MLOps with Azure AutoML
And
Azure DevOps

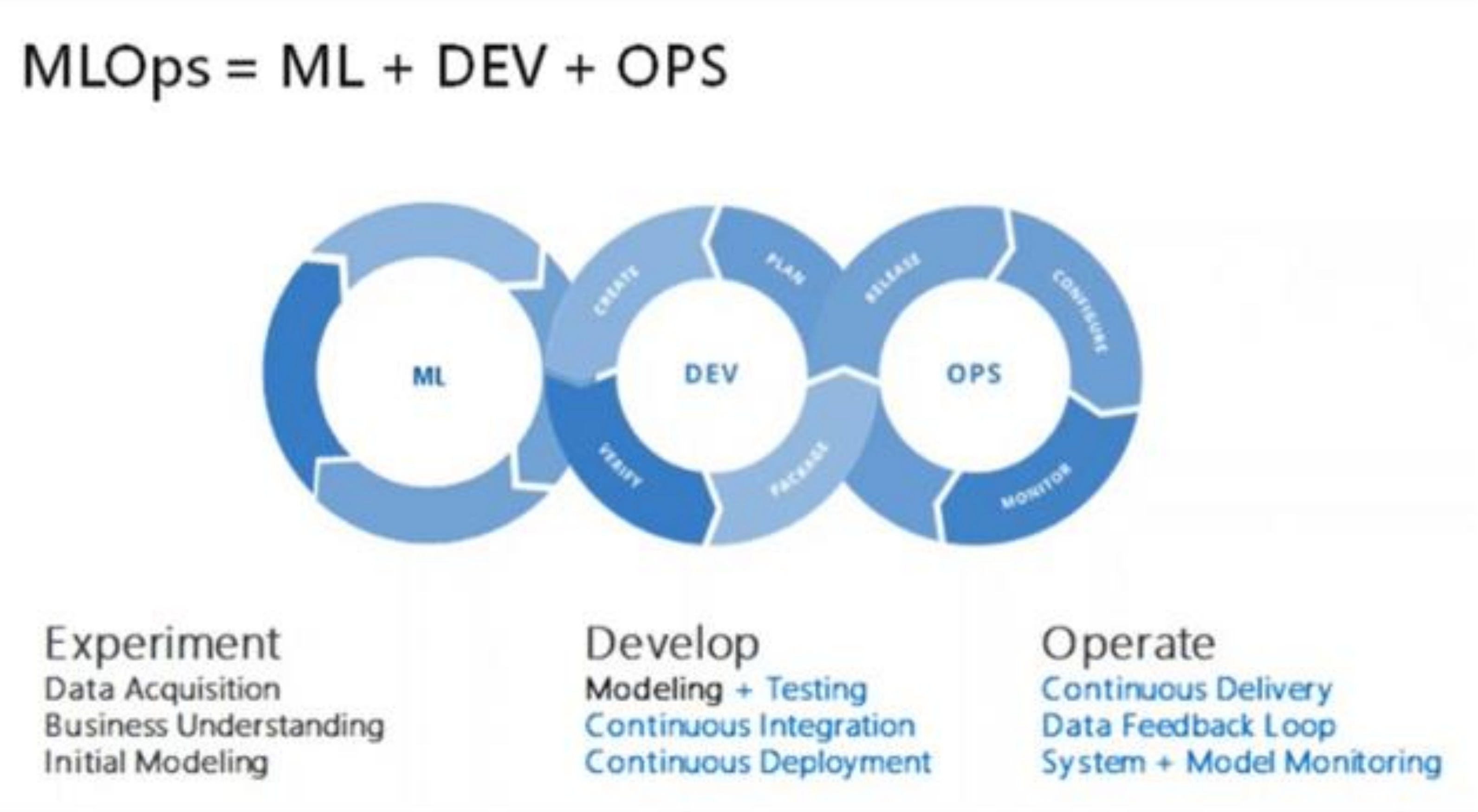
DATA-CORE SYSTEMS

Technology we use to run this project

Technology	Objective/Reason
Azure DevOps	The platform to help you implement DevOps practices on our scenario
Azure Machine Learning Service (Automated ML)	Manage Machine Learning models with the power of Azure
Azure Container Instance(ACI) and Azure Kubernetes Service(AKS)	Deploy Machine Learning models as Docker containers

What is MLOps?

MLOps (a compound of Machine Learning and “information technology OPerationS”) is new discipline/focus/practice for collaboration and communication between data scientists and information technology (IT) professionals while automating and productizing machine learning algorithms.



MLOps = ML + DEV + OPS

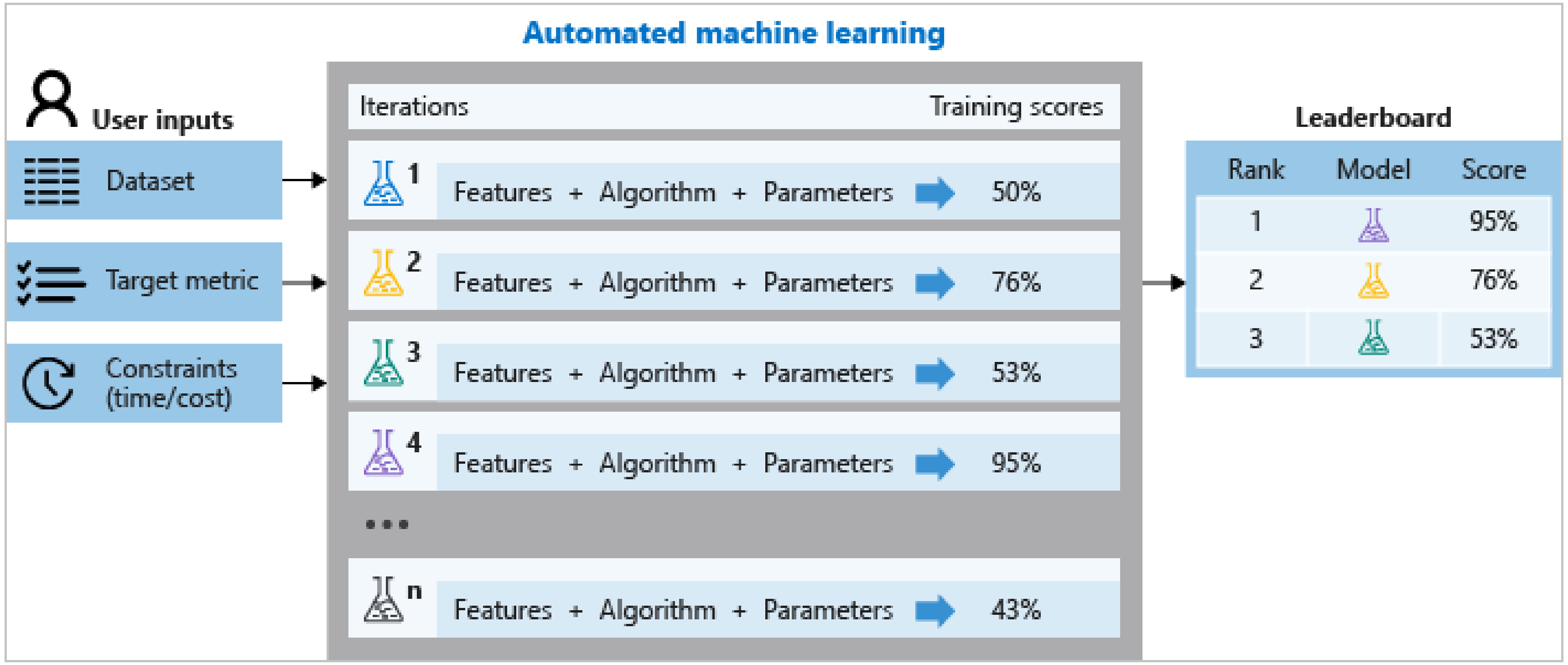
Experiment
Data Acquisition
Business Understanding
Initial Modeling

Develop
Modeling + Testing
Continuous Integration
Continuous Deployment

Operate
Continuous Delivery
Data Feedback Loop
System + Model Monitoring

How AutoML works

- Azure Machine Learning creates a few pipelines in parallel that try different algorithms and parameters for us.
- The service iterates through ML algorithms paired with feature selections, where each iteration produces a model with a training score.
- The higher the score, the better the model is considered to "fit" our data. It will stop once it hits the exit criteria defined in the experiment.



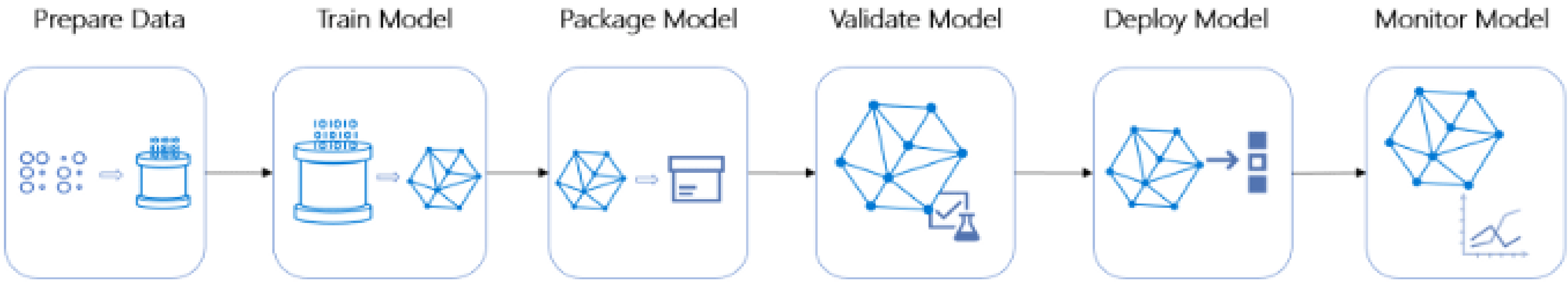
Iterations	Training scores
1 Features + Algorithm + Parameters →	50%
2 Features + Algorithm + Parameters →	76%
3 Features + Algorithm + Parameters →	53%
4 Features + Algorithm + Parameters →	95%
• • •	
n Features + Algorithm + Parameters →	43%

Leaderboard

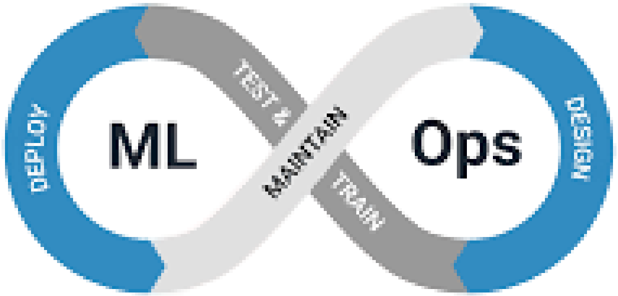
Rank	Model	Score
1	⚗	95%
2	⚗	76%
3	⚗	53%

Azure Pipelines

- Azure Pipelines are cloud-hosted pipelines that are fully integrated with Azure DevOps.
- Azure DevOps allows us to frequently update models, test new models, and continuously roll out new ML models alongside our other applications and services.
- The end-to-end Machine learning pipeline includes data-prep, training, packaging and validating model, deploying model and continuous testing. It looks like below

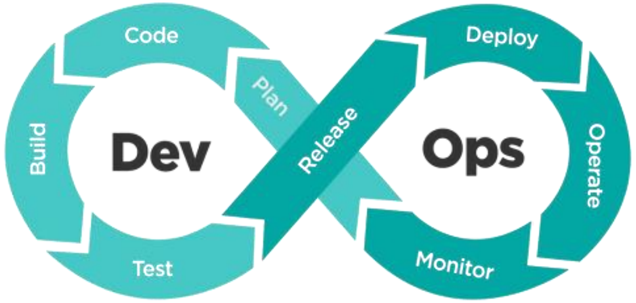


MLOps and DevOps Aims

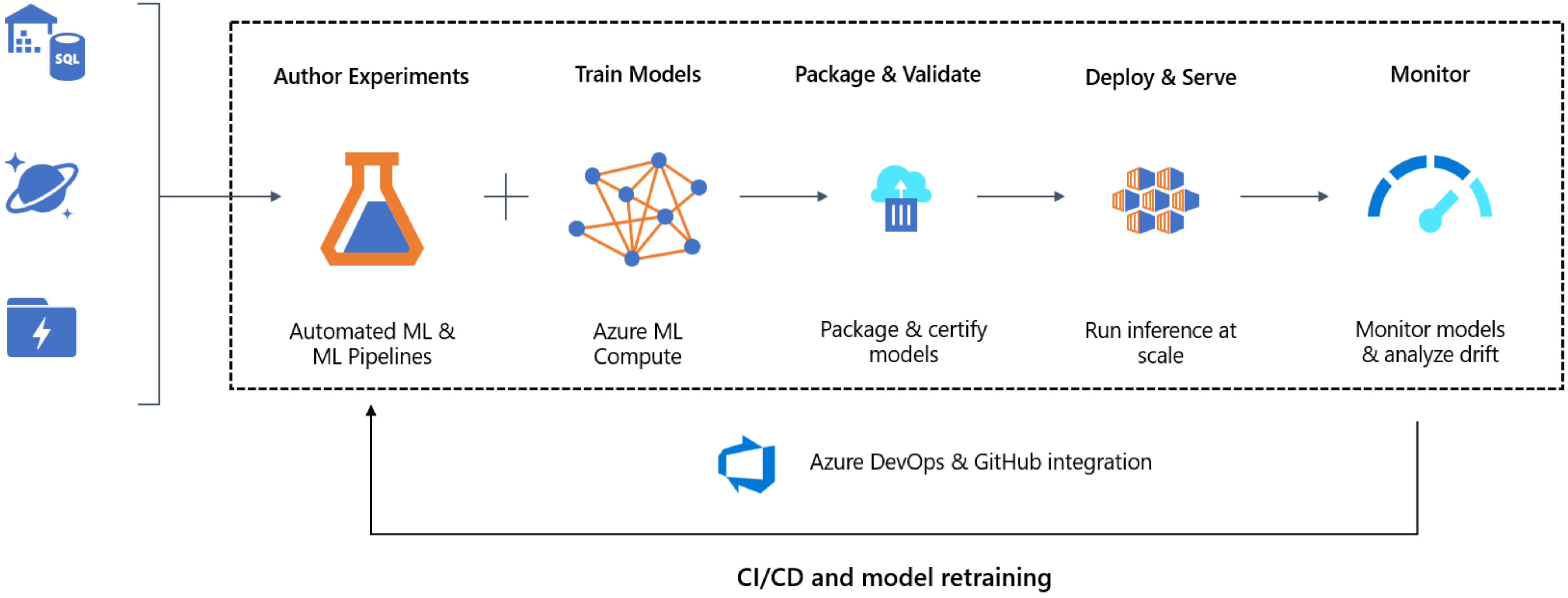


MLOps aims to establish a culture and environment where ML technologies can generate business benefits by rapidly, frequently and reliably building, testing, and releasing ML technology into production.

We will be using the Azure DevOps project for build and release pipelines along with Azure AutoML services for ML/AI model management and operationalization.



Azure Machine Learning service



Author Experiments

Automated ML & ML Pipelines

Train Models

Azure ML Compute

Package & Validate

Package & certify models

Deploy & Serve

Run inference at scale

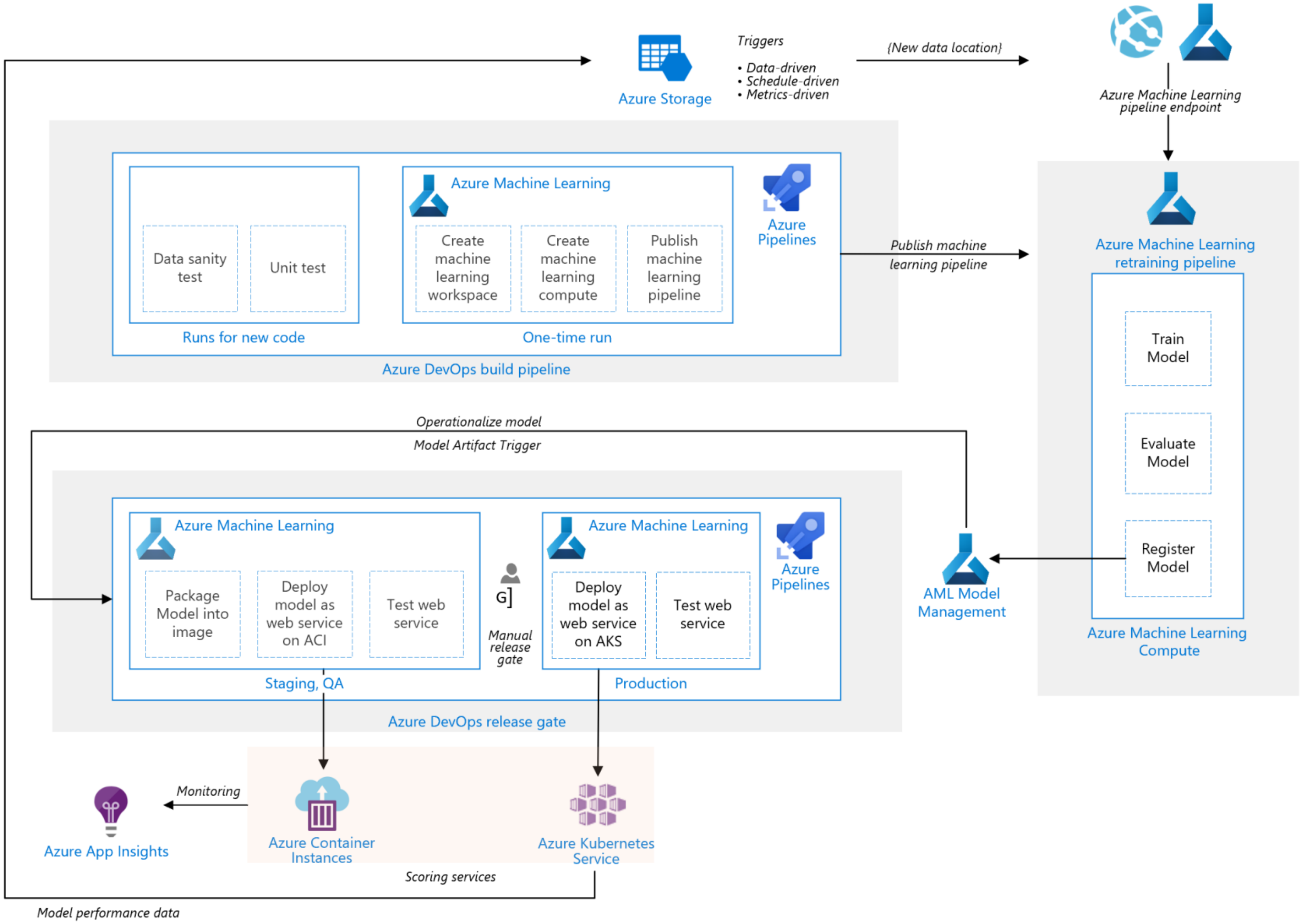
Monitor

Monitor models & analyze drift

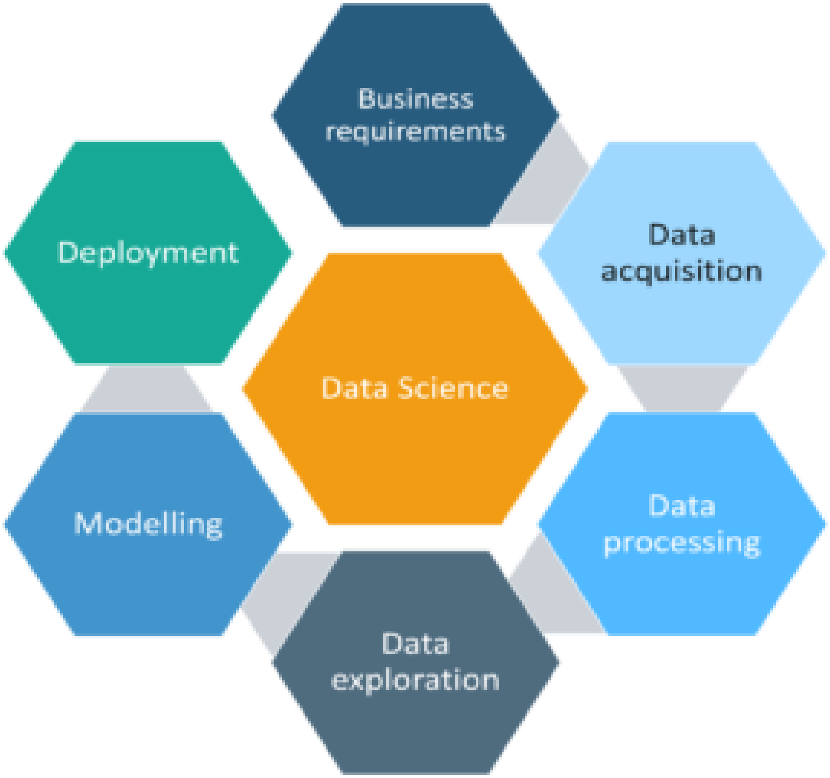
Azure DevOps & GitHub integration

CI/CD and model retraining

CI/CD pipeline will look like below



Architecture and Features



- Implement continuous integration (**CI**), continuous delivery (**CD**), and training pipeline for an AI application using **Azure DevOps** and **Azure Machine Learning**.

- The solution is built on the **classification** scenario but can be easily adapted for any AI scenario.

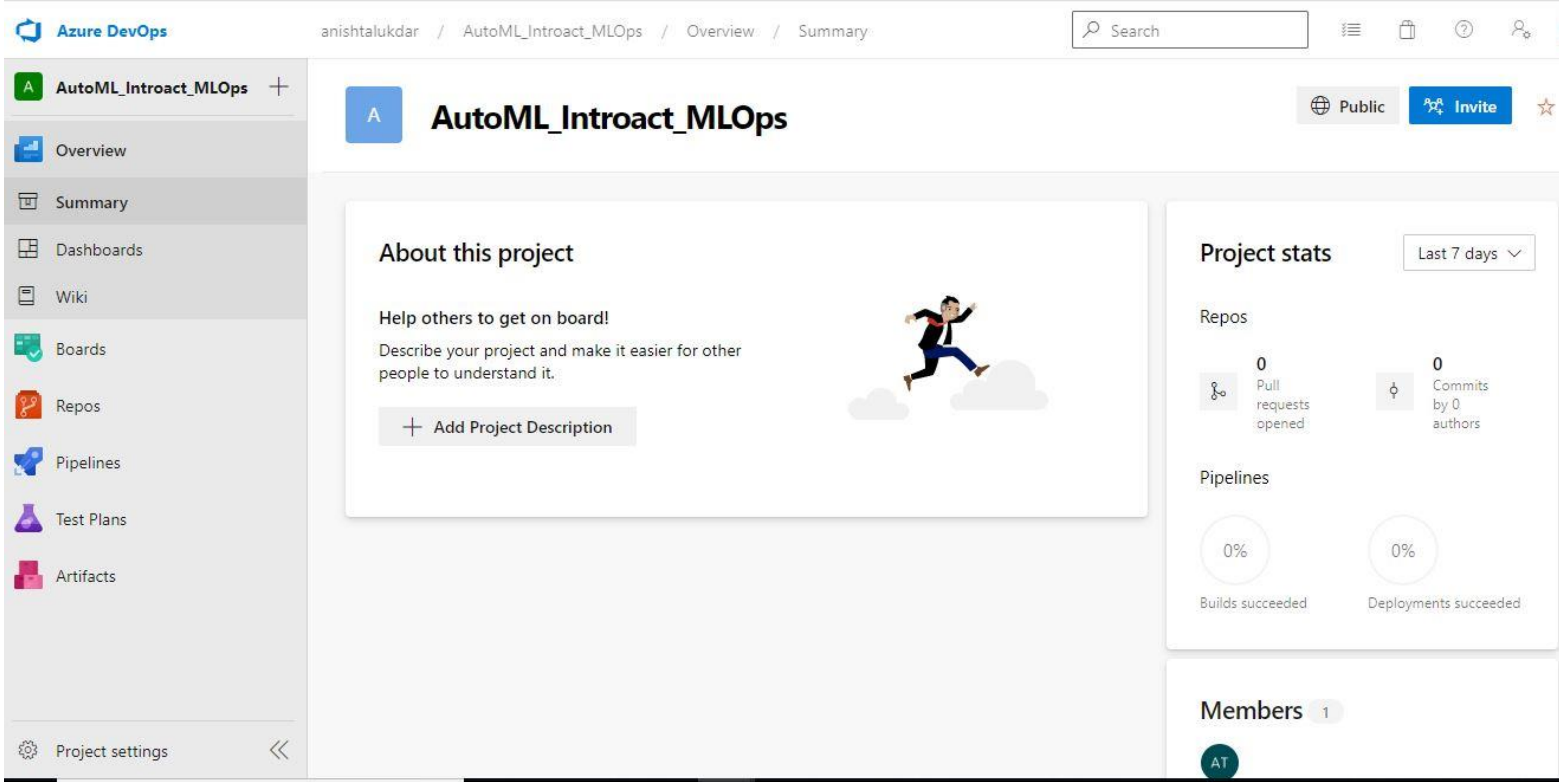
- The build pipelines include DevOps tasks for data sanity tests, unit tests**, automated model training** on different compute targets.

- Selecting the **best model** including model version management, **model evaluation/model selection**, model deployment as **real-time web service**, staged deployment to QA/prod and integration testing.

Configure CI pipeline

- In this step, we will configure CI pipeline for your ML/AI project. This pipeline will include DevOps tasks for
- data sanity test
- model training on different compute targets
- model version management
- model evaluation/model selection etc.

Azure DevOps Dashboard



AutoML_Introact_MLOps

About this project

Help others to get on board!

Describe your project and make it easier for other people to understand it.

+ Add Project Description

Project stats — Last 7 days

Repos

0 Pull requests opened

0 Commits by 0 authors

Pipelines

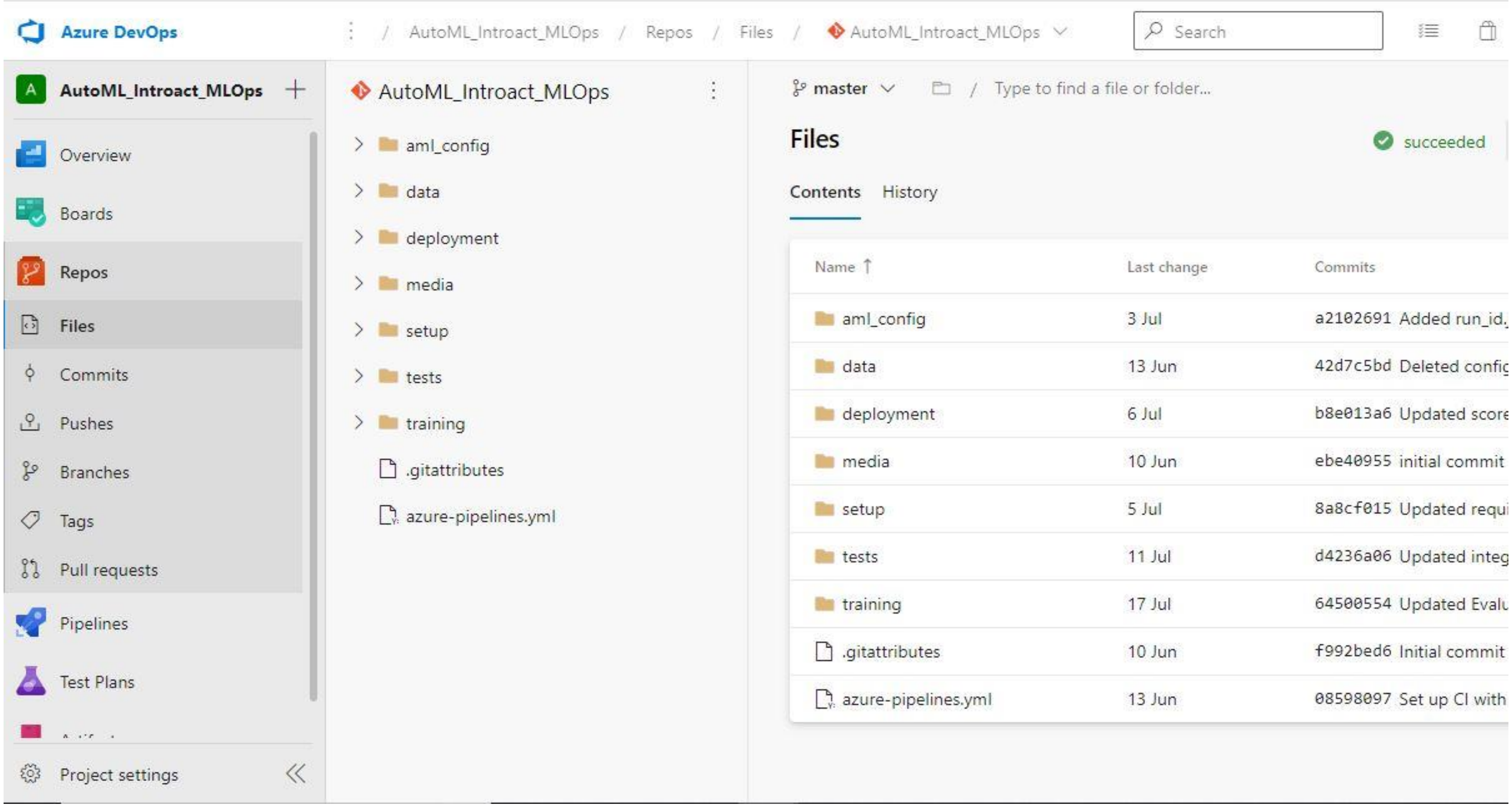
0% Builds succeeded

0% Deployments succeeded

Members 1

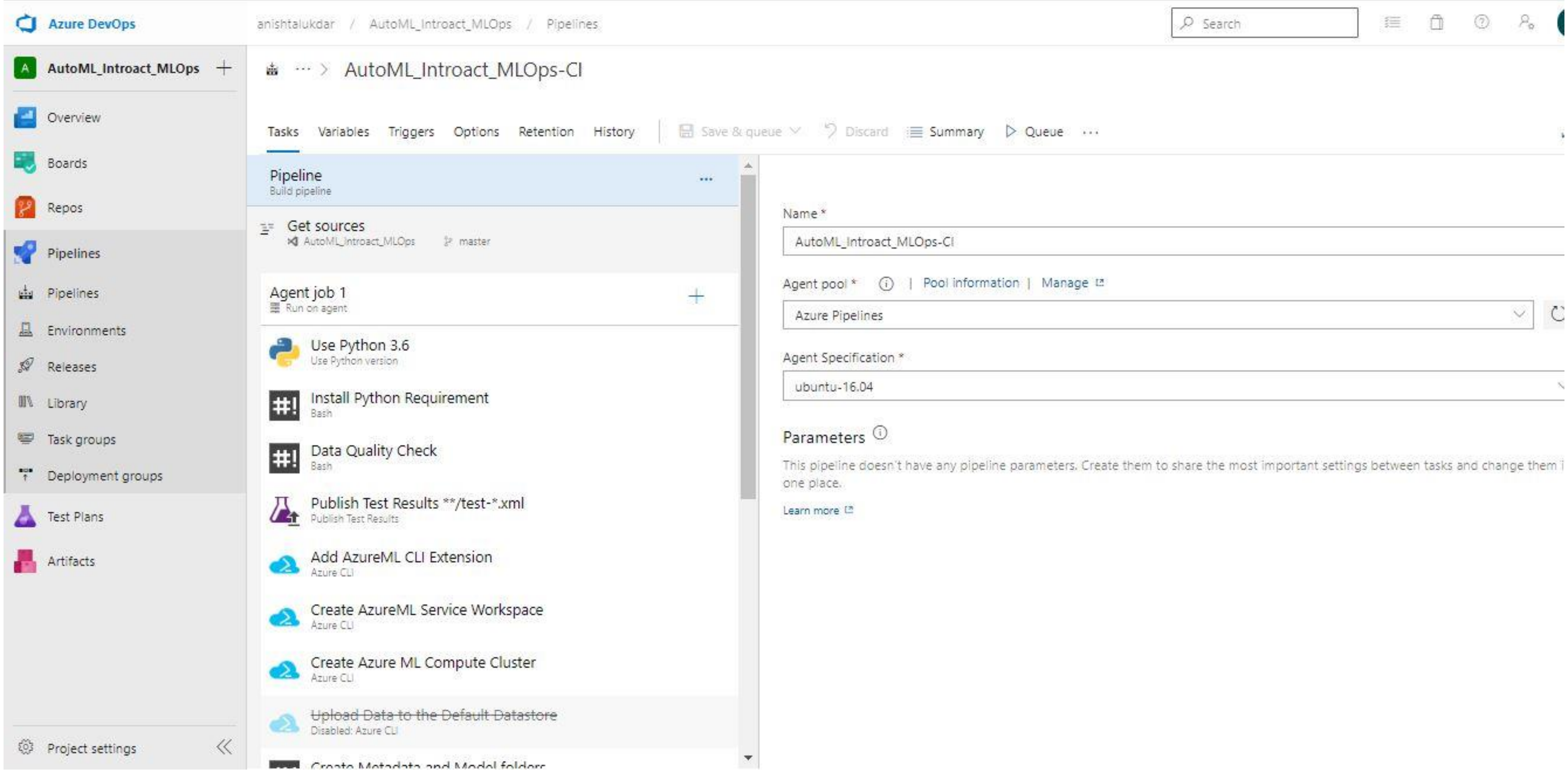
AT

Git Repository



Name ↑	Last change	Commits
aml_config	3 Jul	a2102691 Added run_id,
data	13 Jun	42d7c5bd Deleted config
deployment	6 Jul	b8e013a6 Updated score
media	10 Jun	ebe40955 initial commit
setup	5 Jul	8a8cf015 Updated requi
tests	11 Jul	d4236a06 Updated integ
training	17 Jul	64500554 Updated Evalu
.gitattributes	10 Jun	f992bed6 Initial commit
azure-pipelines.yml	13 Jun	08598097 Set up CI with

Pipeline configs will look like below



AutoML_Introact_MLOps-CI

Pipeline
Build pipeline

Get sources
AutoML_Introact_MLOps   master

Agent job 1
Run on agent

Use Python 3.6
Use Python version

Install Python Requirement
Bash

Data Quality Check
Bash

Publish Test Results **/test-*.xml
Publish Test Results

Add AzureML CLI Extension
Azure CLI

Create AzureML Service Workspace
Azure CLI

Create Azure ML Compute Cluster
Azure CLI

~~Upload Data to the Default Datastore~~
Disabled: Azure CLI

Create Metadata and Model folders

Name *
AutoML_Introact_MLOps-CI

Agent pool * | Pool information | Manage
Azure Pipelines

Agent Specification *
ubuntu-16.04

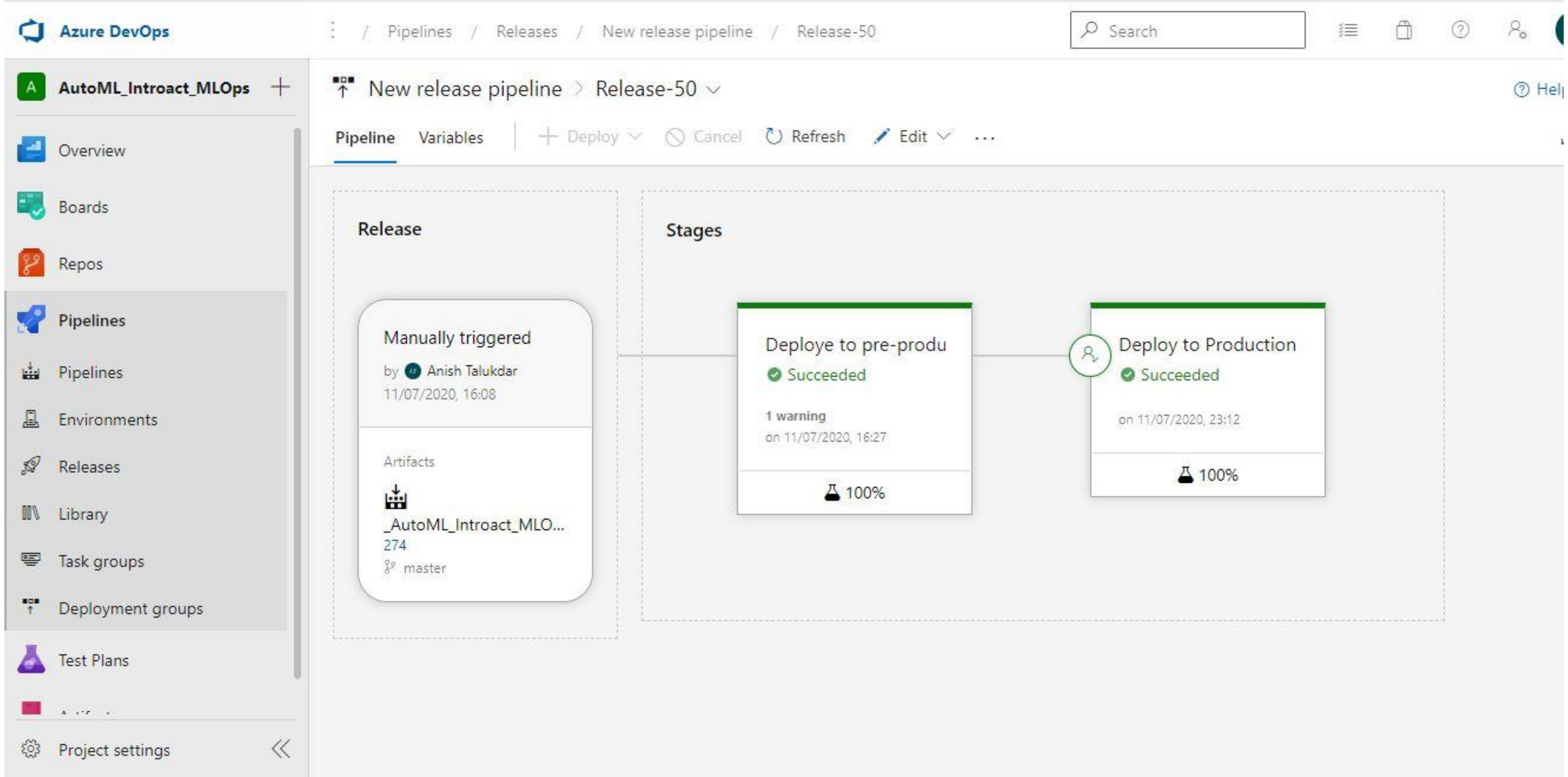
Parameters

This pipeline doesn't have any pipeline parameters. Create them to share the most important settings between tasks and change them i one place.

Learn more

Configure CD pipeline

So, as we are done with CI pipeline let’s configure Release pipeline which will deploy the image created from the build pipeline to Azure Container Instance and Azure Kubernetes Services.



Manually triggered
by Anish Talukdar
11/07/2020, 16:08

Artifacts
_AutoML_Introact_MLO...
274
master

Deploye to pre-produ
Succeeded
1 warning
on 11/07/2020, 16:27
100%

Deploy to Production
Succeeded
on 11/07/2020, 23:12
100%

Thank You

DATA-CORE SYSTEMS