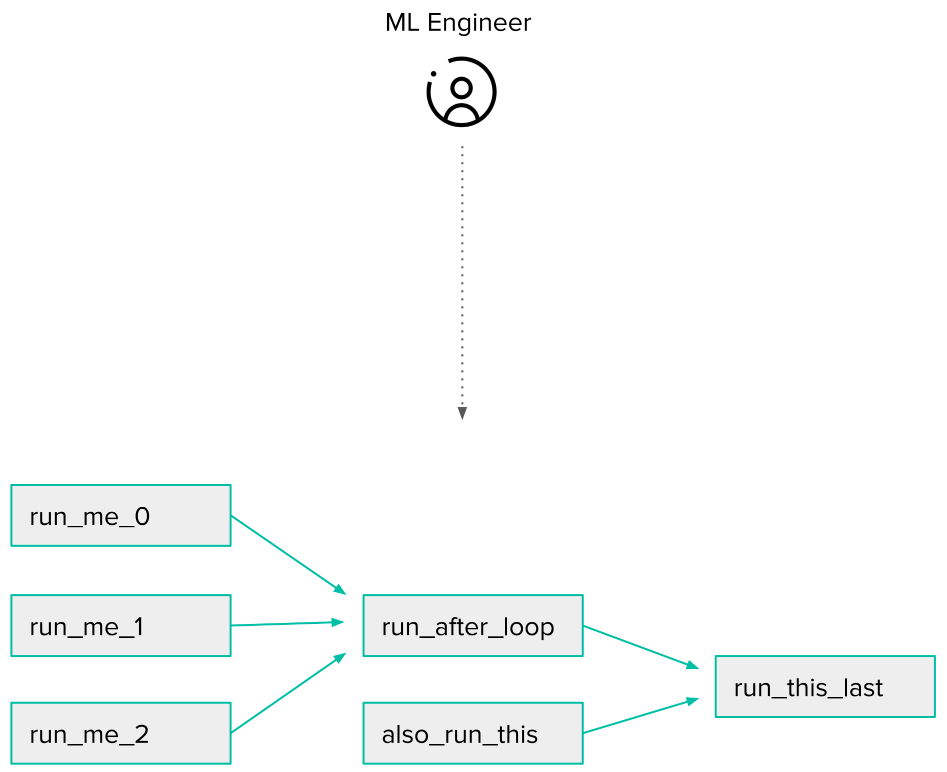
- Create DAGs supporting image as an input
- Create image as part of developer branch CI
- Execute the DAG with a given image



Riskified's Trainer-as-a-Service

A Real as-a-Service

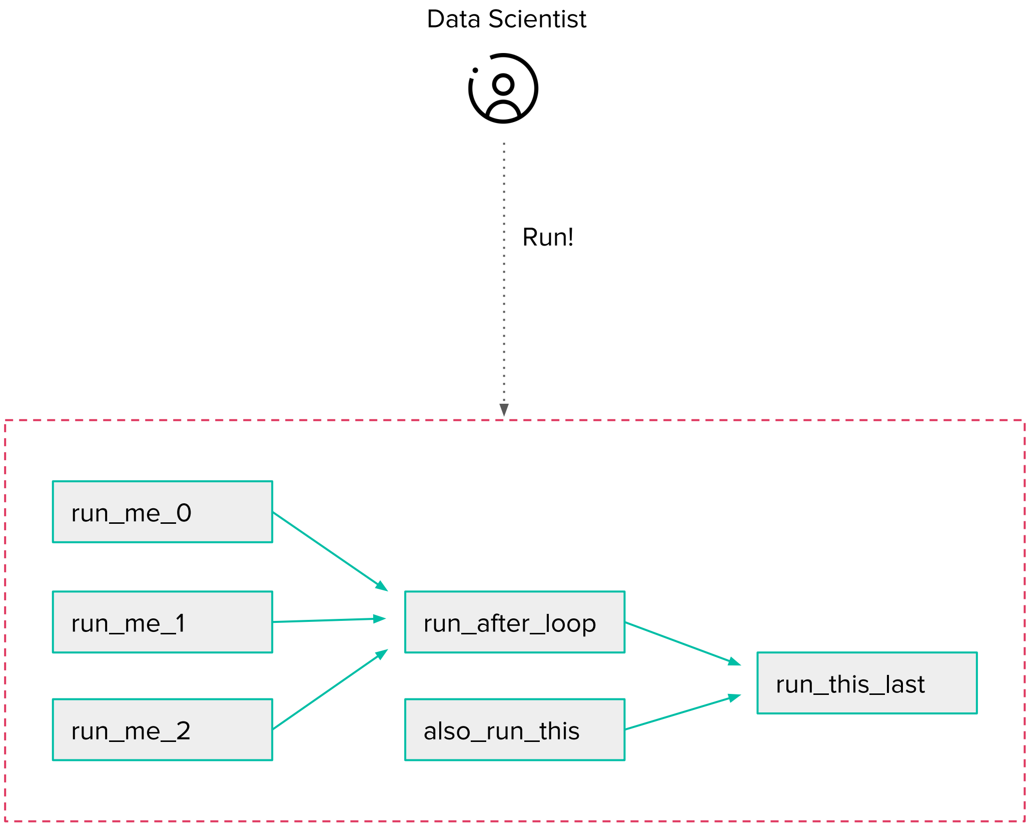
- Clear API
- Easy Triggering Based On Json/YAML File
- Flexible Pipeline - Choose What To Run



Riskified's Trainer-as-a-Service

A Real as-a-Service

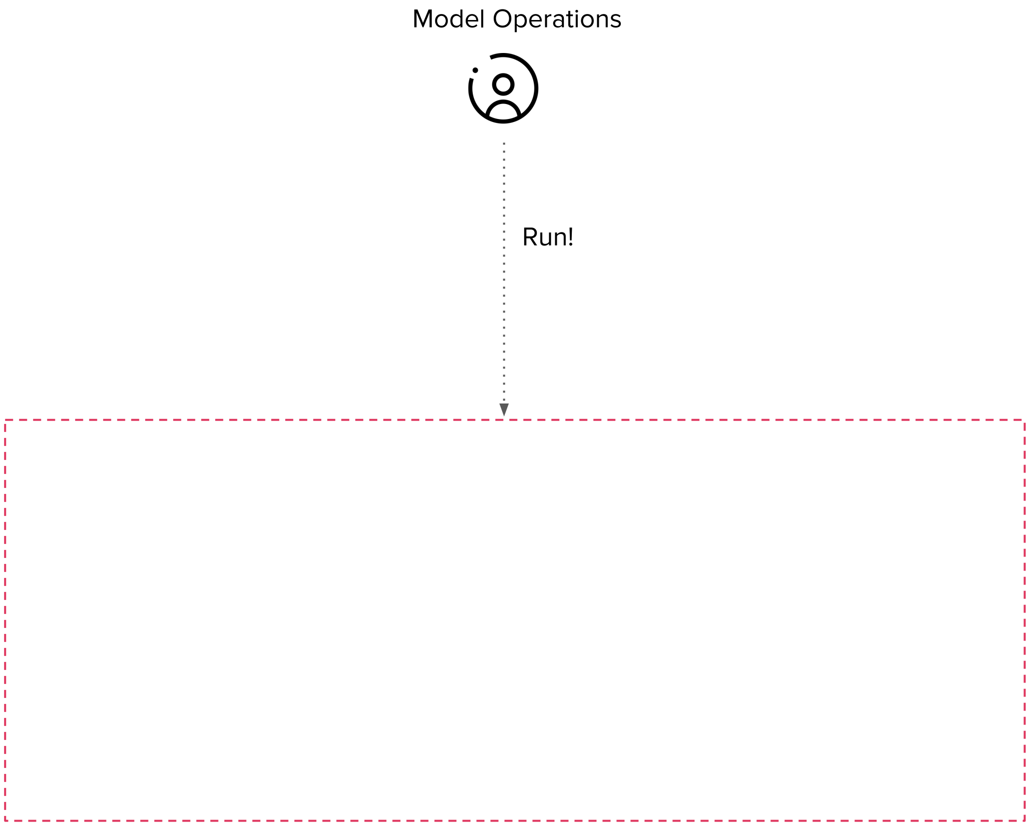
- Clear API
- Easy Triggering Based On Json/YAML File
- Flexible Pipeline - Choose What To Run



Riskified’s Trainer-as-a-Service

A Real as-a-Service

- Clear API
- Easy Triggering Based On Json/YAML File
- Flexible Pipeline - Choose What To Run



Summary

14 Days → 1 Day

We decreased significantly Riskified's operational training effort

Step-by-step

Implementing an automated ML pipeline consists of many steps

Gradually implement and automate towards MLOps level 2

Tech Stack

ECR Vault Airflow Docker k8s CircleCI MLFlow Spark riskified tech;

Thank you for your time!

Q&A