



# Workshop Azure MLOps

16 mars 2020



# Vos interlocuteurs



- Serge Retkowsky
- [serge.retkowsky@microsoft.com](mailto:serge.retkowsky@microsoft.com)

# Version du document

- V 1.1
- 16 mars 2020

# MLOps ?



# The story of Machine Learning



**ginablaber**  
@ginablaber

[Follow](#)



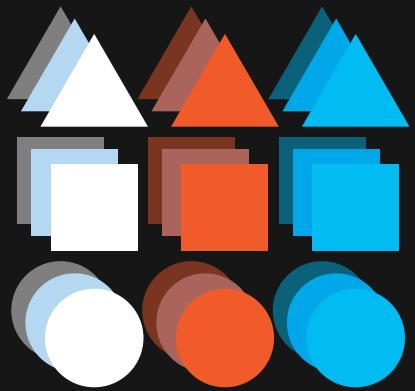
The story of enterprise Machine Learning: “It took me 3 weeks to develop the model. It’s been >11 months, and it’s still not deployed.”

**@DineshNirmalIBM #StrataData #strataconf**

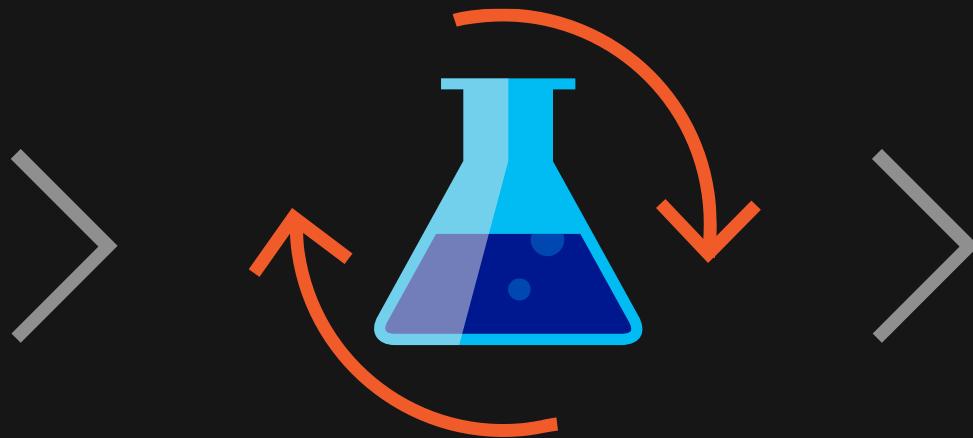
10:19 AM - 7 Mar 2018

# Custom AI

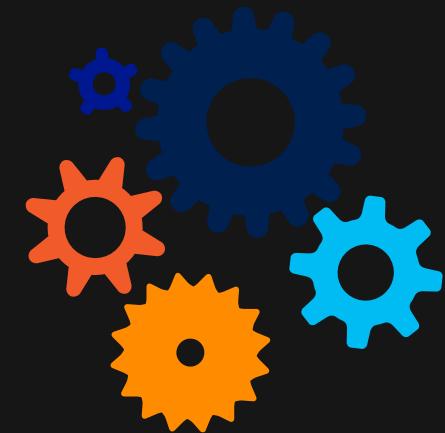
*Building your own AI models for Transforming Data into Intelligence*



Prepare Data

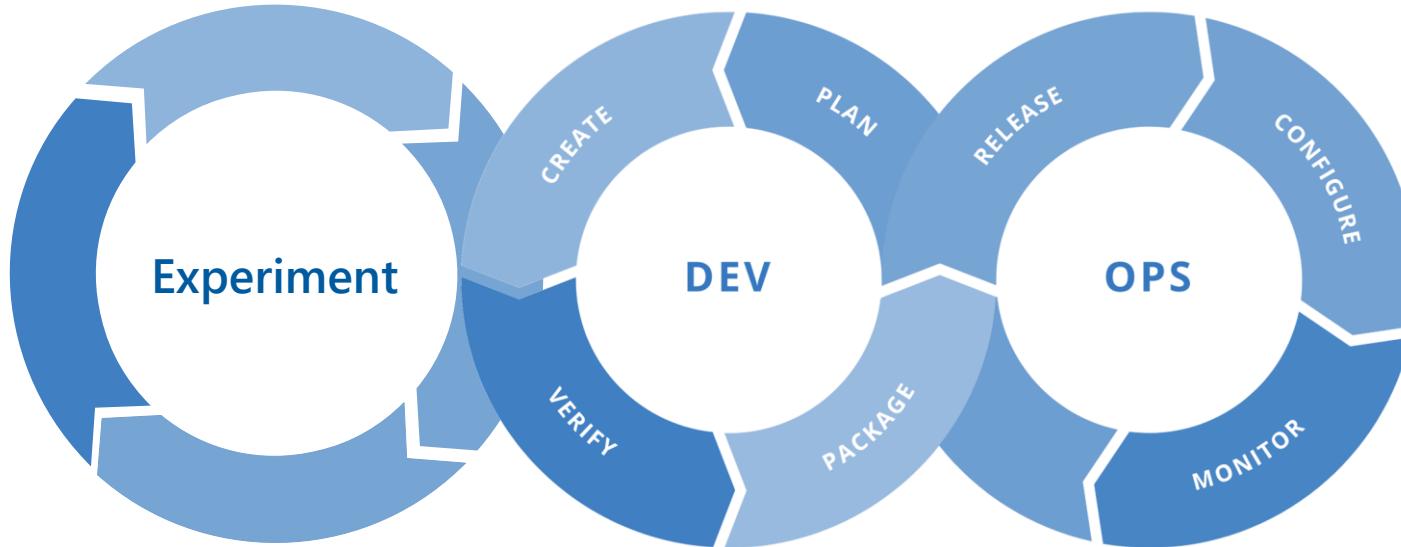


Build & Train



Deploy

# AI DevOps lifecycle



## Experiment

Data Acquisition  
Business Understanding  
Initial Modeling

## Develop

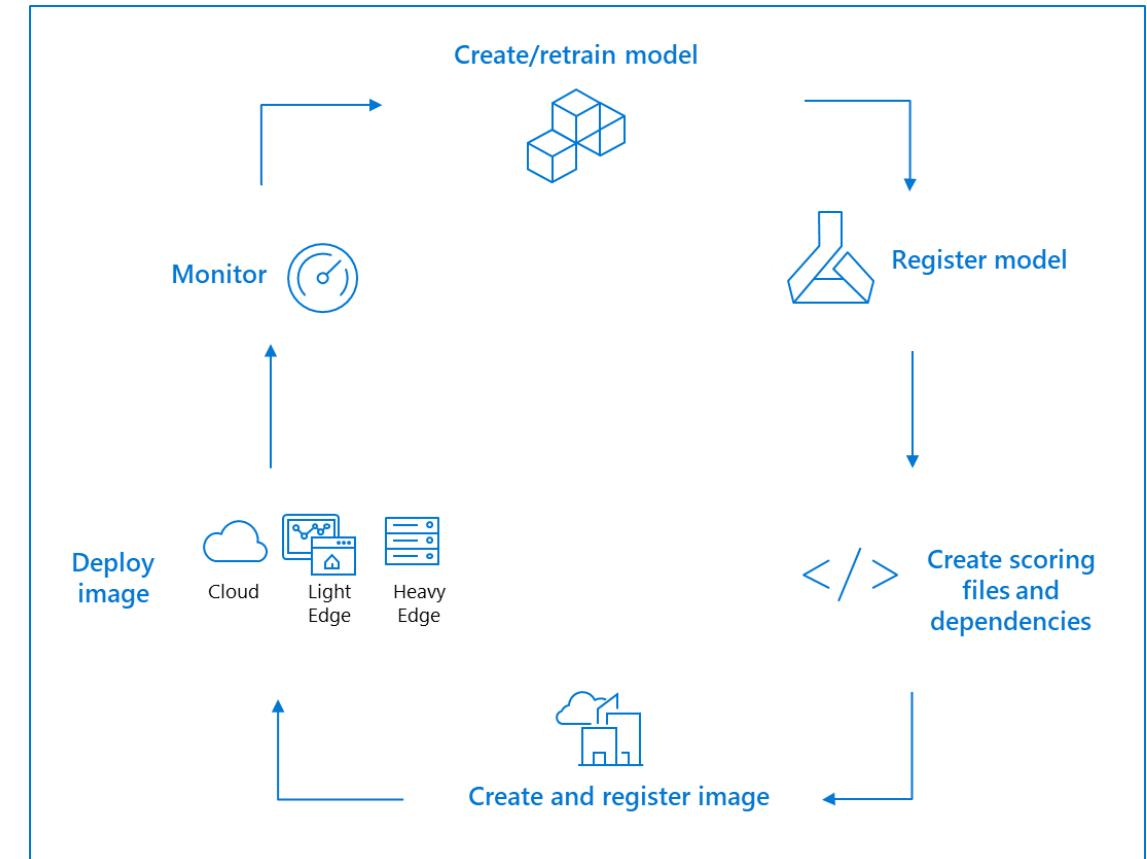
Modeling + Testing  
Continuous Integration  
Continuous Deployment

## Operate

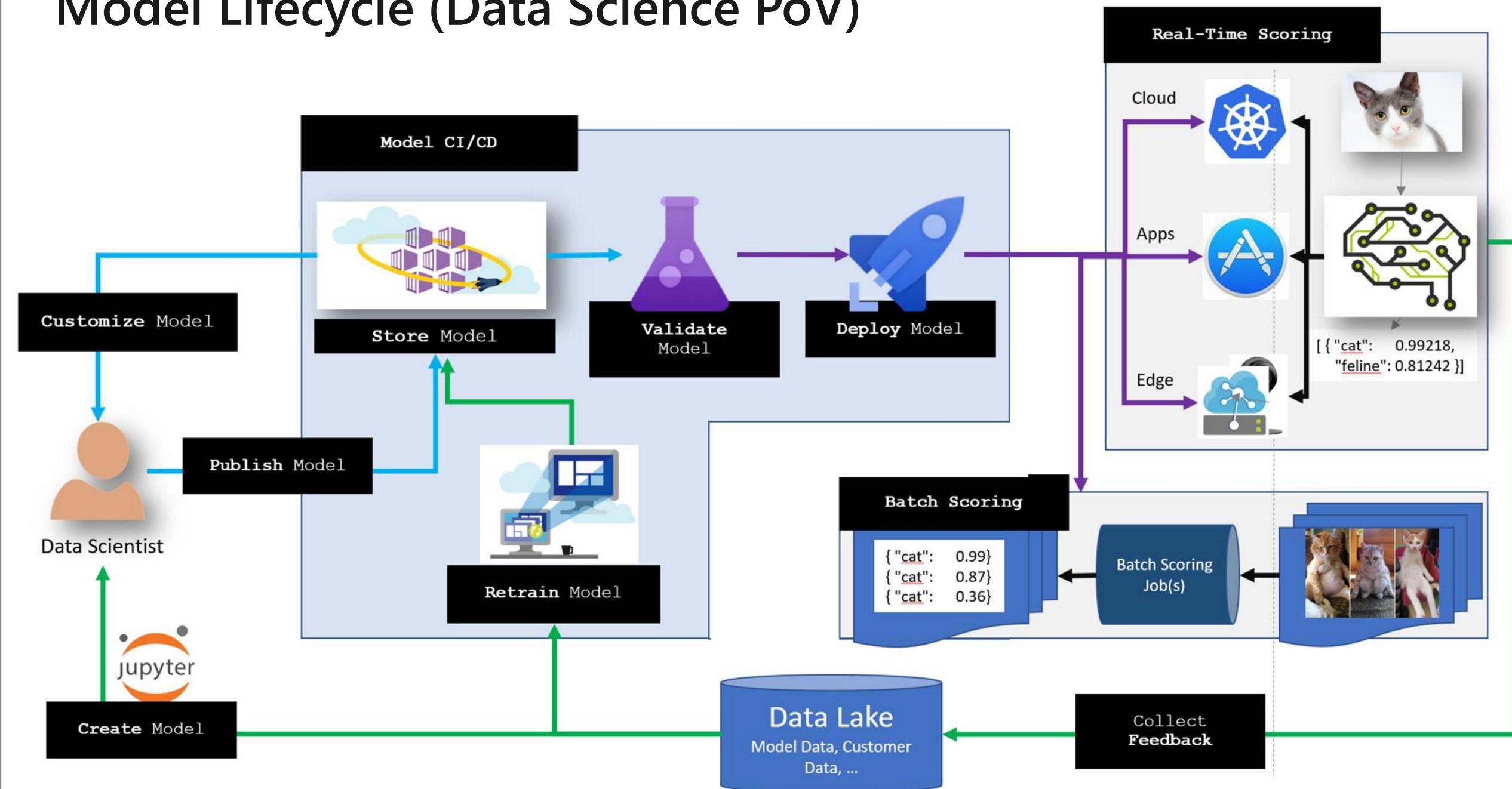
Continuous Delivery  
Data Feedback Loop  
System + Model Monitoring

# Manage Model Lifecycle

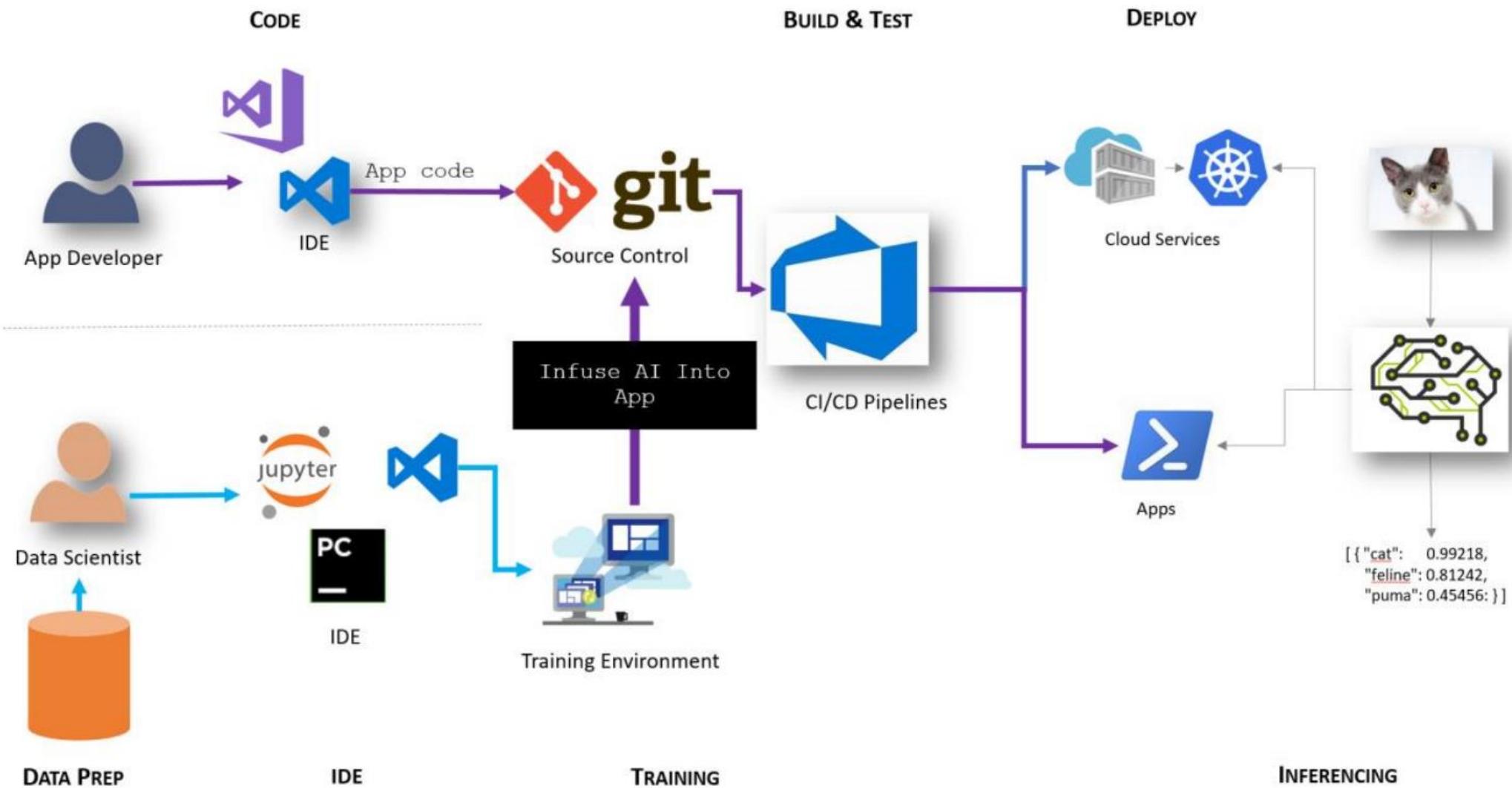
- **Track model versions & metadata** with a centralized **model registry**
- **Leverage containers** to capture runtime dependencies for inference
- Leverage an orchestrator like **Kubernetes** to provide **scalable inference**
- Capture **model telemetry** – health, performance, inputs / outputs
- **Encapsulate each step** in the lifecycle to **enable CI/CD and DevOps**
- Automatically **optimize models** to take advantage of hardware acceleration



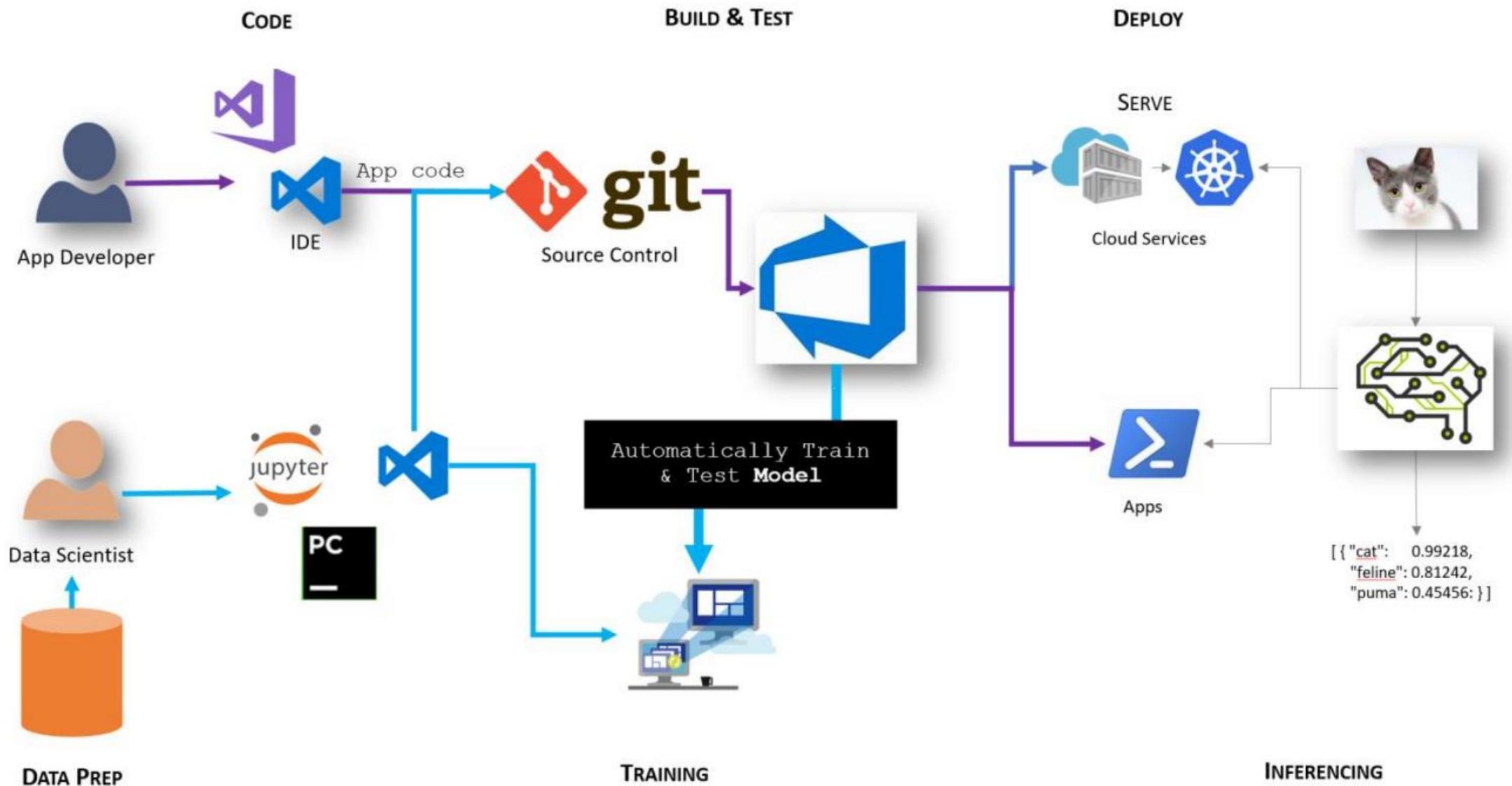
# Model Lifecycle (Data Science PoV)



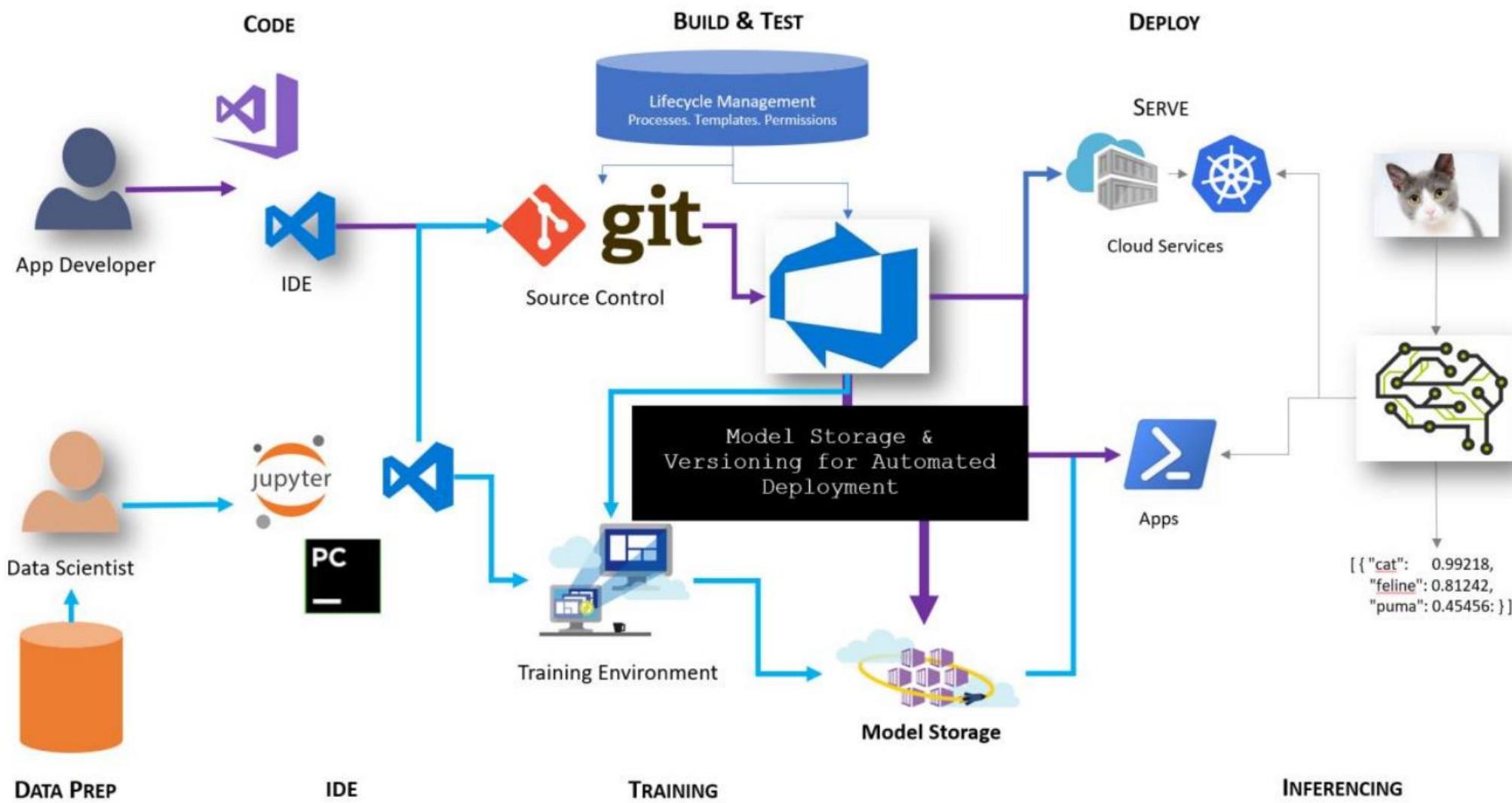
# Step 1 – Infuse AI into App



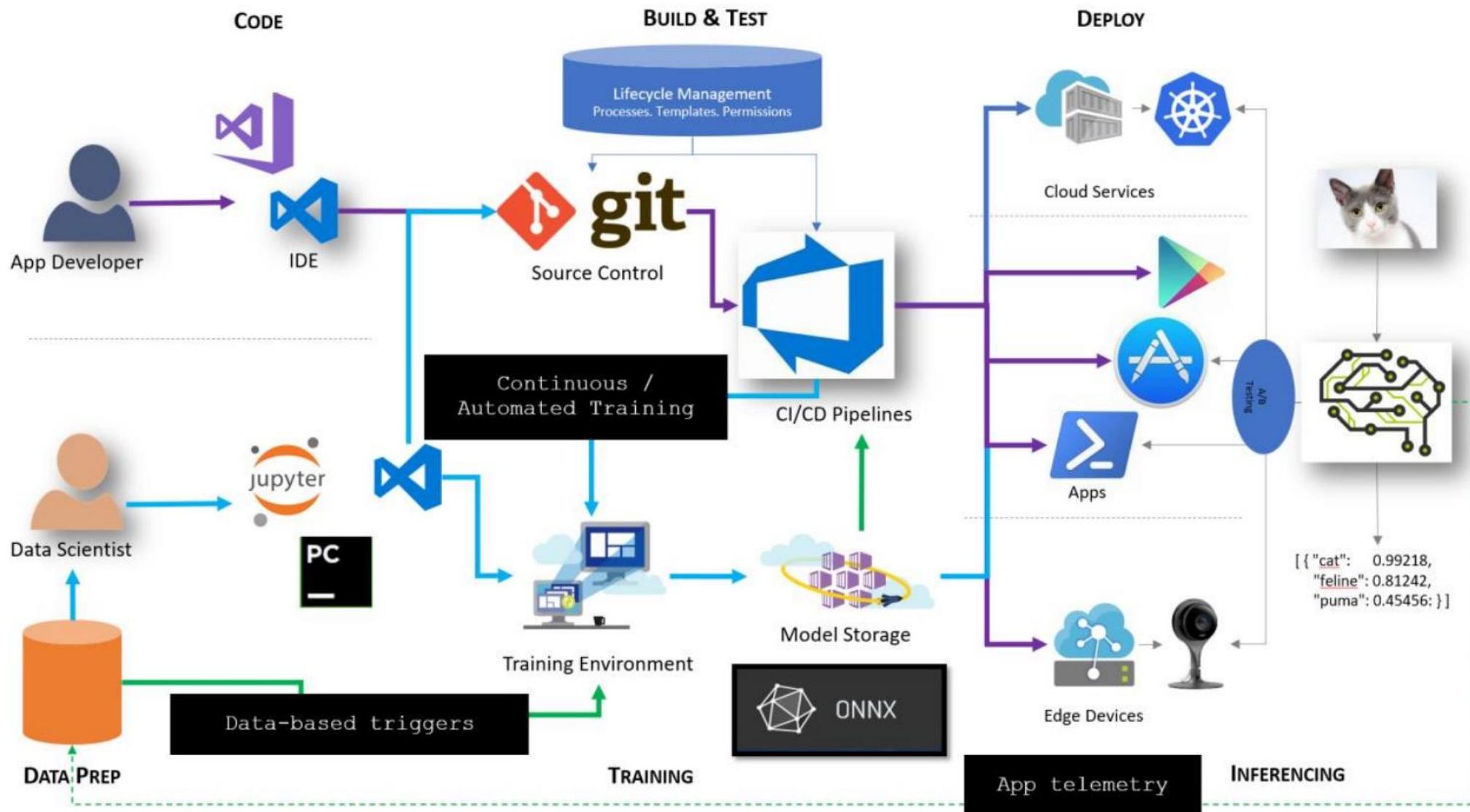
# Step 2 – Automate Model Training



# Step 3 – Store, Version and Validate Models

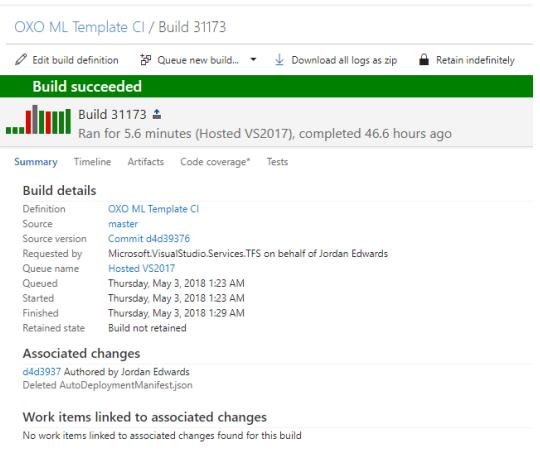
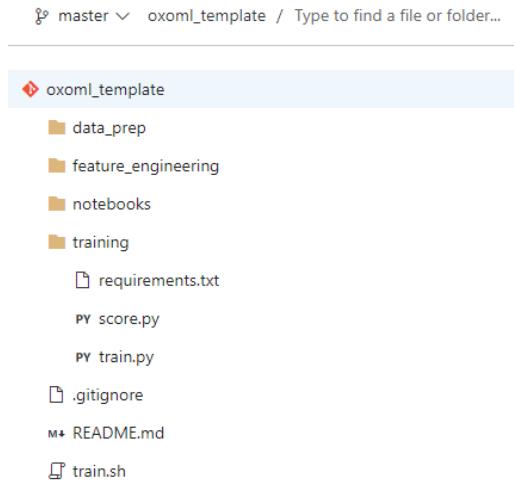
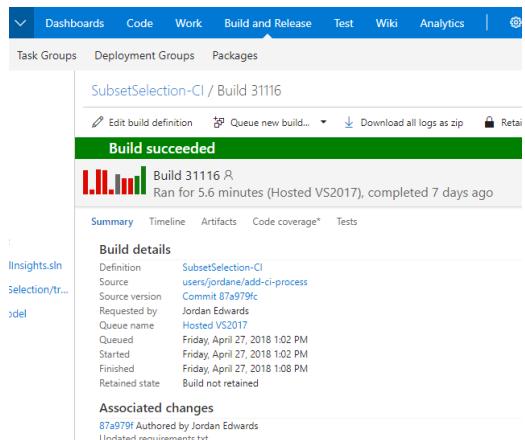
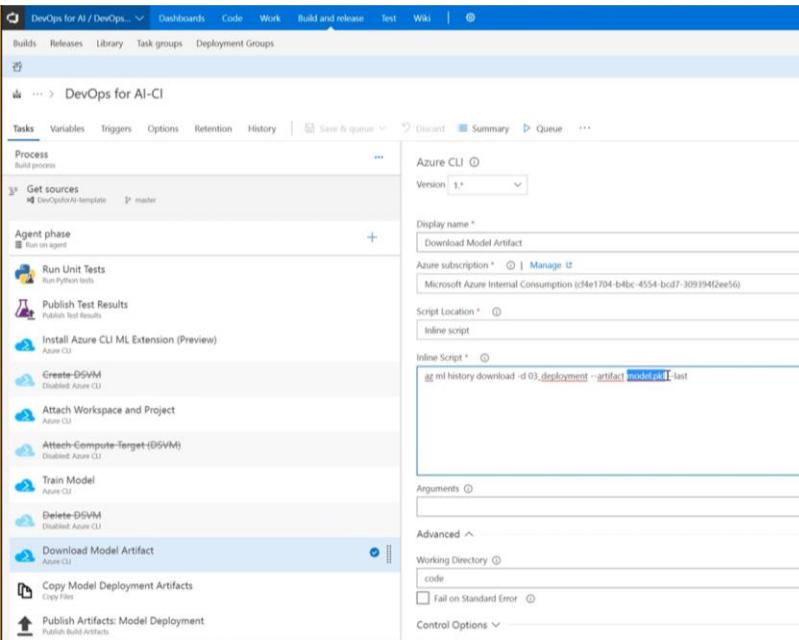


# Step 4 – Automate Model Release



# Model Reproducibility

- Processes and procedures to make models reproducible
  - (code / data / config)
- Capture
  - Featurization code (w/ tests)
  - Training pipeline
  - Dependencies
  - Training data persistence
  - **Evidence chain**
  - Model config
  - Training job info
  - Sample data
  - Data profile



# Model Validation

- Profile and validate
  - Data (changes to shape / profile)
  - Model in isolation (offline A/B)
  - Model + app (functional testing)
- Gate promotion of the model on validation
- Assess
  - Functional behavior
  - Performance characteristics

The screenshot shows a validation interface with the following sections:

- Policies**:
  - Required**:
    - ✓ 2 reviewers approved
    - ✓ Required reviewers have approved
    - ✓ Work items linked
    - ✓ Gated Build - AEther Modules succeeded
    - ✓ Gated Build - Main succeeded
    - ✓ Gated Build - Quantum succeeded
    - ✓ Gated Build - WebTools succeeded
    - ✓ TEE Perf Metrics succeeded
- ATTRIBUTES**:

Status	Completed
Start Time	May 1, 2018, 11:18:46 AM
Duration	5s
Target	sdk
Run Id	9ee0e521-d32b-45f1-8958-0eddf9acd9cf
Run Number	40
Script Name	run_client.py
Arguments	
- PROPERTIES**:

accuracy	0.995253164556962
precision	0.8888888888888888
recall	0.958333333333334
feature_importance	{"Importance": [{"min": 0.016218788, "numericalSomeness": 0.0194017469, "max1": 0.0205532641, "variance1": 0.026353736, "uniqueValues": 12}]} A line chart titled "feature_importance" showing a sharp increase starting around index 10.

The screenshot shows a metrics dashboard with the following sections:

- Metrics**:
  - Build filter: a83c0947|125b6e2b|b238a3d For example: develop&4f2d346|1415 Filter
  - develop a83c094 develop 125b6e2 PR#6821 b238a3d
  - Build summary**:

	develop a83c094	develop 125b6e2	PR#6821 b238a3d
.NET CLR Memory(Hippogriff.Server)\% Time in GC (85th %)	4.35	↑ +2.5% 4.46	↓ -4% 4.28
Process(Hippogriff.Server)\% Processor Time (85th %)	29.4	↓ -4.4% 28.1	↑ +3.9% 29.2
Process(Hippogriff.Server)\Private Bytes (85th %)	490Mb	↓ -0.8% 486Mb	↑ +13.8% 552Mb
Process(Hippogriff.Server)\Working Set - Private (85th %)	459Mb	↓ -0.9% 455Mb	↑ +14.5% 521Mb

# Model Versioning & Storage

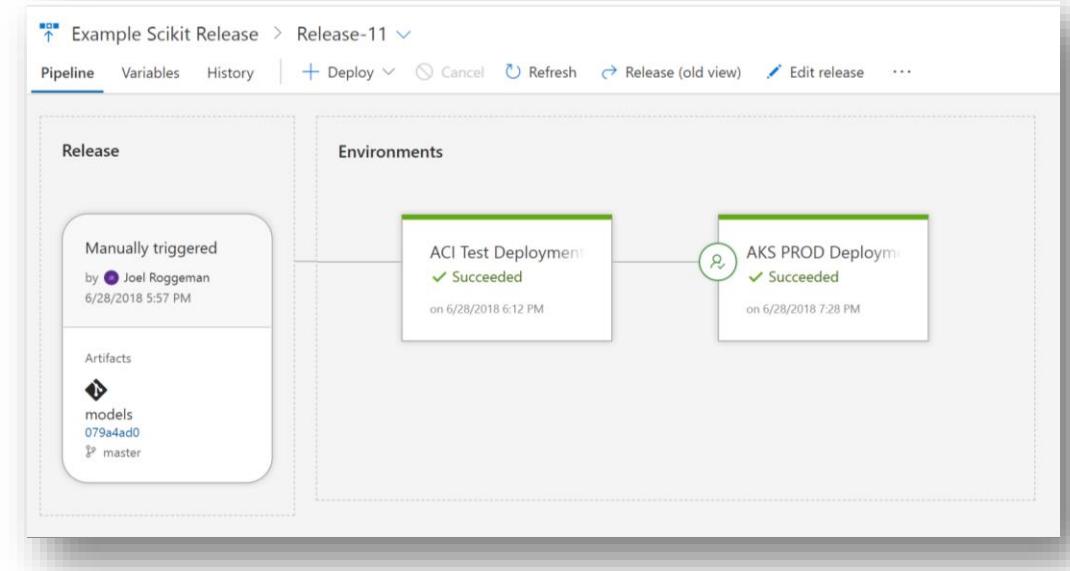
- Provide a consistent way to discover, store, track & share models
- Provide a consistent model metadata format
- Track where a model came from
  - which **data**,
  - which **experiment / previous model(s)**
  - where's the **code / notebook**
  - Was it **converted / quantized?**
- Track where model is running
- Control who has access to what models
  - Private / compliant data

**ONNX** – goal is to be interoperable format for models

Model Registry		
<input type="checkbox"/>	NAME	VERSION
<input type="checkbox"/>	best_model.pkl	13
<input type="checkbox"/>	best_model.pkl	12
<input type="checkbox"/>	best_model.pkl	11
<input type="checkbox"/>	best_model.pkl	10
<input type="checkbox"/>	best_model.pkl	9
<input type="checkbox"/>	best_model.pkl	8
<input type="checkbox"/>	best_model.pkl	7
<input type="checkbox"/>	best model.pkl	6

# Model Deployment

- Safe and efficient deployment & feedback
- Simplify consumption - code-generation, API specifications / interfaces
- Support a variety of inferencing targets
  - Cloud Services
  - Mobile / Embedded Applications
  - Edge Devices
- Convert / quantize / optimize models for target platform
- Control the rollout of your models (with A/B)
- Feed telemetry back into your system on service health and model behavior



Models	Service "pairingexcel"
Manifests	Service id pairingexcel.airestestcluster-2f813172.eastus2
Images	Creation date 5/1/2018, 12:42:38 PM
Services	Last updated 5/1/2018, 12:42:58 PM
	State Succeeded
	Environment airestestcluster
	URL http://13.68.75.180/api/v1/service/pairingel
	Primary key eqsCYasTxddYwePMo5wHiuqVjoNAJbef
	Secondary key pN01Sr51r95uLZ9alht3vimmw4UYR17L
	Event hub false
	Storage true
	CPU reserve capacity 0.1
	Memory reserve capacity 500M
	Scale settings
	Scale Type Auto

# What are ML pipelines?

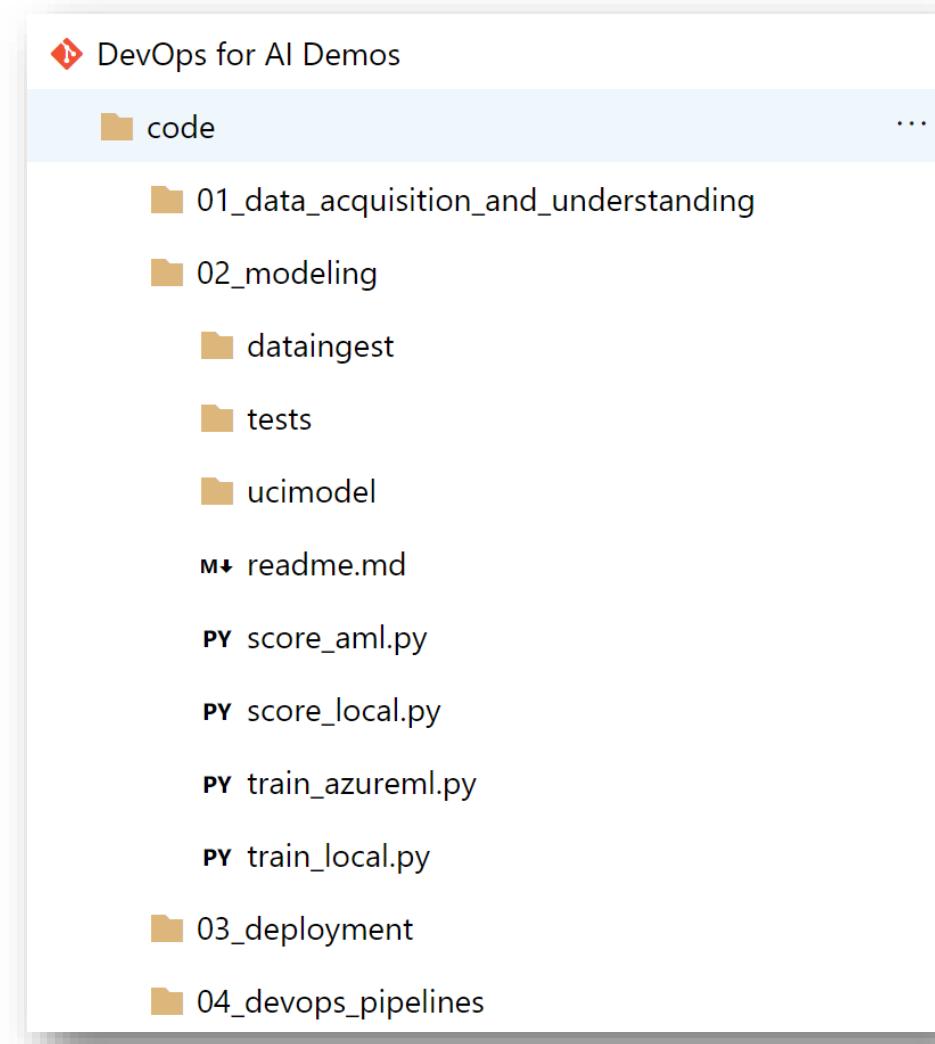
- Machine learning (ML) pipelines are used by data scientists to build, optimize, and manage their machine learning workflows.
- A typical pipeline involves a sequence of steps that cover the following areas:
  - Data preparation, such as normalizations and transformations
  - Model training, such as hyper parameter tuning and validation
  - Model deployment and evaluation
- They provide the job management infra we need for batch prediction pipelines

# Why use ML pipelines?

Key advantage	Description
Unattended runs	Schedule a few steps to run in parallel or in sequence in a reliable and unattended manner. Since data prep and modeling can last days or weeks, you can now focus on other tasks while your pipeline is running.
Mixed and diverse compute	Use multiple pipelines that are reliably coordinated across heterogeneous and scalable computes and storages. Individual pipeline steps can be run on different compute targets, such as HDInsight, GPU Data Science VMs, and Databricks, to make efficient use of available compute options.
Reusability	Pipelines can be templated for specific scenarios such as retraining and batch scoring. They can be triggered from external systems via simple REST calls.
Tracking and versioning	Instead of manually tracking data and result paths as you iterate, use the pipelines SDK to explicitly name and version your data sources, inputs, and outputs as well as manage scripts and data separately for increased productivity.

# Model Training

- **Training script** (`train.py`): The training script contains logic specific to the model that you are training.
- **Scoring file** (`score.py`): When the model is deployed as a web service, the scoring file receives data from clients and scores it against the model. The output is then returned to the client.
- **Experiment settings** (`project.json`): Links your project (`training.py` and other files required for training) with an experiment in your workspace.
- **RunConfig settings** (`myconfig.runconfig`): Defines how the training script is ran on the compute target that is used for training.
- **Conda environment** (`conda_dependencies.yml`): Defines the packages needed to run the training script.
- **Deployment environment** (`prod_dependencies.yml`): Defines the packages needed to run the model and scoring file in the deployment environment.



# Model Training

CI pipeline captures:

1. Create sandbox
2. Run unit tests and code quality checks
3. Attach to compute
4. Train model
5. Evaluate model
6. Register model

The screenshot shows a pipeline configuration in Azure DevOps. The title bar reads "DevOps for AI - Build Model - AzureML". Below the title, there are tabs for "Tasks", "Variables", "Triggers", "Options", "Retention", and "History". On the right side, there are "Save & queue" and "Delete" buttons. The main area displays 14 tasks:

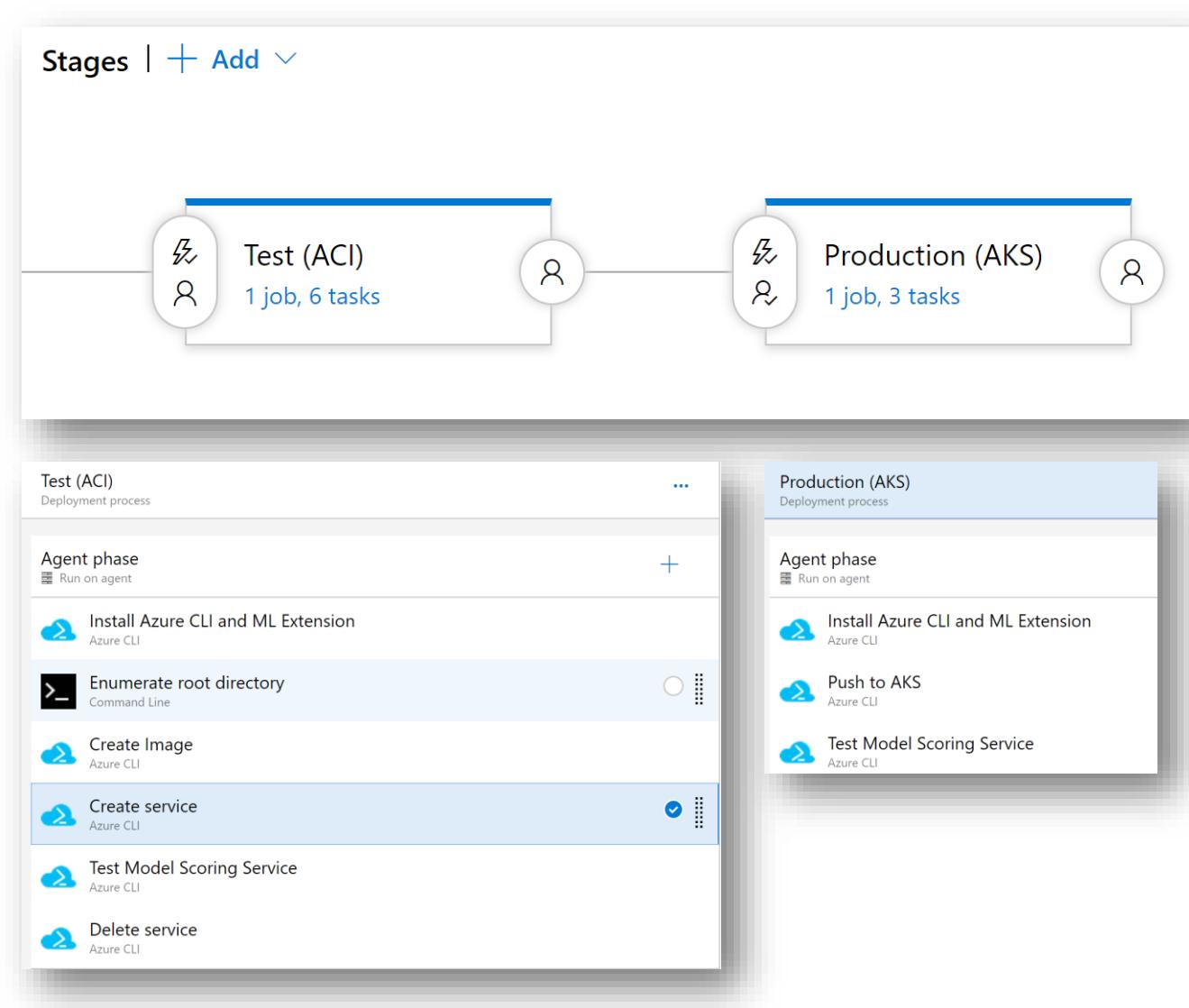
- Create Conda Environment (Conda Environment)
- Prepare Conda Environment (using yml) (Command Line)
- Install Azure CLI ML Extension (Preview) (Azure CLI)
- Unit tests (model code) (Command Line)
- Code Quality (flake8) (Command Line)
- Publish Unit Test Results (Publish Test Results)
- AzureML: Attach to an experiment (Azure CLI)
- AzureML: create dsvm run config (Azure CLI)
- AzureML: train model (Azure CLI)
- AzureML: download trained model (Azure CLI)
- Integration tests (Command Line)
- Publish Test Results (Publish Test Results)
- AzureML: register model (Azure CLI)
- Copy Model Requirements (Copy Files)

# Model Deployment

CD pipeline captures:

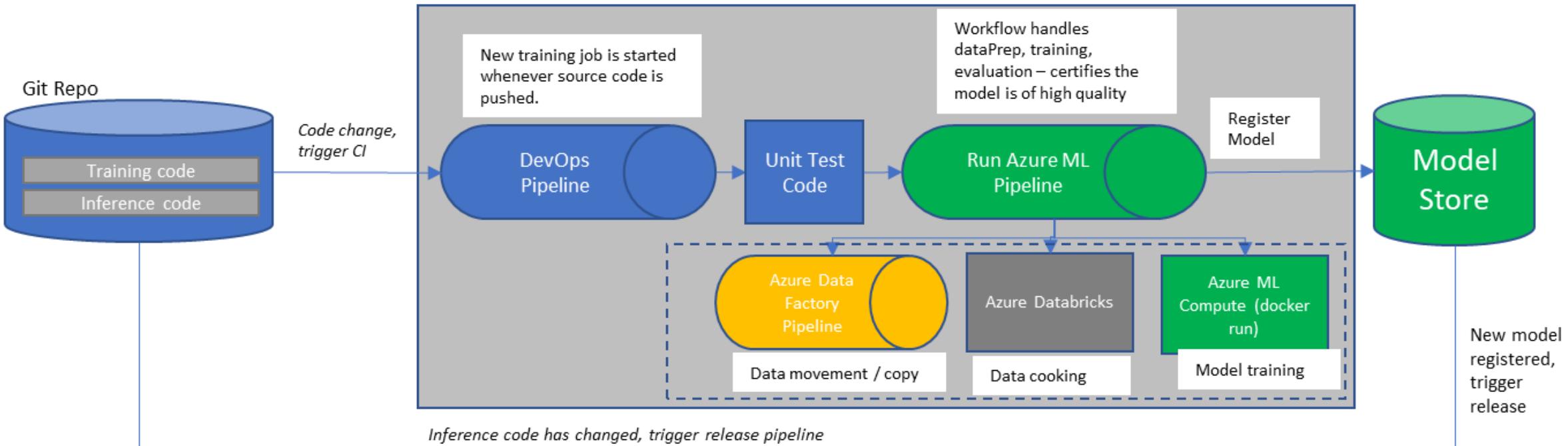
1. Package model into container image
2. Validate and profile model
3. Deploy model to Dev/Test (ACI)
4. If all is well, proceed to rollout to AKS

Everything is done via the CLI

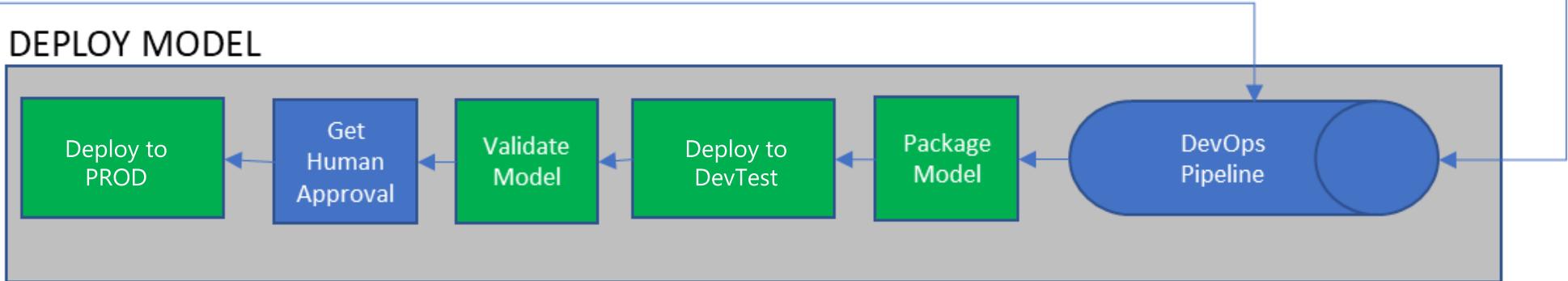


# Model CI/CD Pipeline

## TRAIN MODEL



## DEPLOY MODEL



Azure DevOps

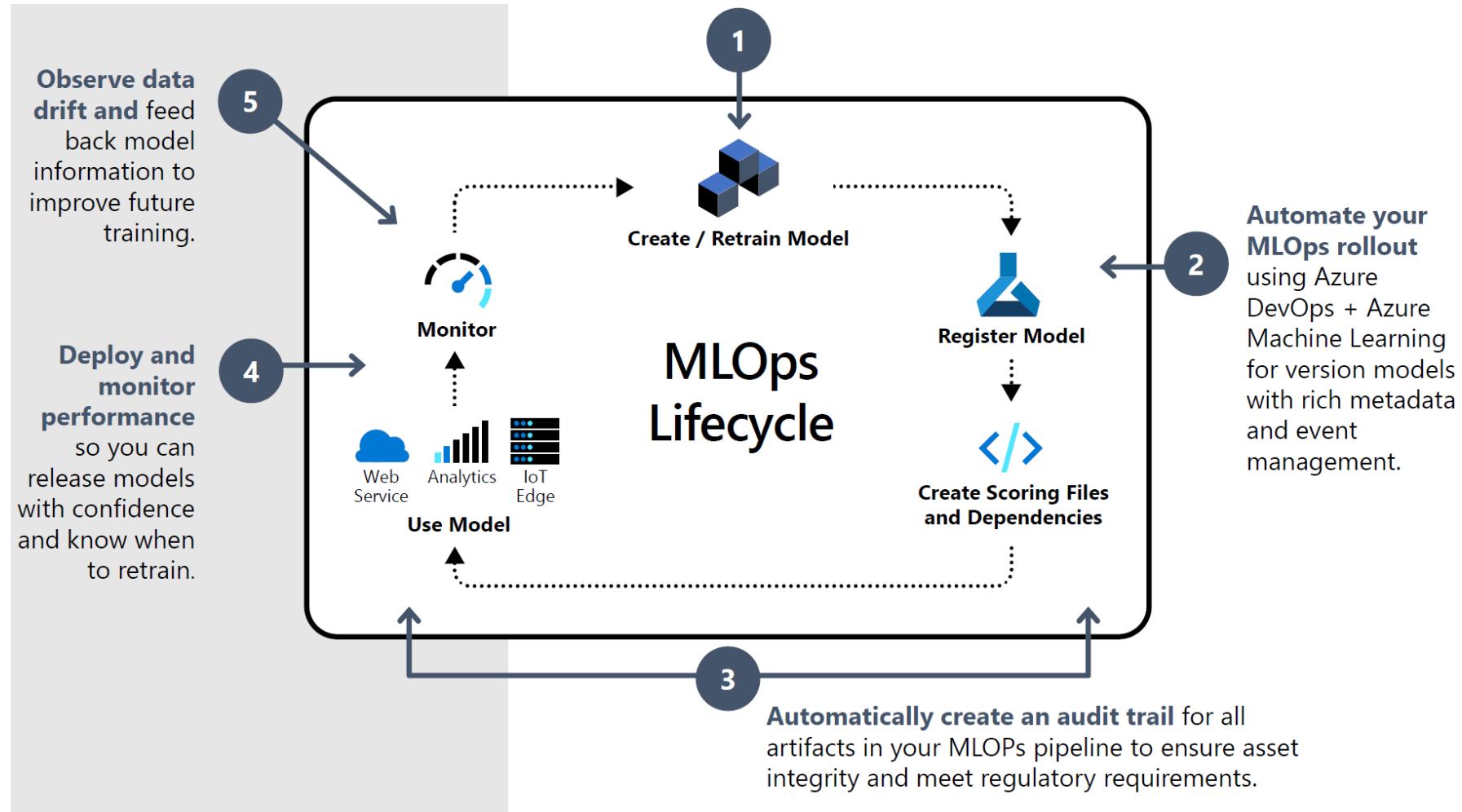
Azure Machine Learning

Azure Data Factory

# MLOps workshop

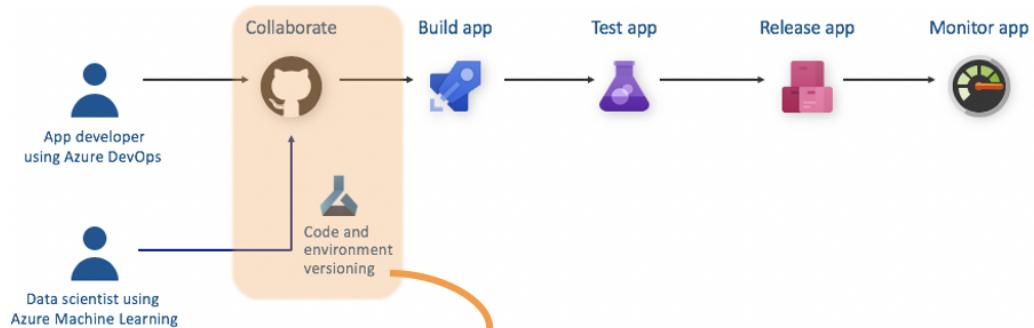


# MLOps



# MLOps

**Overall - looks complicated..**  
**As a data scientist I need to do what, now?**

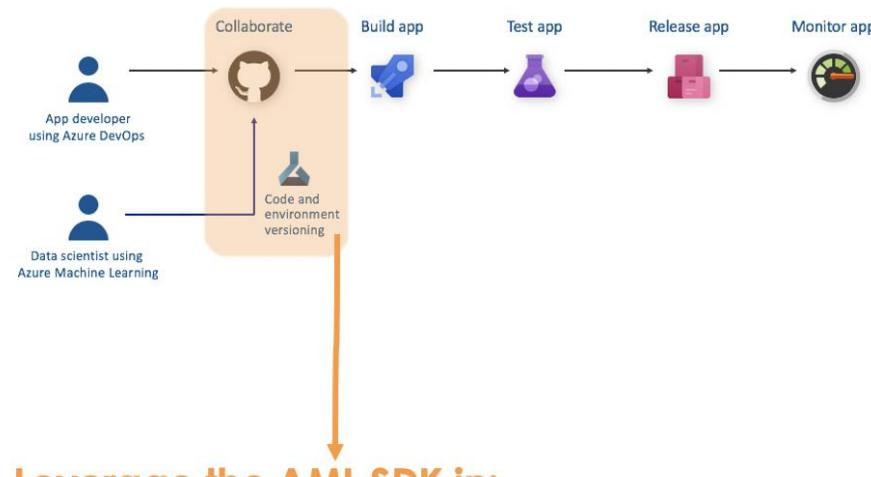


Develop and check this into git....  
**train.py**  
**containerize.py**  
**score.py**

# MLOps

As a data scientist...

**What's additional in my regular code?**



At a high-level...  
In the lab, review the source code



**Leverage the AML SDK in:**

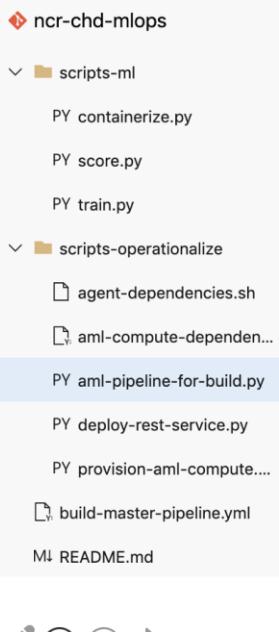
- train.py:** Provision transient compute, push training to it, and register the trained model to AML Model Registry
- containerize.py:** Create a docker container image and register into AML container registry
- score.py:** Input and output schema, scoring REST service, call log model metrics

# MLOps

As a data scientist, I tend to be curious...  
What else is needed for this MLOps?..

A whole bunch....lets review

## More code



## A "Build" pipeline

```
1 trigger:  
2 - master  
3  
4 pool:  
5 | vmImage: 'Ubuntu-16.04'  
6  
7 variables:  
8 | resourceGroup: 'ncr-mlops-rg'  
9 | amlWorkspace: 'ncramlws'  
10 | experiment: 'chd-prediction'  
11 | amlComputeTargetName: 'chd-temp-compute'  
12 | modelName: 'chd-predictor'  
13 | containerImageName: 'chd-predictor-image'  
14 | serviceConnection: 'ncrMlopsServiceConexion'  
15  
16 steps:  
17  
18 - task: UsePythonVersion@0  
19 | displayName: 'Build step 1 – Python configuration for agent'  
20 | inputs:  
21 | | versionSpec: '3.6'  
22 | | architecture: 'x64'  
23  
24 - task: Bash@3  
25 | displayName: 'Build step 2 – Install dependencies on agent'  
26 | inputs:  
27 | | targetType: filePath  
28 | | filePath: 'scripts-operationalize/agent-dependencies.sh'
```

## A "Release" pipeline

The screenshot shows a pipeline interface with the following details:

Pipeline: chd-release-pipeline > Release-5

Stages:

- Release**: Continuous deployment for Anagha Khanolkar on 1/28/2020, 11:08 PM.
- Stages**: Deploy REST Service succeeded with 1 warning on 1/28/2020, 11:29 PM.

Artifacts:

- \_chd-build-pipeli... 20200129.6

## Contenu du workshop

<https://aka.ms/WorkshopMLOPS>

# Workshop

# Azure ML



# Création d'un workspace Azure ML

Main \* Tags Review \*

Workspace Name \*

 ✓

Subscription

 ▼

Resource group

 ▼

Create new

Location

 ▼

Workspace edition [View full pricing details](#) ⓘ

 ▼

**ⓘ** For your convenience, these resources are added automatically to the workspace, if regionally available: [Azure storage](#), [Azure Application Insights](#) and [Azure Key Vault](#).

# Création en cours

Home > Microsoft.MachineLearningServices | Overview

## Microsoft.MachineLearningServices | Overview

Deployment

Search (Ctrl+ /) <>

Delete Cancel Redeploy Refresh

Overview Inputs Outputs Template

■■■ Your deployment is underway

Deployment name: Microsoft.MachineLearningServices  
Subscription: Microsoft Azure Internal Consumption  
Resource group: MLOpsWorkshopRG

Start time: 12/03/2020 à 12:02:53  
Correlation ID: 313c2f8b-3bc3-47b9-b548-0a2a60d51f8b

^ Deployment details (Download)

Resource	Type	Status	Operation details
No results.			

# Création terminée

Home > Microsoft.MachineLearningServices | Overview

## Microsoft.MachineLearningServices | Overview

Deployment

Search (Ctrl+ /) <<

Delete Cancel Redeploy Refresh

Overview

Inputs

Outputs

Template

Your deployment is complete

Deployment name: Microsoft.MachineLearningServices  
Subscription: Microsoft Azure Internal Consumption  
Resource group: MLOpsWorkshopRG

Start time: 12/03/2020 à 12:02:53  
Correlation ID: 313c2f8b-3bc3-47b9-b

Deployment details (Download)

Next steps

Go to resource

# Accès au workspace Azure ML

Search (Ctrl +/)

Download config.json Delete

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Events

Assets

Experiments

Pipelines

Compute

Models

Images

Deployments

Activities

Settings

Properties

Locks

Export template

Monitoring

Workspace edition : Enterprise

Resource group : MLOpsWorkshopRG

Location : West Europe

Subscription : Microsoft Azure Internal Consumption

Subscription ID : [REDACTED]

Storage : mlopsworkshop6119741507

Registry : ...

Key Vault : mlopsworkshop7771179297

Application Insights : mlopsworkshop9122234384

Try the new Azure Machine Learning studio

Introducing a new immersive experience (preview) for managing the end-to-end machine learning lifecycle.

Launch now Learn more

Getting Started

View Documentation

Learn how to use Azure Machine Learning.

View more samples at GitHub

Get inspired by a large collection of machine learning examples.

View Forum

Learn about Enterprise Edition

# Azure ML Studio

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Home

Welcome to the studio!

+ Create new

Notebooks

Code with Python SDK and run sample experiments.

Start now

Automated ML

Automatically train and tune a model using a target metric.

Start now

Designer

Drag-and-drop interface from prepping data to deploying models.

Start now

Tutorials

What is Azure Machine Learning?

Train your first ML model with Notebook

Create, explore and deploy Automated ML experiments.

The screenshot shows the Microsoft Azure Machine Learning Studio interface. At the top, there's a navigation bar with 'Preview' and 'Microsoft Azure Machine Learning'. Below it is a breadcrumb trail 'MLOpsWorkshop > Home'. The main area has a title 'Welcome to the studio!' and several cards: one for 'Create new' (with a plus sign icon), 'Notebooks' (with a calendar icon), 'Automated ML' (with a lightning bolt icon), and 'Designer' (with a three-tiered cube icon). Each card has a brief description and a 'Start now' button. Below this is a 'Tutorials' section with three items: 'What is Azure Machine Learning?' (with a flask icon), 'Train your first ML model with Notebook' (with a notebook icon), and 'Create, explore and deploy Automated ML experiments.' (with a gear icon). On the left, there's a sidebar with a menu: 'New' (selected), 'Home' (selected), 'Author' (disabled), 'Notebooks', 'Automated ML', 'Designer', 'Assets' (disabled), 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage' (disabled), and 'Compute' (disabled).

# Création d'une compute instance

Compute

Aining clusters    Inference clusters    Attached compute

Using the Notebook VM. You cannot create new Notebook VMs, but you can still use existing Notebook VMs. [Learn More.](#)

Start  Stop  Restart  Delete  Show created by me only

Status	Application URI	Virtual Machine size
--------	-----------------	----------------------



## New Compute Instance

Compute name \* [i](#) [e](#)

instancenotebook

Region \* [i](#)

westeurope

Virtual Machine size \* [i](#) [e](#)

Standard\_D4\_v2

Enable SSH access [?](#)

[>](#) Advanced settings

# Création en cours de la compute instance

MLOpsWorkshop > Compute

## Compute

[Compute Instances](#)   [Training clusters](#)   [Inference clusters](#)   [Attached compute](#)

i Compute instance is replacing the Notebook VM. You cannot create new Notebook VMs, but you can still use existing Notebook VMs. [Learn More.](#)

[+ New](#) [⟳ Refresh](#) [⟳ Start](#) [ⓧ Stop](#) [⟳ Restart](#) [Delete](#) [>Show created by me only](#)

Name	Status	Application URI	Virtual Machine size	Created on
instancenotebook	<span>○</span> Creating	JupyterLab Jupyter RStudio SSH	STANDARD_D4_V2	Mar 12, 2020 12:06 PM

< Prev Next >

# Compute instance disponible

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Compute

## Compute

Compute Instances Training clusters Inference clusters Attached compute

Compute instance is replacing the Notebook VM. You cannot create new Notebook VMs, but you can still use existing Notebook VMs. [Learn More.](#)

+ New Refresh Start Stop Restart Delete Show created by me only

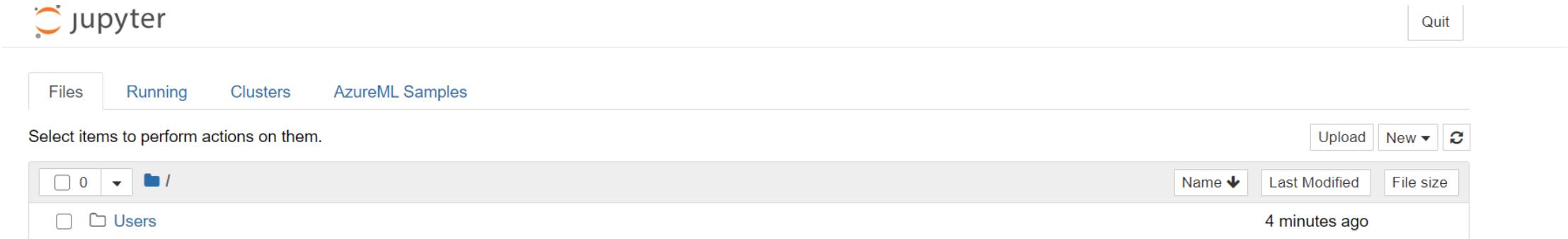
Name	Status	Application URI	Virtual Machine size	Created on
instancenotebook	Running	JupyterLab Jupyter RStudio SSH	STANDARD_D4_V2	Mar 12, 2020 12:06 PM

< Prev Next >

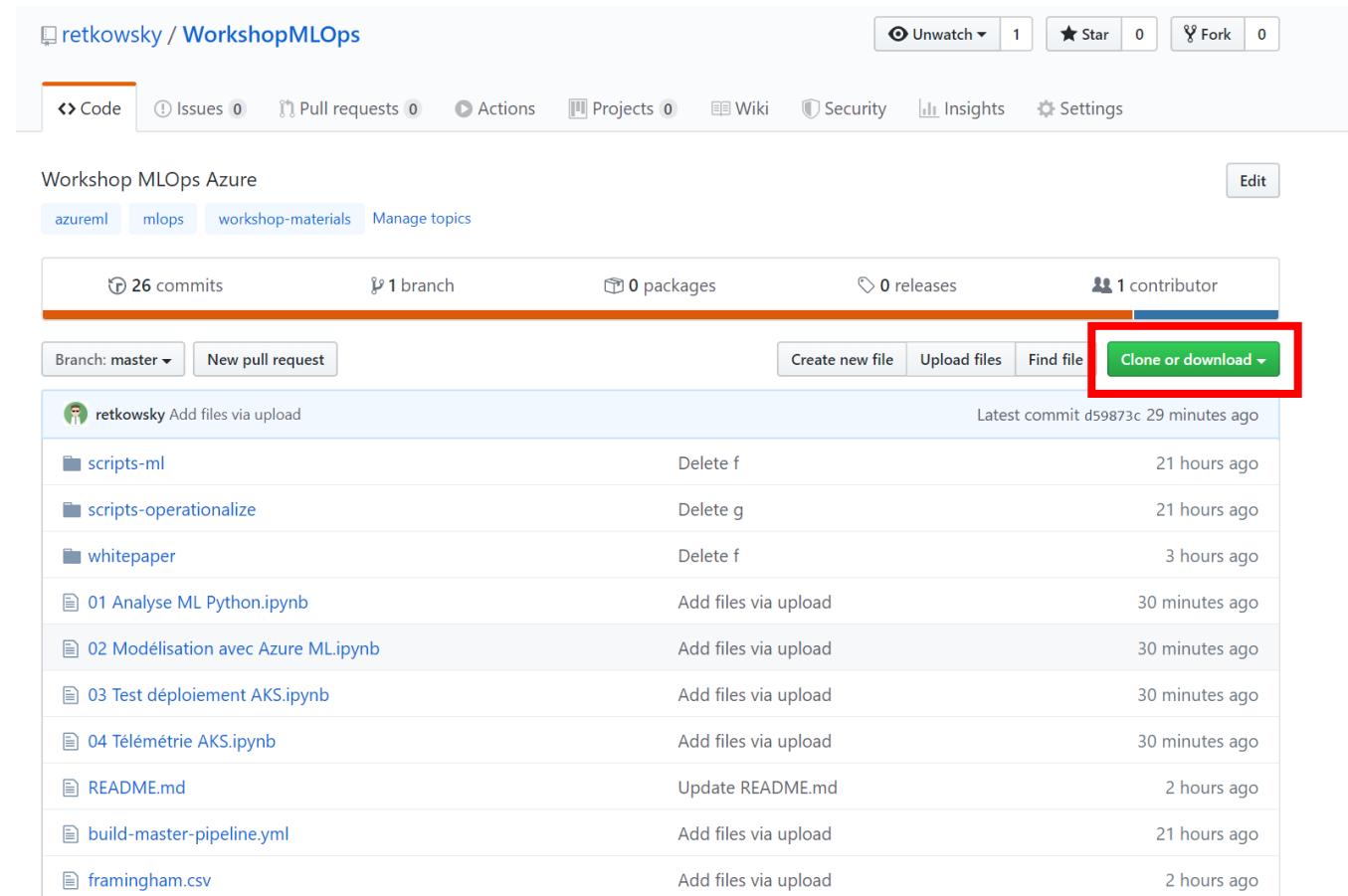
≡ New Home Author Notebooks Automated ML Designer Assets Datasets Experiments Pipelines Models Endpoints Manage Compute Databases

The screenshot shows the Microsoft Azure Machine Learning Compute Instances page. On the left is a navigation sidebar with various options like New, Home, Notebooks, and Compute (which is selected and highlighted in grey). The main content area has a blue header bar with 'Preview' and 'Microsoft Azure Machine Learning'. Below the header, the path 'MLOpsWorkshop > Compute' is shown. The main title is 'Compute' with tabs for 'Compute Instances', 'Training clusters', 'Inference clusters', and 'Attached compute'. A note says 'Compute instance is replacing the Notebook VM. You cannot create new Notebook VMs, but you can still use existing Notebook VMs. [Learn More.](#)'. Below the note are buttons for '+ New', 'Refresh', 'Start', 'Stop', 'Restart', 'Delete', and a toggle for 'Show created by me only'. A table lists the instances: one row for 'instancenotebook' which is 'Running' and associated with 'JupyterLab Jupyter RStudio SSH' under the 'Application URI' column, and a 'Virtual Machine size' of 'STANDARD\_D4\_V2'. The 'Created on' timestamp is 'Mar 12, 2020 12:06 PM'. The entire row for 'instancenotebook' is highlighted with a red box. At the bottom of the table are navigation arrows for 'Prev' and 'Next'.

# Accès à Jupyter notebooks



# Téléchargement du repo pour le workshop

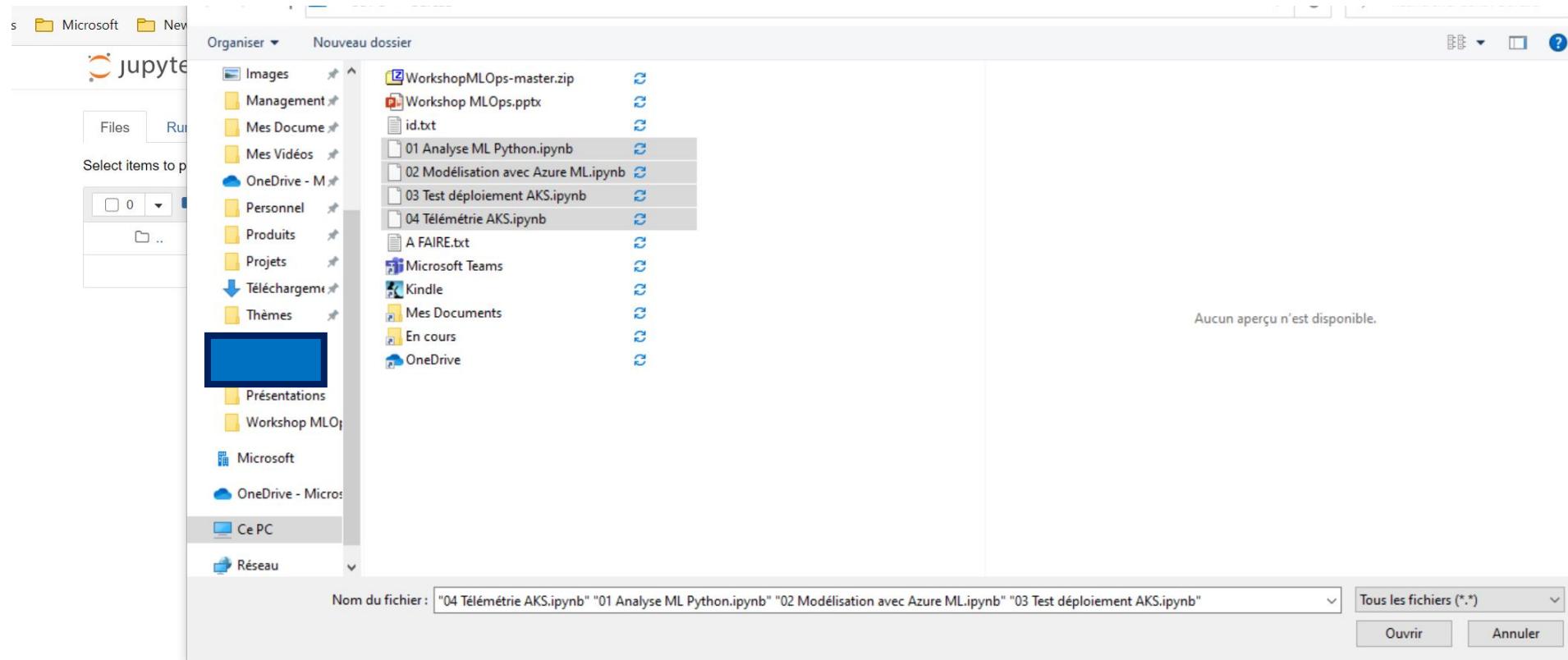


<https://github.com/retkowsky/WorkshopMLOps>

# Fichiers du repo

Nom	Taille	Com
📁 ..		
📁 scripts-ml	15 319	
📁 scripts-operationalize	16 185	
📁 whitepaper	572 793	3
📄 01 Analyse ML Python.ipynb	50 825	
📄 02 Modélisation avec Azure ML.ipynb	171 970	
📄 03 Test déploiement AKS.ipynb	3 091	
📄 04 Télémétrie AKS.ipynb	8 601	
📝 build-master-pipeline.yml	2 771	
📄 framingham.csv	191 803	
📝 README.md	317	

# Importation des notebooks dans Jupyter



# Chargement des notebooks dans Jupyter

The screenshot shows the Jupyter Notebook interface. At the top, there is a navigation bar with tabs: Files (selected), Running, Clusters, and AzureML Samples. On the far right of the header is a "Quit" button. Below the header, there is a toolbar with buttons for Upload, New (with a dropdown arrow), and a refresh icon. A message "Select items to perform actions on them." is displayed above the file list. The file list itself shows a directory structure under "/ Users / seretkow". It contains five entries: "04 Télémétrie AKS.ipynb", "03 Test déploiement AKS.ipynb", "02 Modélisation avec Azure ML.ipynb", a folder named "..", and "01 Analyse ML Python.ipynb". Each entry has an "Upload" and a "Cancel" button to its right. The text "The notebook list is empty." is centered above the list of files. A timestamp "seconds ago" is visible near the bottom right of the list area.

File Path	Name	Action
/ Users / seretkow		
	04 Télémétrie AKS.ipynb	Upload Cancel
	03 Test déploiement AKS.ipynb	Upload Cancel
	02 Modélisation avec Azure ML.ipynb	Upload Cancel
	..	seconds ago
	01 Analyse ML Python.ipynb	Upload Cancel

# Visualisation des notebooks importés

The screenshot shows the Jupyter Notebook interface. At the top, there's a navigation bar with tabs: 'Files' (selected), 'Running', 'Clusters', and 'AzureML Samples'. On the far right of the header are 'Quit', 'Upload', 'New ▾', and a refresh icon. Below the header, a message says 'Select items to perform actions on them.' To the left is a file browser showing the path 'Users / seretkow'. It lists four notebooks: '01 Analyse ML Python.ipynb', '02 Modélisation avec Azure ML.ipynb', '03 Test déploiement AKS.ipynb', and '04 Télémétrie AKS.ipynb'. Each entry includes a checkbox, the file name, the last modified time ('seconds ago'), and the file size. The columns are labeled 'Name' with a downward arrow, 'Last Modified', and 'File size'.

	Name	Last Modified	File size
<input type="checkbox"/> 0	..	seconds ago	
<input type="checkbox"/>	01 Analyse ML Python.ipynb	seconds ago	50.8 kB
<input type="checkbox"/>	02 Modélisation avec Azure ML.ipynb	seconds ago	172 kB
<input type="checkbox"/>	03 Test déploiement AKS.ipynb	seconds ago	3.09 kB
<input type="checkbox"/>	04 Télémétrie AKS.ipynb	seconds ago	8.6 kB

# Workshop Notebooks Azure ML



# Notebook 1. Analyse ML Python

Ce notebook contient du code python pour construire un modèle de ML.

The screenshot shows a Jupyter Notebook interface with the following details:

- Title Bar:** jupyter 01 Analyse ML Python Last Checkpoint: a few seconds ago (autosaved)
- Toolbar:** File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Not Trusted, Python 3
- Header:** nbviewer
- Section 1: 1. Analyse ML Python**
  - Diagram:** A flowchart illustrating the Machine Learning Pipeline:
    - Prepare:** Shows icons for databases and code.
    - Experiment:** Shows icons for Jupyter, CODE, and an ML service visual interface.
    - Deploy (MLOps):** Shows icons for training, deployment, and monitoring.
  - Sub-sections:** Prep data > Build model / feature engineering > Train & test model > Register & manage model > Package & validate > Deploy & monitor
- Section 2: 1. Informations**
  - In [1]:**

```
import sys  
sys.version
```
  - Out[1]:** '3.6.9 |Anaconda, Inc.| (default, Jul 30 2019, 19:07:31) \n[GCC 7.3.0]'
  - In [2]:**

```
import datetime  
now = datetime.datetime.now()
```

# Notebook 2. Analyse ML avec Azure ML

Ce notebook réalise les mêmes instructions que le notebook précédent mais en exploitant cette fois Azure ML.

Une analyse autoML sera également réalisée.

The screenshot shows a Jupyter Notebook interface with the title "jupyter 02 Modélisation avec Azure ML Last Checkpoint: 3 minutes ago (unsaved changes)". The toolbar includes File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Not Trusted, Python 3.6 - AzureML, and nbdiff. Below the toolbar is a toolbar with icons for file operations, Run, and cell types. The main content area has a header "2. Analyse ML avec Azure ML". Below it is a diagram illustrating the Azure ML workflow:

```
graph LR; subgraph Prepare; A[Prep data]; end --> subgraph Experiment; B[Build model / feature engineering]; end --> subgraph Deploy["Deploy (MLOps)"]; C[Deploy & monitor]; end; A --> B --> C;
```

The diagram shows a flow from "Prepare" (with icons for databases and storage), through "Experiment" (with icons for jupyter, code editor, and ML service visual interface), and finally "Deploy (MLOps)" (with icons for cloud and upload). Below the diagram, the steps are listed: Prep data > Build model / feature engineering > Train & test model > Register & manage model > Package & validate > Deploy & monitor.

**Données Framingham**  
<https://www.kaggle.com/amanaJmera1/framingham-heart-study-dataset>

Attributes/columns:

- male: 0 = Female; 1 = Male
- age: Age at exam time
- education: 1 = Some High School; 2 = High School or GED; 3 = Some College or Vocational School; 4 = college

# Modifier les champs suivants avec les infos du workspace Azure ML

```
import azureml.core
print("Version Azure ML service :", azureml.core.VERSION)

Version Azure ML service : 1.0.83

In [4]: # Azure subscription
subscription_id = "ARENSEIGNER"

# Resource Group
resource_group = "MLOpsworkshopRG"

# Workspace Name and Azure Region of the Azure Machine Learning Workspace
workspace_name = "MLOpsWorkshop"
workspace_region = "westeurope"

# Other variables
experiment_name = 'chd-prediction-manual'
project_dir = './chd'
deployment_dir = './deploy'
model_name = 'chd-predictor-manual'
model_description = 'Model to predict coronary heart disease'

# AML managed compute to be spun up for training
vm_name = "chd-manual"
```

# Authentification

## 2. Workspace Azure ML

```
In [*]: ws = Workspace.create(  
    name = workspace_name,  
    subscription_id = subscription_id,  
    resource_group = resource_group,  
    location = workspace_region,  
    exist_ok = True) #Leverage existing  
  
ws.write_config()  
print('Workspace configuration succeeded')
```

Performing interactive authentication. Please follow the instructions on the terminal.

To sign in, use a web browser to open the page <https://microsoft.com/devicelogin> and enter the code CC7M2DDJZ to authenticate.

# Exécution en cours du notebook

**Widget pour suivre l'avancement du run**

In [21]: RunDetails(run).show()

Run Properties		Output Logs
Status	Running	<input type="text" value="logs/azureml/155_azureml.log"/> <input checked="" type="checkbox"/> Auto-switch
Compute Target Nodes		2020-03-12 11:22:48,704 azureml DEBUG Inputs:: kwargs: {'OutputCollection': True, 'snapshotProject': True, 'only_in_process_features': True, 'skip_track_logs_dir': True}, track_folders: None, deny_list: None, directories_to_watch: [] 2020-03-12 11:22:48,705 azureml.history._tracking.PythonWorkingDirectory DEBUG Execution target type: batchai 2020-03-12 11:22:48,710 azureml.history._tracking.PythonWorkingDirectory DEBUG Failed to import pyspark with error: No module named 'pyspark' 2020-03-12 11:22:48,710 azureml.history._tracking.PythonWorkingDirectory.workingdir DEBUG Pinning working directory for filesystems: ['pyfs'] 2020-03-12 11:22:48,949 azureml._base_sdk_common.user_agent DEBUG Fetching client info from /root/.azureml/clientinfo.json 2020-03-12 11:22:48,949 azureml._base_sdk_common.user_agent DEBUG Error loading client info: [Errno 2] No such file or directory: '/root/.azureml/clientinfo.json'
Start Time	12/03/2020 12:21:00	
Duration	0:01:50	
Run Id	chd-prediction-manual_1584012057_28c60248	
Arguments	N/A	

# Référencement du modèle

```
In [27]: ┌─ if run.get_status() == 'Completed':  
    print("Training completed successfully!")  
    model_run = run.register_model(model_name=model_name,  
                                    model_path='./outputs/model/chd-rf-model',  
                                    tags={"type": "classification", "description": model_description, "run_id": run.id})  
    print("Model registered with version number: ", model_run.version)  
else:  
    print("Training failed!")  
    Exception("Training failed!")
```

```
Training completed successfully!  
Model registered with version number: 1
```

# Visualisation du modèle référencé dans Azure ML

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Models

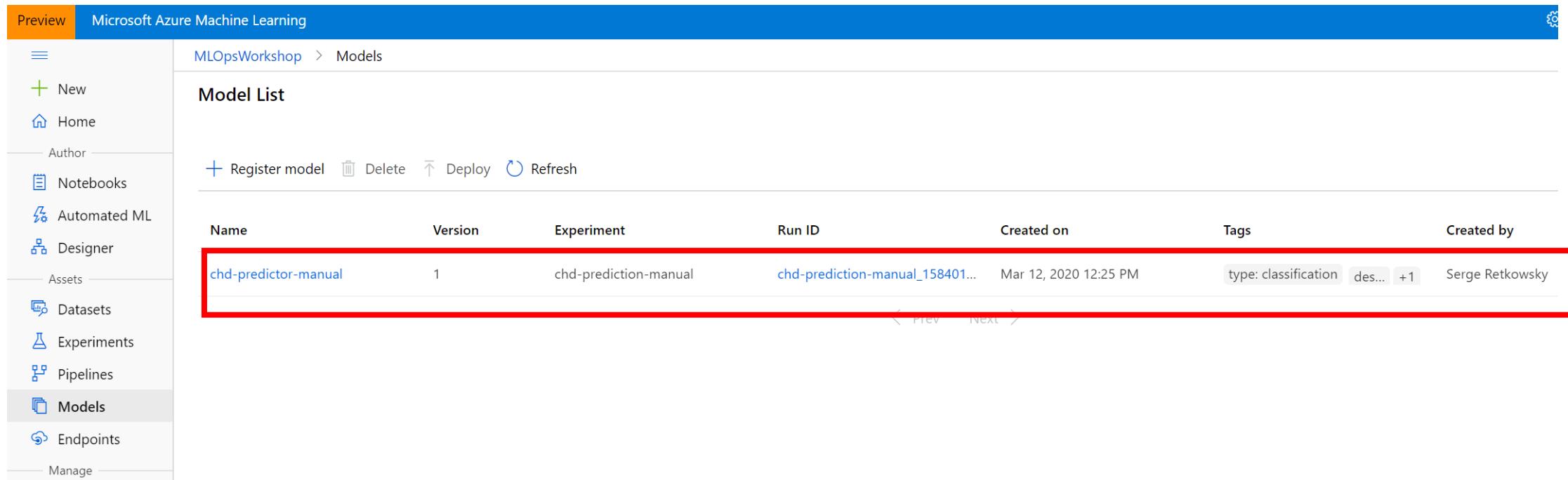
Model List

+ Register model Delete Deploy Refresh

Name	Version	Experiment	Run ID	Created on	Tags	Created by
chd-predictor-manual	1	chd-prediction-manual	chd-prediction-manual_158401...	Mar 12, 2020 12:25 PM	type: classification des... +1	Serge Retkowsky

Prev Next

New Home Author Notebooks Automated ML Designer Assets Datasets Experiments Pipelines Models Endpoints Manage



# Après exécution du notebook on peut visualiser le training cluster créé

The screenshot shows the Microsoft Azure Machine Learning interface. The top navigation bar has 'Preview' and 'Microsoft Azure Machine Learning' tabs. The left sidebar includes 'New', 'Home', 'Author' (Notebooks, Automated ML, Designer), 'Assets' (Datasets, Experiments, Pipelines, Models, Endpoints), and 'Manage' sections. The main content area is titled 'Compute' under 'MLOpsWorkshop > Compute'. It shows four tabs: 'Compute Instances', 'Training clusters' (which is selected and highlighted in blue), 'Inference clusters', and 'Attached compute'. Below these tabs are 'New', 'Refresh', and 'Delete' buttons. A table lists training clusters with columns: Name, Type, Provisioning state, and Created on. The first row, named 'chd-manual', is highlighted with a red box. Its details are: Type 'Machine Learning Com...', Provisioning state 'Succeeded (1 node)', and Created on '2020-03-12T11:18:12.4152503+00:00'. Navigation arrows for 'Prev' and 'Next' are at the bottom right of the table.

Name	Type	Provisioning state	Created on
chd-manual	Machine Learning Com...	Succeeded (1 node)	2020-03-12T11:18:12.4152503+00:00

# AutoML avec Azure ML

```
In [38]: automl_settings = {  
    "enable_early_stopping": True,  
    "iterations" : 10,                      #20 itérations max  
    "iteration_timeout_minutes" : 2,          #Temps max par itération  
    "experiment_timeout_minutes" : 10,         #Temps max pour le process AutoML  
    "n_cross_validations": 5,                 #Validation croisée  
    "primary_metric": 'accuracy',            #Choix de La métrique  
    "verbosity": logging.INFO  
}  
  
automl_config = AutoMLConfig(task = 'classification',  
                             debug_log = 'automl.log',  
                             preprocess=True,  
                             training_data = df,  
                             label_column_name = targetvariable,  
                             **automl_settings  
)  
  
In [*]: automl_run = experiment.submit(automl_config, show_output = True)
```

<https://docs.microsoft.com/en-us/azure/machine-learning/concept-automated-ml>

# Résultats AutoML

```
In [41]: # Widget
from azureml.widgets import RunDetails
RunDetails(automl_run).show()
```

AutoML\_7d755811-d947-47a2-b8a4-f17e12df0a15:  
Status: Completed

Status -

0 1 2 3 4 5 6 7 8 9

Iteration	Pipeline	Iteration metric	Best metric	Status	Duration	Started
8	VotingEnsemble	0.84929245	0.84929245	Completed	0:00:16	Mar 12, 2020 12:29 PM
1	MaxAbsScaler, SGD	0.84764151	0.84764151	Completed	0:00:14	Mar 12, 2020 12:27 PM
4	MaxAbsScaler, SGD	0.84716981	0.84764151	Completed	0:00:13	Mar 12, 2020 12:28 PM
6	MaxAbsScaler, SGD	0.84457547	0.84764151	Completed	0:00:14	Mar 12, 2020 12:29 PM
0	MaxAbsScaler, LightGBM	0.84221698	0.84221698	Completed	0:00:19	Mar 12, 2020 12:27 PM
9	StackEnsemble	0.83632075	0.84929245	Completed	0:00:27	Mar 12, 2020 12:30 PM
5	MaxAbsScaler, RandomForest	0.66462264	0.84764151	Completed	0:00:15	Mar 12, 2020 12:28 PM
3	MaxAbsScaler, ExtremeRandomTrees	0.65896226	0.84764151	Completed	0:00:14	Mar 12, 2020 12:28 PM
7	MaxAbsScaler, RandomForest	0.64740566	0.84764151	Completed	0:00:14	Mar 12, 2020 12:29 PM
2	MaxAbsScaler, SGD	0.64292453	0.84764151	Completed	0:00:14	Mar 12, 2020 12:28 PM

10 ▾ per page

accuracy ▾

# Accès à l'expérimentation AutoML depuis Azure ML Studio

The screenshot shows the Microsoft Azure Machine Learning Studio interface. The top navigation bar includes 'Preview' (orange), 'Microsoft Azure Machine Learning', and various icons for settings, help, and user profile. The left sidebar has a 'New' button, 'Home', 'Author', 'Notebooks', 'Automated ML', 'Designer', 'Assets', 'Datasets', 'Experiments' (which is selected and highlighted in blue), 'Pipelines', 'Models', 'Endpoints', 'Manage', and 'Compute'. The main content area shows the 'Experiments' page for the 'MLOpsWorkshop' workspace. It features a header with 'Experiments', 'Refresh' (blue), 'Archive experiment', and a toggle for 'View archived experiments'. Below is a search bar with '+ Add filter'. A table lists experiments with columns: 'Experiment' (with 'chd-prediction-manual' highlighted and surrounded by a red box), 'Latest run' (2), 'Last submitted' (Mar 12, 2020 12:27 PM), 'Created' (Mar 12, 2020 12:18 PM), 'Created by' (Serge Retkowsky), and 'Run types' (Script, Automated ML). Navigation arrows for 'Prev' and 'Next' are at the bottom of the table.

Experiment	Latest run	Last submitted	Created	Created by	Run types
chd-prediction-manual	2	Mar 12, 2020 12:27 PM	Mar 12, 2020 12:18 PM	Serge Retkowsky	Script, Automated ML

# Résultats AutoML

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Experiments > chd-prediction-manual > Run 2

Run 2 ✓ Completed [Switch to old experience](#) (?)

⟳ Refresh ✖ Cancel

[Details](#) [Data guardrails](#) [Models](#) [Logs](#) [Outputs](#)

Data guardrails are run by Automated ML when automatic featurization is enabled. This is a sequence of checks over the input data to ensure high quality data is being used to train model.

Type	Status	Description	
Class balancing detection	passed	Classes are balanced in the training data.	<span style="color: green;">✓</span>
Missing values imputation	fixed	The training data had the following missing values which were resolved. Please review your data source for data quality issues and possibly filter out the rows with these missing values. If the missing values are expected, you can either accept the above imputation, or implement your own custom imputation that may be more appropriate based on the data type and business process.	<span style="color: green;">✓</span>
<a href="#">Additional details</a>			
High cardinality feature detection	passed	Your inputs were analyzed, and no high cardinality features were detected.	<span style="color: green;">✓</span>

# Résultats AutoML

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Experiments > chd-prediction-manual > Run 2

Run 2 Completed Switch ↗

⟳ Refresh ✖ Cancel

Details Data guardrails **Models** Logs Outputs Search to filter items...

Algorithm name	Accuracy ↓	Created	Duration	Status	Model
VotingEnsemble	0.8492924528301888	Mar 12, 2020 12:29 PM	17s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, SGD	0.8476415094339622	Mar 12, 2020 12:27 PM	14s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, SGD	0.8471698113207546	Mar 12, 2020 12:28 PM	14s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, SGD	0.8445754716981131	Mar 12, 2020 12:29 PM	14s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, LightGBM	0.8422169811320754	Mar 12, 2020 12:27 PM	19s	Completed	<span style="color: blue;">⬇️</span> Download
StackEnsemble	0.8363207547169811	Mar 12, 2020 12:30 PM	27s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, RandomForest	0.664622641509434	Mar 12, 2020 12:28 PM	15s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, ExtremeRandomTrees	0.6589622641509434	Mar 12, 2020 12:28 PM	14s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, RandomForest	0.6474056603773585	Mar 12, 2020 12:29 PM	14s	Completed	<span style="color: blue;">⬇️</span> Download
MaxAbsScaler, SGD	0.6420245283018868	Mar 12, 2020 12:28 PM	14s	Completed	<span style="color: blue;">⬇️</span> Download

# Résultats AutoML

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Experiments > chd-prediction-manual > Run 2 > Run 11

Run 11 Completed

Refresh Explain model Cancel

Model details Visualizations Explanations (preview) Logs Outputs

**Model summary**

Algorithm name  
VotingEnsemble

Accuracy  
0.8492924528301888

Registered models  
[AutoMLmodel:1](#)

Deploy status  
No deployment yet

**Run details**

Status  
Completed

Run ID  
AutoML\_7d755811-d947-47a2-b8a4-f17e12df0a15\_8

Input datasets  
--

Created time  
Mar 12, 2020 12:29 PM

Duration  
17s

**Run Metrics**

Accuracy  
0.84929

AUC macro  
0.69607

AUC micro

Deploy model Download model

# Résultats AutoML

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Experiments > chd-prediction-manual > Run 2 > Run 11

Run 11 Completed Switch to old experience

Refresh Explain model Cancel

Model details Visualizations Explanations (preview) Logs Outputs

Predicted Probability Percentile

Gain Curve

Percentile

Gain

Weighted Average  
Macro Average  
Micro Average  
Baseline  
0  
1

Confusion Matrix

Raw Predicted Label

True Label

0 0 3586  
1 1 629  
15

3500  
3000  
2500  
2000  
1500  
1000  
500

The screenshot displays the Microsoft Azure Machine Learning Studio interface. The top navigation bar includes 'Preview', 'Microsoft Azure Machine Learning', and various icons. The breadcrumb path indicates the current experiment: MLOpsWorkshop > Experiments > chd-prediction-manual > Run 2 > Run 11. The status 'Run 11 Completed' is shown, with a link to 'Switch to old experience'. Below the header are buttons for 'Refresh', 'Explain model', and 'Cancel'. The main content area has tabs for 'Model details', 'Visualizations' (which is selected), 'Explanations (preview)', 'Logs', and 'Outputs'. Under 'Visualizations', there are two charts: a 'Gain Curve' plot showing 'Predicted Probability' on the y-axis and 'Percentile' on the x-axis, and a 'Confusion Matrix' showing 'Predicted Label' on the top and 'True Label' on the left. The confusion matrix values are: True Label 0, Predicted Label 0: 3586; True Label 0, Predicted Label 1: 629; True Label 1, Predicted Label 0: 15; True Label 1, Predicted Label 1: 15. A color scale on the right of the confusion matrix ranges from 500 to 3500.

# Workshop Interfaces graphiques



# Interfaces graphiques



AutoML user interface

Pour la réalisation des analyses AutoML  
sans programmation Python



Designer

Pour la réalisation de vos analyses ML sans  
programmation Python

# AutoML User Interface

# AutoML user interface

Utilisation de l'interface graphique pour la réalisation du processus AutoML:

- 1. Chargement du dataset  
<https://raw.githubusercontent.com/retkowsky/WorkshopMLOps/master/framingham.csv>
- 2. Création AutoML depuis l'interface graphique (wizard)

# AutoML user interface

## Ajout source de données

The screenshot shows the Microsoft Azure Machine Learning Studio interface. The top navigation bar includes 'Preview' and 'Microsoft Azure Machine Learning'. The left sidebar has sections for 'Author' (Notebooks, Automated ML, Designer) and 'Assets' (Datasets, Experiments, Pipelines, Models, Endpoints). The 'Datasets' section is selected and highlighted with a red box. The main content area displays 'Registered datasets' and 'Dataset monitors (Preview)'. It features a 'Create dataset' button, a 'Refresh' button, and an 'Unregister' button. A table lists registered datasets with columns: Name, Version, Created on, Modified on, Properties, Created by, and Tags. One dataset, 'framingham', is listed with a red box around its row. At the bottom, there are navigation arrows for 'Prev' and 'Next'.

Name	Version	Created on	Modified on	Properties	Created by	Tags
framingham	1	Mar 16, 2020 2:19 PM	Mar 16, 2020 2:19 PM	Tabular	Serge Retkowsky	

# AutoML user interface

# Utilisation de l'interface graphique

MLOpsWorkshop > Automated ML > Start run

Create a new Automated ML run

Select dataset

## Select dataset

Select a dataset from the list below, or create a new dataset. Automated ML currently only supports tabular data for authoring runs.

+ Create dataset

Show supported datasets only

Search to filter items...

Dataset name

Dataset type

Created on

Modified

framingham

Tabular

Mar 16, 2020 2:19 PM

Mar 16, 2020 2:19 PM

< Prev Next >

# AutoML user interface

## Paramétrage

MLOpsWorkshop > Automated ML > Start run

Create a new Automated ML run

- Select dataset
- Configure run
- Task type and settings

### Configure run

Configure the experiment. Select from existing experiments or define a new name, select the target column and the training compute.

Dataset  
framingham ([View dataset](#))

Experiment name \*

AutoML-visualinterface

Target column \* ⓘ

TenYearCHD

Select training cluster \* ⓘ

chd-manual

[Create a new compute](#) [Refresh compute](#)

# AutoML user interface

MLOpsWorkshop > Automated ML > Run Detail

Run 1  Running

 Refresh  Cancel

[Details](#) [Data guardrails](#) [Models](#) [Logs](#) [Outputs](#)

## Run summary

### Task type

Classification  [View all run settings](#)

### Primary metric

Accuracy

### Run status

Running

### Experiment name

AutoML-visualinterface

### Run ID

AutoML\_9ef5e1e7-95d1-4329-a733-976d2075afb2

### Input datasets

Input name: input\_data, ID: [22785128-8a5c-431a-8889-4d57fdb37d91](#)

# AutoML user interface

## Meilleur modèle

The screenshot shows the Microsoft Azure Machine Learning interface for the 'Run Detail' page of 'Run 1'. The page is titled 'Run 1' and indicates it is 'Completed'. It includes a 'Refresh' and 'Cancel' button. The main content is divided into two sections: 'Model summary' and 'Run summary'.

**Model summary:**

- Algorithm name: VotingEnsemble
- Accuracy: 0.8492947717580371 [View all other metrics](#)
- Registered models: No registration yet
- Deploy status: No deployment yet

**Run summary:**

- Task type: Classification [View all run settings](#)
- Primary metric: Accuracy
- Run status: Completed
- Experiment name: AutoML-visualinterface
- Run ID: AutoML\_9ef5e1e7-95d1-4329-a733-976d2075afb2
- Input datasets: Input name: input\_data, ID: [22785128-8a5c-431a-8889-4d57fdb37d91](#)

At the bottom, there are three buttons: 'Deploy best model' (blue), 'View model details' (white), and 'Download best model' (white).

# AutoML user interface

## Résultats

The screenshot shows the Microsoft Azure Machine Learning Studio interface. The top navigation bar includes 'Preview' (orange), 'Microsoft Azure Machine Learning' (blue), and various icons for settings, help, and user profile.

The left sidebar has a tree view with nodes like 'New', 'Home', 'Author', 'Notebooks', 'Automated ML' (selected), 'Designer', 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Datastores', and 'Data Labeling'. The 'Automated ML' node is highlighted with a blue background.

The main content area shows 'Run 1' status as 'Completed' with a green checkmark. There are 'Refresh' and 'Cancel' buttons below it.

The 'Data guardrails' tab is selected, showing a table of validation results:

Type	Status	Description
Cross validation	done	Each iteration of the trained model was validated through cross-validation.
<a href="#">&gt; Additional details</a>		
Class balancing detection	passed	Classes are balanced in the training data.
<a href="#">&gt; Additional details</a>		
Missing values imputation	fixed	The training data had the following missing values which were resolved. Please review your data source for data quality issues and possibly filter out the rows with these missing values. If the missing values are expected, you can either accept the above imputation, or implement your own custom imputation that may be more appropriate based on the data type and business process.
<a href="#">&gt; Additional details</a>		
High cardinality feature detection	done	High cardinality inputs were detected in dataset and were featurized as {}.
<a href="#">&gt; Additional details</a>		

# AutoML user interface

## Liste des itérations

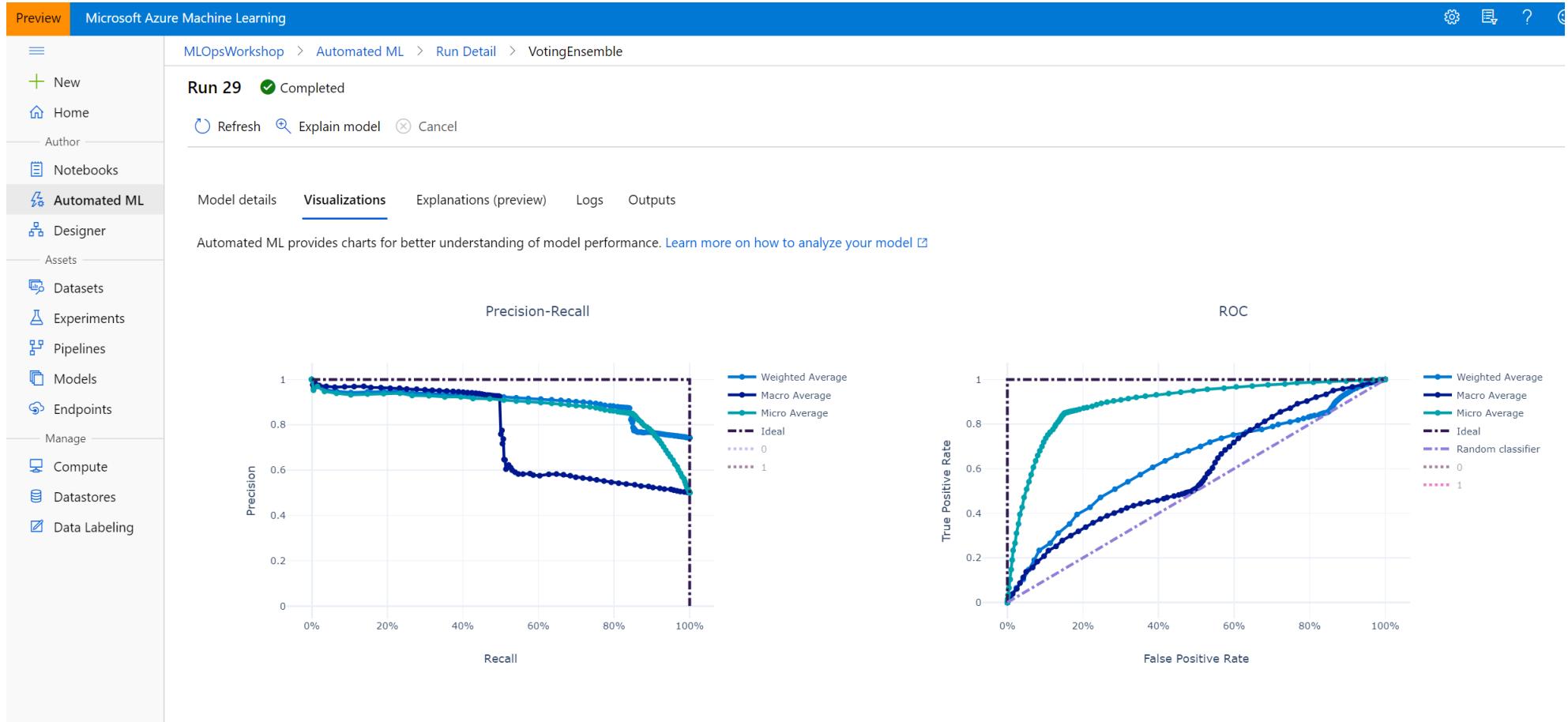
The screenshot shows the Microsoft Azure Machine Learning Studio interface. The top navigation bar includes 'Preview' (highlighted in orange), 'Microsoft Azure Machine Learning', and a user icon. The left sidebar has a 'New' button, 'Home' link, and sections for 'Author', 'Automated ML' (selected), 'Designer', 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Datastores', and 'Data Labeling'. The main content area shows 'Run 1' status as 'Completed'. It includes 'Refresh' and 'Cancel' buttons. Below this, tabs for 'Details', 'Data guardrails', 'Models' (selected), 'Logs', and 'Outputs' are present. A search bar at the top right says 'Search to filter items'. The 'Models' section displays a table with columns: Algorithm name, Accuracy ↓, Created, Duration, Status, and Model. Each row contains a download link. The table lists ten different models with their respective details.

Algorithm name	Accuracy ↓	Created	Duration	Status	Model
VotingEnsemble	0.8492947717580371	Mar 16, 2020 3:23 PM	1m 58s	Completed	<a href="#">Download</a>
MaxAbsScaler, SGD	0.8483511529800234	Mar 16, 2020 2:34 PM	1m 37s	Completed	<a href="#">Download</a>
MaxAbsScaler, SGD	0.8481152482855201	Mar 16, 2020 2:39 PM	1m 37s	Completed	<a href="#">Download</a>
MaxAbsScaler, ExtremeRandomTrees	0.8481152482855201	Mar 16, 2020 2:47 PM	1m 37s	Completed	<a href="#">Download</a>
MaxAbsScaler, ExtremeRandomTrees	0.8481152482855201	Mar 16, 2020 3:11 PM	1m 39s	Completed	<a href="#">Download</a>
MaxAbsScaler, SGD	0.8481152482855201	Mar 16, 2020 3:21 PM	1m 37s	Completed	<a href="#">Download</a>
MaxAbsScaler, SGD	0.8478795104260198	Mar 16, 2020 3:09 PM	1m 33s	Completed	<a href="#">Download</a>
MaxAbsScaler, XGBoostClassifier	0.8455199629759761	Mar 16, 2020 2:26 PM	1m 40s	Completed	<a href="#">Download</a>
StackEnsemble	0.8448125825624722	Mar 16, 2020 3:26 PM	1m 57s	Completed	<a href="#">Download</a>
MaxAbsScaler, LightGBM	0.8443407731734652	Mar 16, 2020 3:17 PM	1m 36s	Completed	<a href="#">Download</a>

Navigation buttons at the bottom include '< Prev' and 'Next >'.

# AutoML user interface

## Graphiques



# AutoML user interface

## Explanations

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Automated ML > Run Detail > VotingEnsemble

Run 29 Completed

Refresh Explain model Cancel

Model details Visualizations Explanations (preview) Logs Outputs

Model explanations are used to understand what features are directly impacting the model and why. [Learn more](#)

Select Explanation

tabular | mimic | engineered | classification | 2dd87d62-4d1e-4825-8b2d-9f0db2e3d284 | 3/16/2020, 3:40:54 PM

Explainer: mimic

Global Importance

Summary Importance

Chart type: Box Group by Predicted Y Top K Features: 8

Feature Importance

prevalenthyp\_ModeCatImputer male\_ModeCatImputer\_LabelEncoder education\_CharGramCountVectorizer currentSmoker\_ModeCatImputer sysBP\_MeanImputer age\_CharGramCountVectorizer education\_CharGramCountVectorizer age\_CharGramCountVectorizer

# Designer

# Designer

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Designer > Authoring

Visual Pipeline

Autosave on Search in canvas

framingham

Split Data

Two-Class Neural Network

Train Model

Score Model

Evaluate Model

Navigator

The screenshot shows the Microsoft Azure Machine Learning Designer interface. On the left, there's a navigation sidebar with sections like 'New', 'Home', 'Author', 'Designer' (which is selected), 'Assets', 'Datasets', 'Experiments', 'Pipelines', 'Models', 'Endpoints', 'Manage', 'Compute', 'Datastores', and 'Data Labeling'. The main area is titled 'Visual Pipeline' and shows a flowchart. It starts with a dataset node labeled 'framingham', which feeds into a 'Split Data' node. From 'Split Data', two paths emerge: one leading to a 'Two-Class Neural Network' node, and another leading to a 'Train Model' node. Both of these nodes have green checkmarks next to them. The output from 'Train Model' then flows into a 'Score Model' node, followed by an 'Evaluate Model' node, both also with green checkmarks. There are various icons along the top bar, including a search bar, autosave controls, and a 'Search in canvas' button.

# Déploiement du modèle visual Interface

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Designer > Authoring

Visual Pipeline

Submit Create inference pipeline ✓ Publish ...

Real-time inference pipeline

Batch inference pipeline

Settings

Default compute target ⓘ chd-manual Select compute target

Pipeline parameters +

No parameters selected

Draft details

Draft name Visual Pipeline

Draft description (optional) Pipeline created on 20200316

Created on March 16, 2020 2:42 PM

Created by Serge Retkowsky

Last edit time March 16, 2020 2:45 PM

Last edited by Serge Retkowsky

Search

New Home

Author

- Notebooks
- Automated ML
- Designer
- Assets
- Datasets
- Experiments
- Pipelines
- Models
- Endpoints

Manage

- Compute
- Datastores
- Data Labeling

Visual Pipeline

framingham

Split Data

Two-Class Neural Network

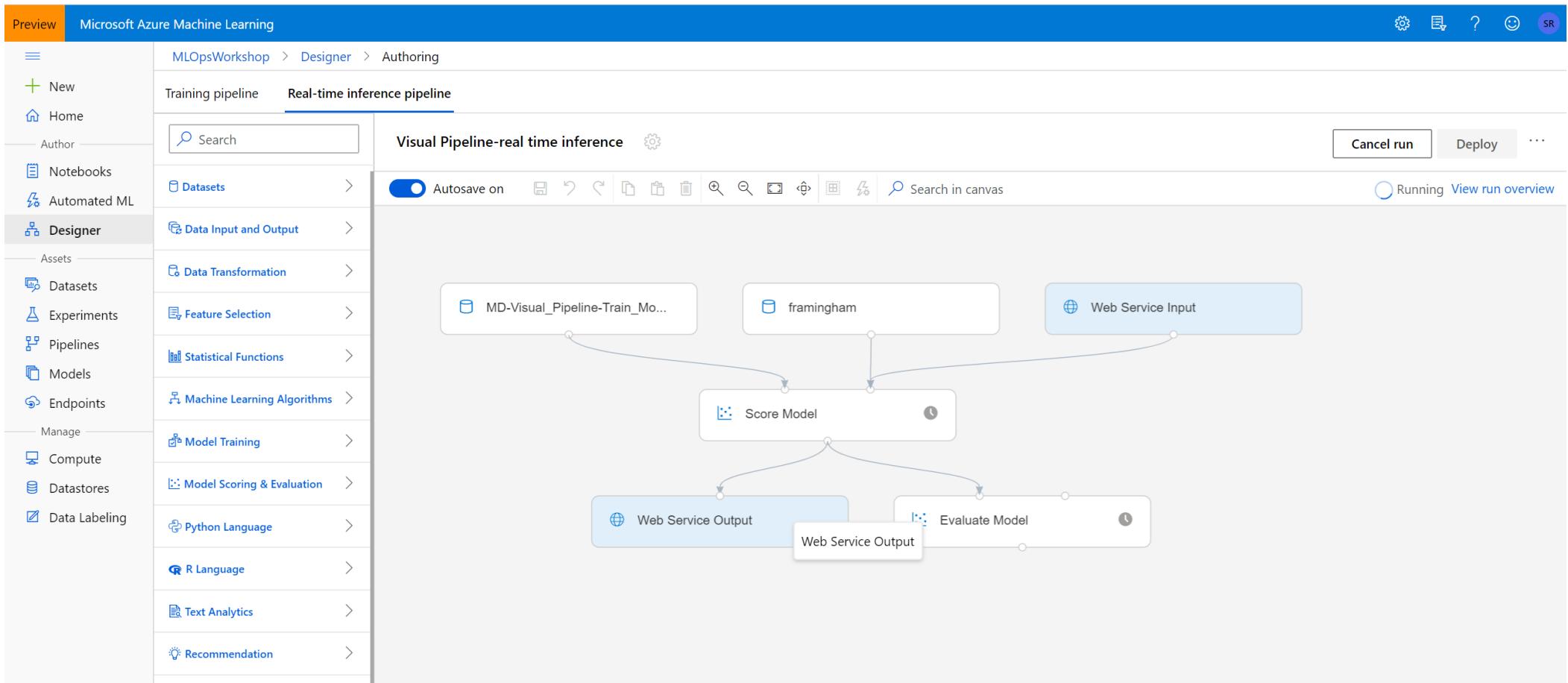
Train Model

Score Model

Evaluate Model

Navigator

# Création pipeline Real Time



# Création d'un nouveau AKS Inference cluster

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Compute

Compute

Compute Instances Training clusters **Inference clusters** Attached compute

+ New ⏪ Refresh ⏷ Delete ⏹ Detach

Name	Type	Created/Attached	Provisioning state	Created on
aksvizual	Kubernetes Service	Created	Creating	Mar 16, 2020 2:50 PM

The screenshot shows the Microsoft Azure Machine Learning interface. The top navigation bar has 'Preview' and 'Microsoft Azure Machine Learning'. Below it, the 'Compute' section is selected under 'MLOpsWorkshop > Compute'. On the left, there's a sidebar with 'New', 'Home', 'Author' (Notebooks, Automated ML, Designer), and 'Assets' (Datasets, Experiments). The main area shows tabs for 'Compute Instances', 'Training clusters', 'Inference clusters' (which is underlined in blue), and 'Attached compute'. Below the tabs are buttons for '+ New', 'Refresh', 'Delete', and 'Detach'. A table lists existing inference clusters: one named 'aksvizual' of type 'Kubernetes Service' was created and is currently in a 'Creating' state, with a timestamp of 'Mar 16, 2020 2:50 PM'. The row for 'aksvizual' is highlighted with a red box.

# Déploiement du pipeline

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Designer > Authoring

Training pipeline Real-time inference pipeline

Visual Pipeline-real time inference

Search in canvas

Submit Deploy

Run finished View run overview

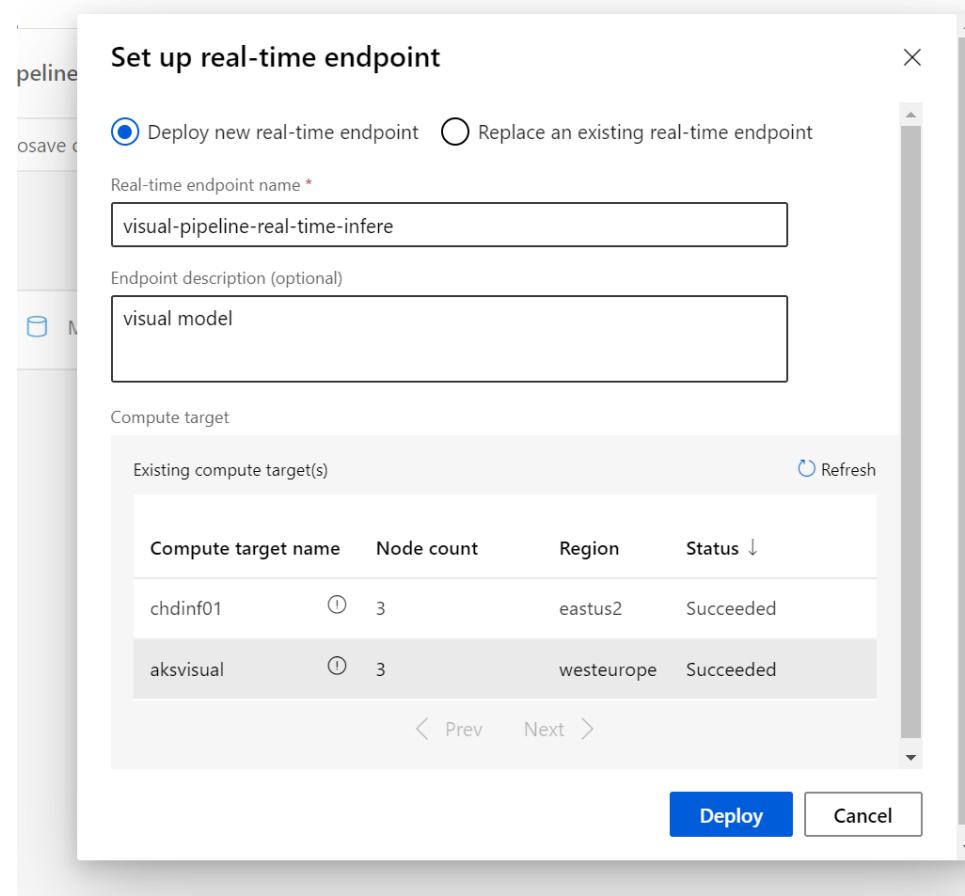
MD-Visual\_Pipeline-Train\_Mo... framingham Web Service Input

Score Model

Web Service Output Evaluate Model

```
graph TD; A[MD-Visual_Pipeline-Train_Mo...] --> B[framingham]; A --> C[Web Service Input]; B --> D[Score Model]; C --> D; D --> E[Web Service Output]; D --> F[Evaluate Model]
```

# Déploiement du pipeline



# Déploiement du pipeline

Preview Microsoft Azure Machine Learning 

New Home Author Notebooks Automated ML Designer Assets Datasets Experiments

MLOpsWorkshop > Endpoints

## Endpoints

Real-time endpoints Pipeline endpoints

Refresh Delete Search to filter items...

Name	Description	Created On	Created By	Updated On ↓	Compute Type	Compute Target
visual-pipeline-re...	visual model	March 16, 2020 3:16 PM	Serge Retkowsky	March 16, 2020 3:16 PM	AKS	aksvisual

# Visualisation du modèle déployé

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Endpoints > visual-pipeline-real-time-infere

## visual-pipeline-real-time-infere

Details Test Consume

Description	visual model	Scoring timeout	60000 ms
Deployment state	Healthy	CPU	0.1
Compute type	AKS	Memory	0.5 GB
Service ID	visual-pipeline-real-time-infere	Autoscale enabled	true
Tags	CreatedByAMLStudio: true	Min replicas	1
Created on	3/16/2020 3:16:42 PM	Max replicas	10
Last updated on	3/16/2020 3:16:46 PM	Target utilization	70%
Compute target	aksvisual	Refresh period	1 s
REST endpoint	<a href="http://52.233.226.122:80/api/v1/service/visual-pipeline-real-time-infere/">http://52.233.226.122:80/api/v1/service/visual-pipeline-real-time-infere/</a>	App Insights enabled	true
score		Event Hubs enabled	false
Key-based authentication enabled	true	Storage enabled	false
Token-based authentication enabled	false	Region	westeurope
Swagger URI	<a href="http://52.233.226.122/api/v1/service/visual-pipeline-real-time-infere/swagger.json">http://52.233.226.122/api/v1/service/visual-pipeline-real-time-infere/swagger.json</a>	Last edited by	N/A
Related inference draft	<a href="#">Related inference draft</a>	Created by	N/A

# Visualisation du modèle déployé

The screenshot shows the Microsoft Azure Machine Learning studio interface. The top navigation bar includes 'Preview' and 'Microsoft Azure Machine Learning'. The left sidebar has a 'Endpoints' section highlighted, with options like 'Manage', 'Compute', 'Datastores', and 'Data Labeling'. The main content area displays the details for an endpoint named 'visual-pipeline-real-time-infere'. The 'Consumption' tab is selected. A red box highlights the 'Basic consumption info' and 'Consumption option' sections. Under 'Basic consumption info', it shows a REST endpoint at <http://52.233.226.122:80/api/v1/service/visual-pipeline-real-time-infere/score>. It provides two authentication options: 'Using key' (selected) and 'Using token'. It also lists two primary keys: 'gJLx3sS8fJbmqghwdU5vT3b0OFBlp7' and 'lbwAbYXG5u1WyzqNzN9zcn5PBpxTh7WRc', each with a 'Regenerate' link. Under 'Consumption option', there are three tabs: C#, Python (selected), and R. Below the tabs is a code snippet:

```
1 import urllib.request
2 import json
3 import os
4 import ssl
5
6 def allowSelfSignedHttps(allowed):
7     # bypass the server certificate verification on client side
8     if allowed and not os.environ.get('PYTHONHTTPSVERIFY', '') and getattr(ssl, '_create_unverified_context', None):
9         ssl._create_default_https_context = ssl._create_unverified_context
10
11 allowSelfSignedHttps(True) # this line is needed if you use self-signed certificate in your scoring service.
12
13 data = {
14     "Inputs": {
```

# Workshop

# Azure DevOps



# Azure DevOps

[Home](#) / [Services](#) / [Azure DevOps](#)

## Azure DevOps

Plan smarter, collaborate better, and ship faster with a set of modern dev services.

[Start free >](#)

[Start free with GitHub >](#)

Already have an account?

[Sign in to Azure DevOps >](#)



Azure DevOps ▾

[Azure DevOps Services](#)

[Customer stories](#)

[Documentation >](#)

[Support >](#)

[Pricing >](#)

[Blog >](#)



### Azure Boards

Deliver value to your users faster using proven agile tools to plan, track, and discuss work across your teams.

[Learn more >](#)



### Azure Pipelines

Build, test, and deploy with CI/CD that works with any language, platform, and cloud. Connect to GitHub or any other Git provider and deploy continuously.



### Azure Repos

Get unlimited, cloud-hosted private Git repos and collaborate to build better code with pull requests and advanced file management.

<https://azure.microsoft.com/en-us/services/devops/>

# Azure DevOps



## Azure Boards

Offrez de la valeur à vos utilisateurs plus rapidement en utilisant des outils agiles éprouvés pour planifier, suivre et commenter le travail entre équipes.

[En savoir plus >](#)



## Azure Pipelines

Générez, testez et déployez avec une approche CI/CD qui fonctionne avec n'importe quel langage, plateforme et cloud. Connectez-vous à GitHub ou à tout autre fournisseur Git et déployez en continu.

[En savoir plus >](#)



## Azure Repos

Bénéficiez de dépôts Git privés et illimités hébergés dans le cloud et collaborez pour créer un code de meilleure qualité avec les demandes de tirage et la gestion avancée des fichiers.

[En savoir plus >](#)



## Azure Test Plans

Testez et livrez en toute confiance à l'aide d'outils de tests manuels et exploratoires.

[En savoir plus >](#)



## Azure Artifacts

Créez, hébergez et partagez des packages avec votre équipe et ajoutez des artefacts à vos pipelines CI/CD en un seul clic.

[En savoir plus >](#)

## Place de marché des extensions

Accédez aux extensions proposées par Slack, SonarCloud et 1 000 autres applications et services, créés par la communauté.

[En savoir plus >](#)

<https://azure.microsoft.com/fr-fr/services/devops/#DevOps>

# Exemple de Pipeline

The screenshot shows the Azure DevOps Pipeline interface. On the left, the sidebar is visible with the following items:

- Overview
- Boards
- Repos
- Pipelines** (selected)
- Builds
- Releases
- Library
- Task groups
- Deployment groups
- WhiteSource Bolt
- Test Plans
- Artifacts
- Compliance

The main area displays the pipeline details:

**Pipeline** tab is selected.

**Release** section shows a "Continuous deployment" entry for Abel Wang on 4/11/2019, 10:15 AM. It includes an "Artifacts" section with "MercuryWebBuild 20190411.4" and "master".

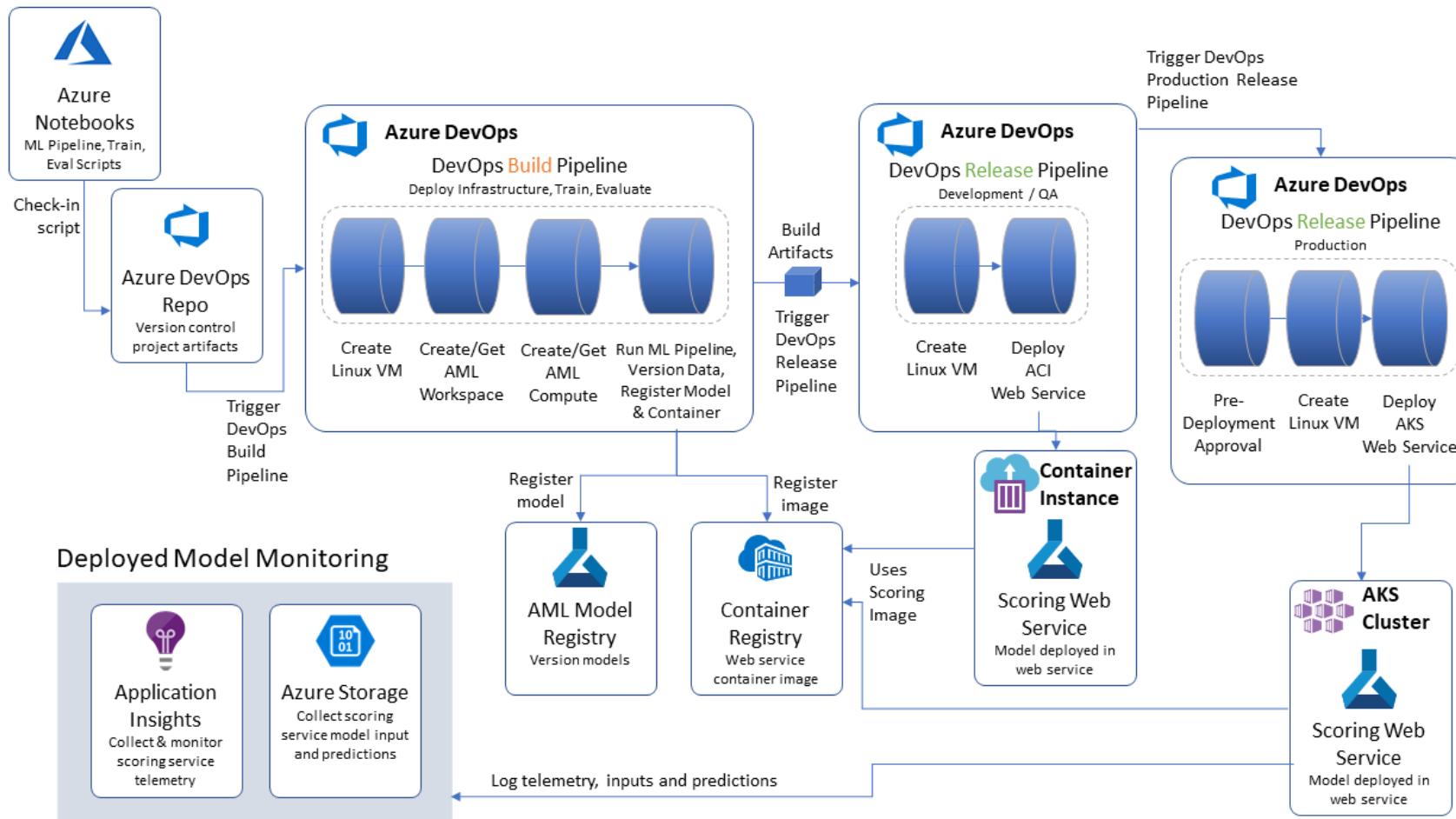
**Stages** section shows two stages: "Dev" and "QA". The "Dev" stage is green and labeled "Succeeded" with a timestamp of "on 4/11/2019, 10:18 AM". An arrow points from "Dev" to "QA". The "QA" stage is blue and labeled "Pending approval".

**QA** section on the right shows pre-deployment conditions and approvers:

- Pre-deployment conditions: Pending approval
- Approvers:
  - [MercuryHealth]\MercuryHealth Team (Pending)
    - Comment: ready to deploy
  - MT [MercuryHealth]\MercuryHealth Team (Pending)
    - Comment: ready to deploy
- Approval pending: Waiting for all approvers to approve in sequence.

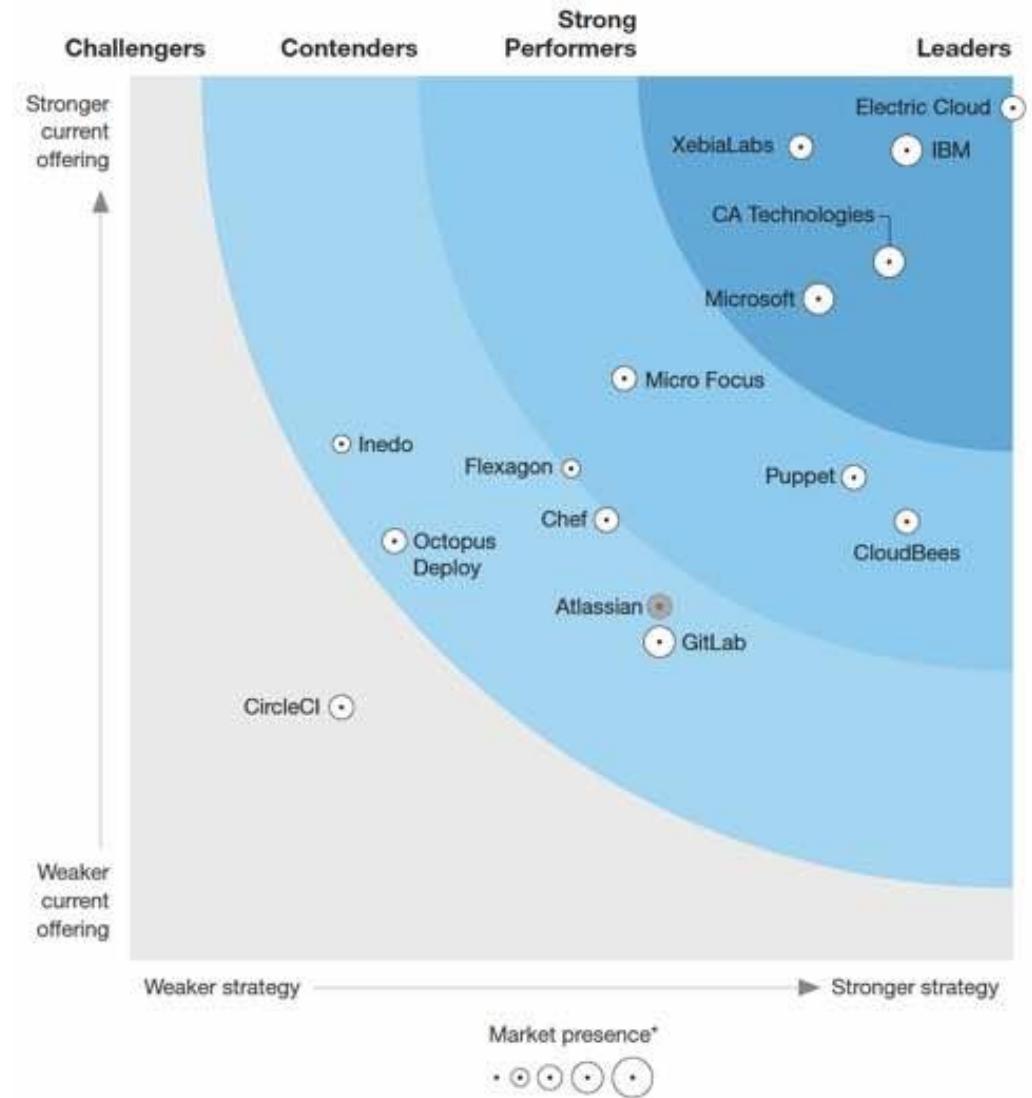
Buttons at the bottom: Approve (blue) and Reject.

# MLOps architecture



# Azure DevOps

Forrester has Microsoft among leaders (behind Electric Cloud, IBM, Xebia, and CA) in its [2018 Continuous Delivery And Release Automation](#) market assessment:



# MLOps

Azure DevOps    seretkov / Projet MLOps / Overview / Summary    Search    ⚙️    🛡️    ⓘ    🔍    🌐    📸

**Projet MLOps**

**About this project**

MLOps Workshop

Languages

Jupyter Notebook   Python

Projet MLOps / README.md

WorkshopMLOps

Workshop MLOps Azure

Azure ML Service: <https://azure.microsoft.com/en-us/services/machine-learning-service/>

MLOps: <https://docs.microsoft.com/fr-fr/azure/machine-learning/concept-model-management-and-deployment>

Serge Retkowsky | [serge.retkowsky@microsoft.com](mailto:serge.retkowsky@microsoft.com) | <https://www.linkedin.com/in/serger/>

**Project stats**

Last 7 days

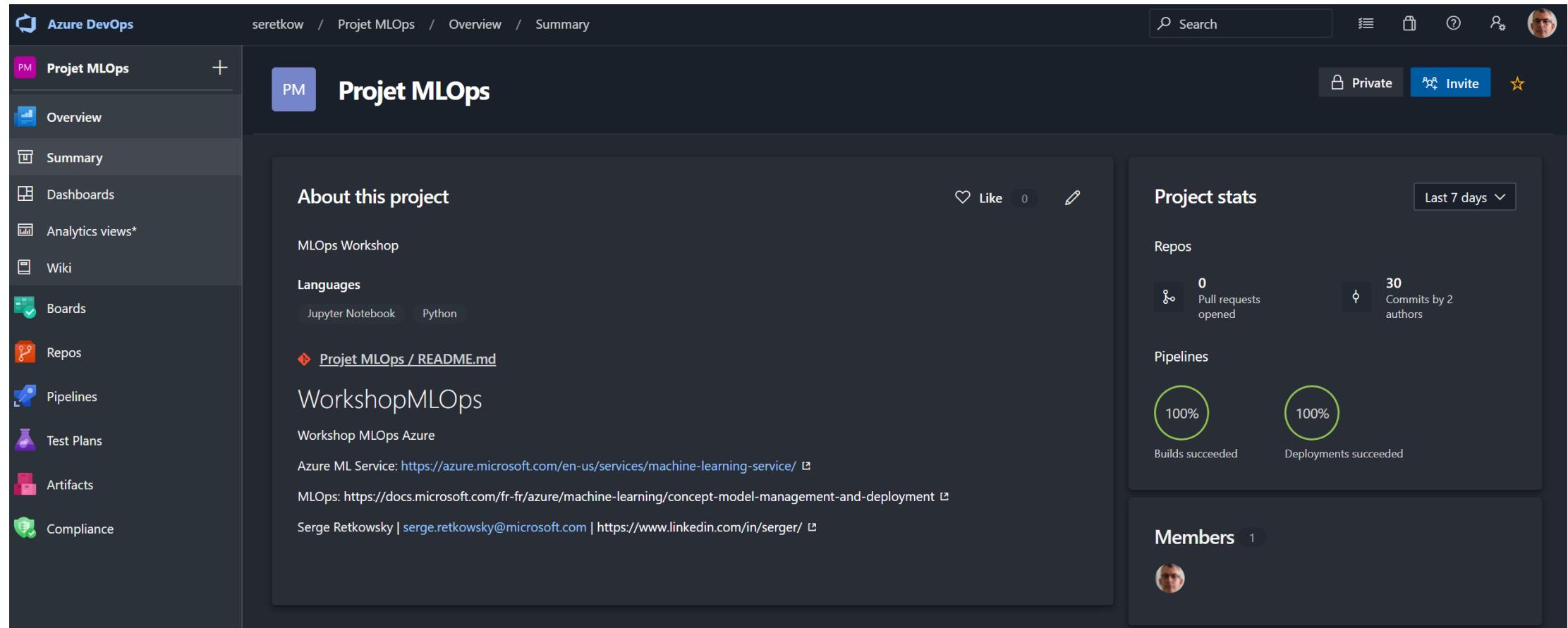
Repos

0 Pull requests opened   30 Commits by 2 authors

Pipelines

100% Builds succeeded   100% Deployments succeeded

Members 1



# Azure DevOps

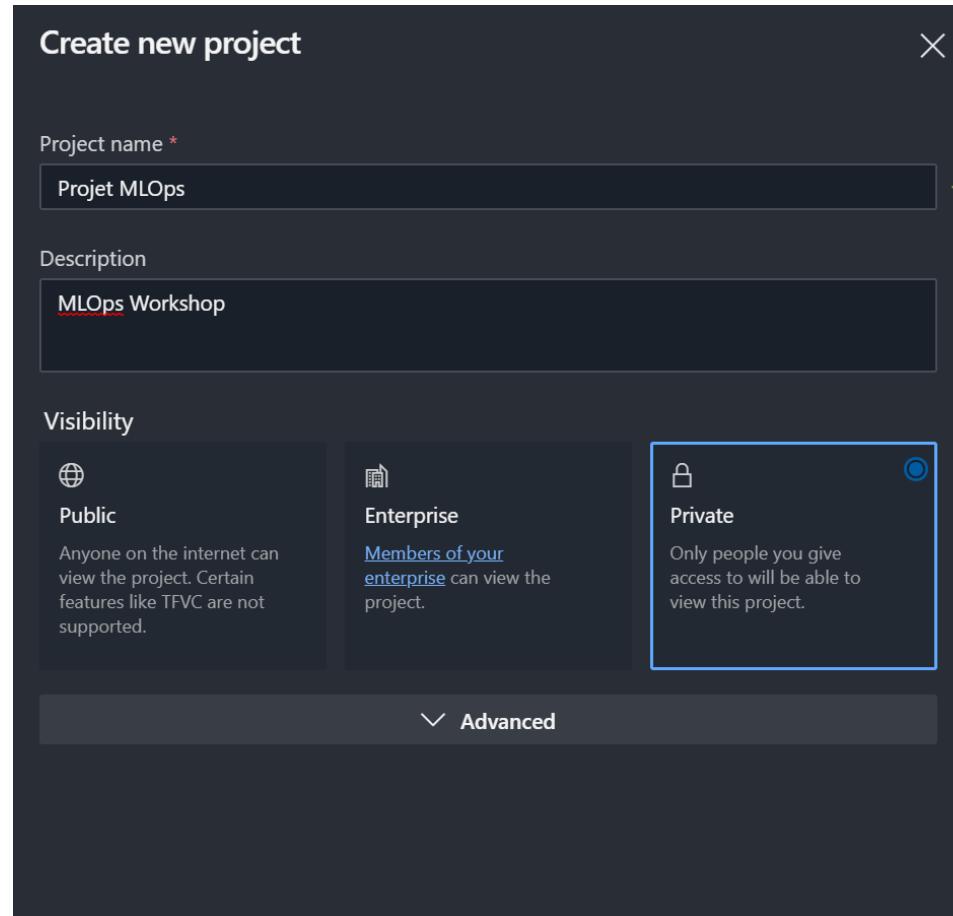
The screenshot shows the Azure DevOps interface for the organization 'seretkow'. The left sidebar lists organizations: 'seretkow' (selected), 'aidemos', 'emeaaits', 'msdata', '6 more organizations', and 'New organization'. A 'What's new' section mentions 'Sprint 165 release notes' and 'Organization settings'. The main area displays project cards:

- MLOps**: Démonstration MLOps avec AzureML service. <https://azure.microsoft.com/en-us/services/machine-learning-service/>
- pythonMLOps**: Démonstration MLOps avec Azure ML et Azure DevOps
- mlopsdemo**
- MLOps**: Démonstration MLOps avec AzureML service. <https://azure.microsoft.com/en-us/services/machine-learning-service/>
- MLOps for Azure Databricks**: MLOps for Azure Databricks. <https://github.com/SaschaDittmann/MLOps-Databricks>
- MLOps\_AzureDatabricks\_QuickDemo**: MLOps avec Azure Databricks. Utilisation MLFlow. <https://github.com/SaschaDittmann/MLOps-Databricks>
- mlopsdemo**
- MLOpsDemoSerge**

At the top right, there is a search bar, a 'New project' button, and a user profile icon.

<https://dev.azure.com>

# Création d'un projet Azure DevOps



# Projet

Azure DevOps

seretkov / Projet MLOps / Overview / Summary

Search

Project stats

Welcome to the project!

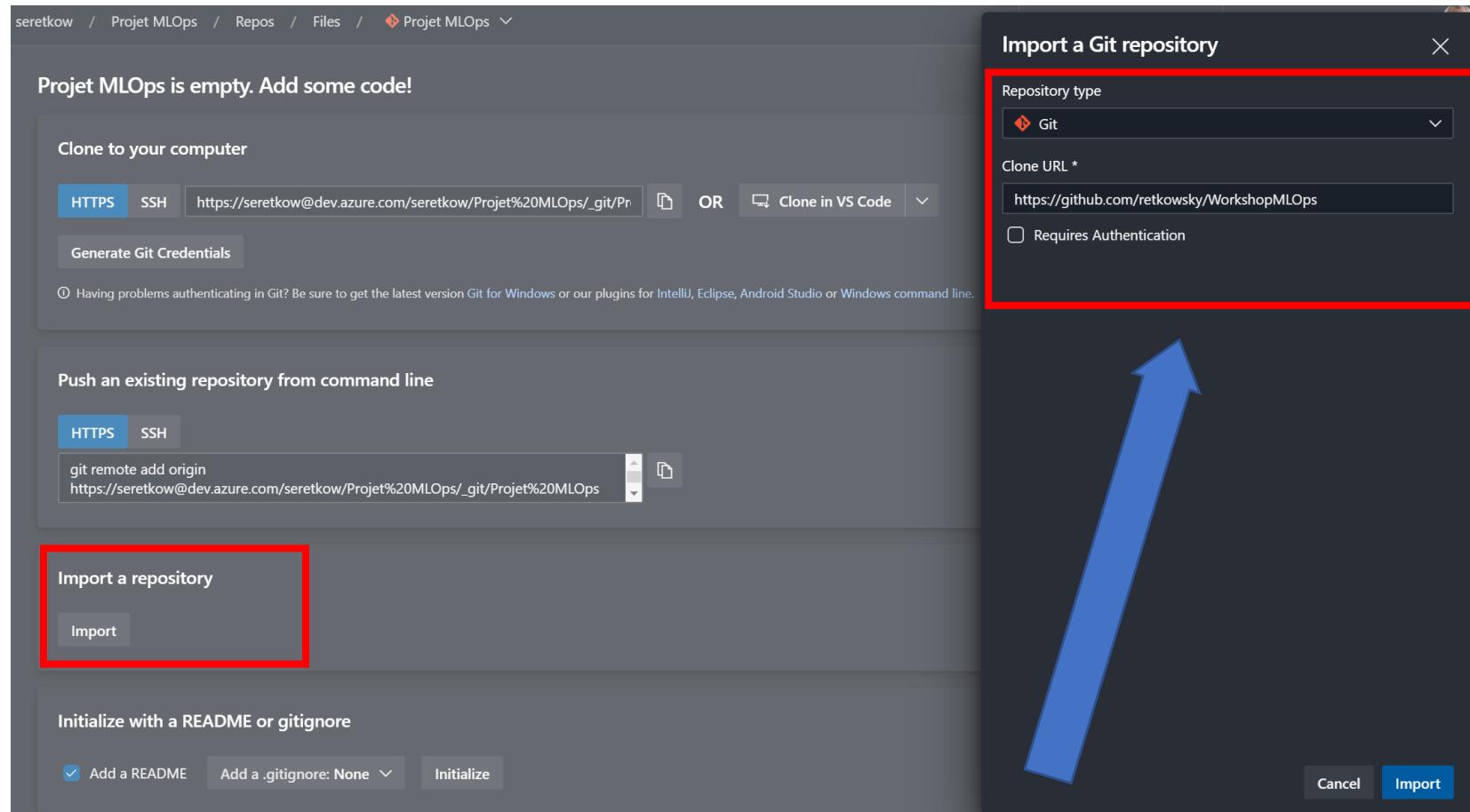
No stats are available at this moment

Setup a service to see project activity.

Members 1

The screenshot shows the Azure DevOps interface for the 'Projet MLOps' project. The left sidebar contains navigation links for Overview, Summary, Dashboards, Analytics views\*, Wiki, Boards, Repos, Pipelines, Test Plans, Artifacts, and Compliance. The main area features a central illustration of a person working at a desk with a dog. Below the illustration, the text 'Welcome to the project!' is displayed, followed by 'What service would you like to start with?' and a row of buttons for Boards, Repos, Pipelines, Test Plans, and Artifacts. To the right, there is a 'Project stats' section with a large circular icon containing a play button, indicating no data is available. Below this, a message says 'No stats are available at this moment' and 'Setup a service to see project activity.' At the bottom, a 'Members' section shows one user profile. The top right corner includes a 'Private' button, an 'Invite' button, and a user profile picture.

# Importation repo Git



<https://github.com/retkowsky/WorkshopMLOps>

# Résultats de l'importation

The screenshot shows the Azure DevOps interface for a repository named "Projet MLOps". The left sidebar is the navigation menu, and the main area displays the repository structure and file history.

**Repository Structure:**

- scripts-ml
- scripts-operationalize
- whitepaper
  - 01 Analyse ML Python.ipynb
  - 02 Modélisation avec Az...
  - 03 Test déploiement AKS.ipynb
  - 04 Télémétrie AKS.ipynb
  - build-master-pipeline.yml
  - framingham.csv
- M README.md

**File History (master branch):**

Name	Last change	Commits
scripts-ml	Yesterday	07b41045 Delete f Serge Retkowsky
scripts-operationalize	Yesterday	6e242aff Delete g Serge Retkowsky
whitepaper	4h ago	f98dbac4 Delete f Serge Retkowsky
01 Analyse ML Python.ipynb	59m ago	7ff69c4d Add files via upload Serge Retkowsky
02 Modélisation avec Azure ML.ipynb	59m ago	7ff69c4d Add files via upload Serge Retkowsky
03 Test déploiement AKS.ipynb	59m ago	7ff69c4d Add files via upload Serge Retkowsky
04 Télémétrie AKS.ipynb	59m ago	7ff69c4d Add files via upload Serge Retkowsky
build-master-pipeline.yml	Yesterday	0b725825 Add files via upload Serge Retkowsky
framingham.csv	2h ago	86aff7a5 Add files via upload Serge Retkowsky
M README.md	2h ago	a454bc48 Update README.md Serge Retkowsky

**Footer:**

WorkshopMLOps  
Workshop MLOps Azure  
Azure ML Service: <https://azure.microsoft.com/en-us/services/machine-learning-service/>

# Création d'un service connection

The screenshot shows the 'Service connections\*' page in the Azure DevOps interface. The left sidebar lists various project settings like General, Boards, and Repos. The main area features a user profile icon and the text 'Create your first service connection'. A call-to-action button labeled 'Create service connection' is highlighted with a red box.

seretkov / Projet MLOps / Settings / Service connections\*

Project Settings

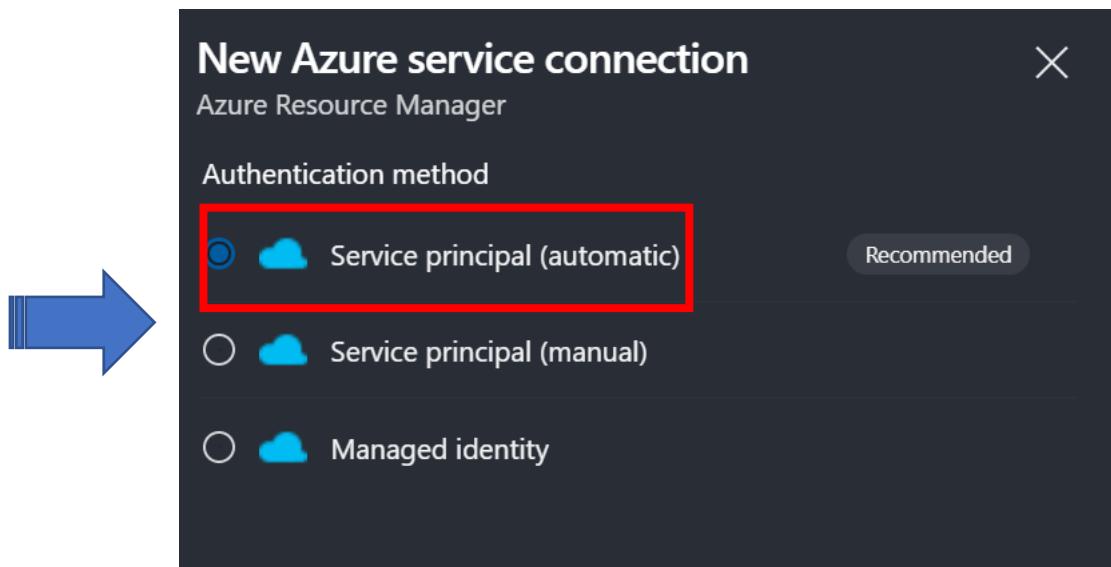
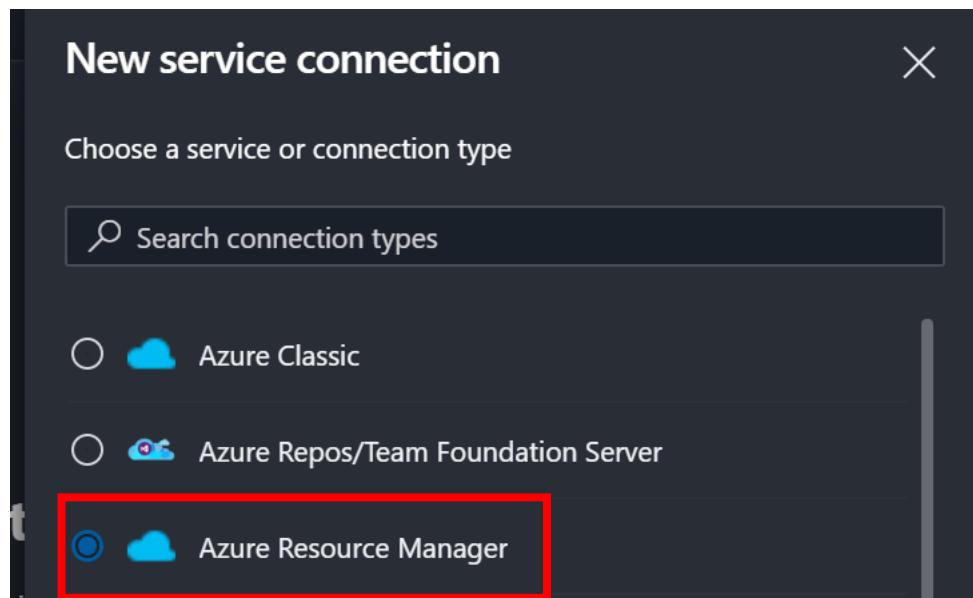
- Projet MLOps
- General
  - Overview
  - Teams
  - Permissions
  - Notifications
  - Service hooks
  - Dashboards
- Boards
  - Project configuration
  - Team configuration
  - GitHub connections
- Repos
  - Repositories
  - Cross-repo policies
- Pipelines

Create your first service connection

Service connections help you manage, protect, and reuse authentications to external services.

Create service connection

# Création d'un service connection



# Création d'un service connection

The screenshot shows the 'Create your first service connection' wizard on the left and the 'New Azure service connection' configuration dialog on the right. The configuration dialog is highlighted with a red border.

**New Azure service connection**  
Azure Resource Manager using service principal (automatic)

**Scope level**  
 Subscription  
 Management Group  
 Machine Learning Workspace

**Subscription**  
Microsoft Azure Internal Consumption (70b8f39e-8863-49f7-b...)

**Resource group**  
MLOpsWorkshopRG

**Machine Learning Workspace**  
MLOpsWorkshop

**Details**

**Service connection name**  
ncrMlopsServiceConnection

**Description (optional)**  
Connexion

**Security**  
 Grant access permission to all pipelines

Learn more      Back      Save

# Service connection créé

The screenshot shows the 'Service connections' page in the Azure DevOps interface. The left sidebar displays 'Project Settings' for the 'Projet MLOps' project, with sections for General, Boards, and GitHub connections. The main area is titled 'Service connections' and contains a single item: 'ncrMlopsServiceConnection'. This item is highlighted with a red rectangular box. The top right of the page includes a search bar, filter options, and a 'New service connection' button.

seretkow / Projet MLOps / Settings / Service connections\*

Search

New service connection

Project Settings

General

- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards

Boards

- Project configuration
- Team configuration
- Github connections

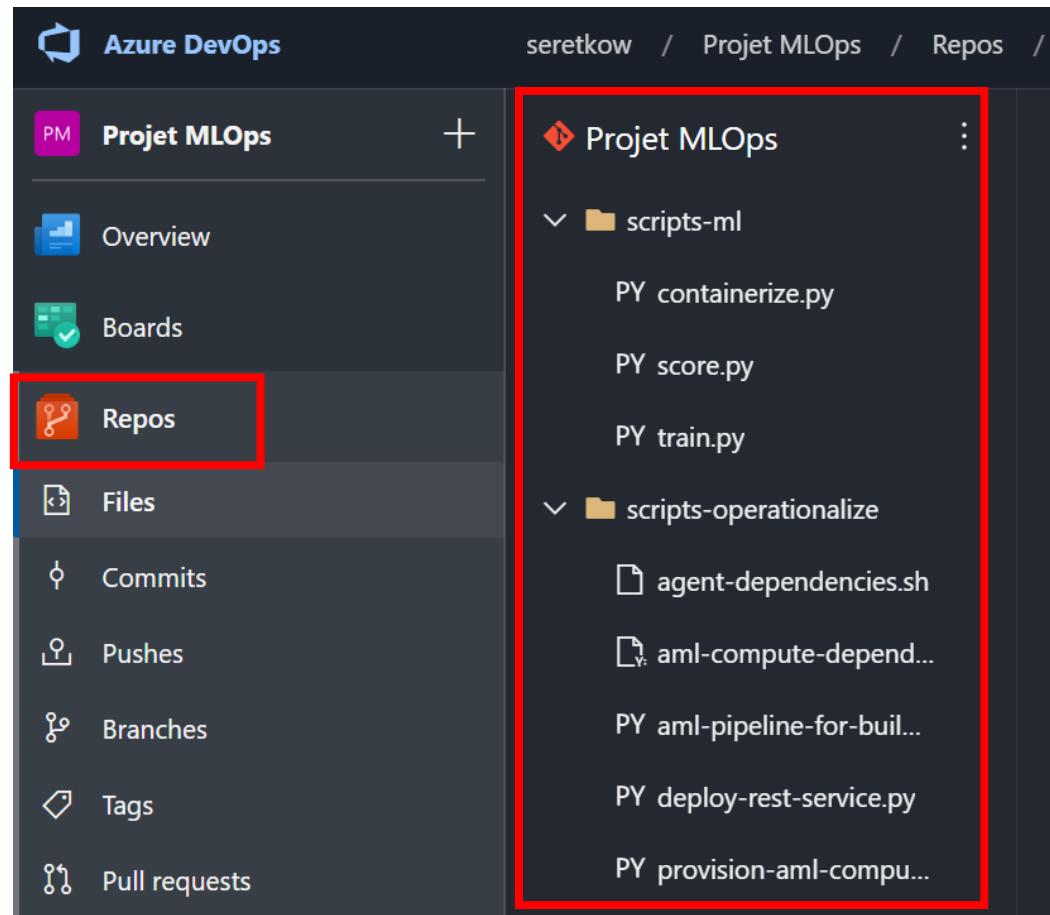
Service connections

Filter by keywords

Created by

ncrMlopsServiceConnection

# Visualisation des fichiers et scripts



# Détails des fichiers et scripts

## **build-master-pipeline.yml (Build pipeline YAML)**

- This is a script authored by the DevOps engineer typically
- It describes trigger for the pipeline, variables and pipeline steps in the order to be executed - it calls additional scripts to be executed.

## **2. train.py (Model training script)**

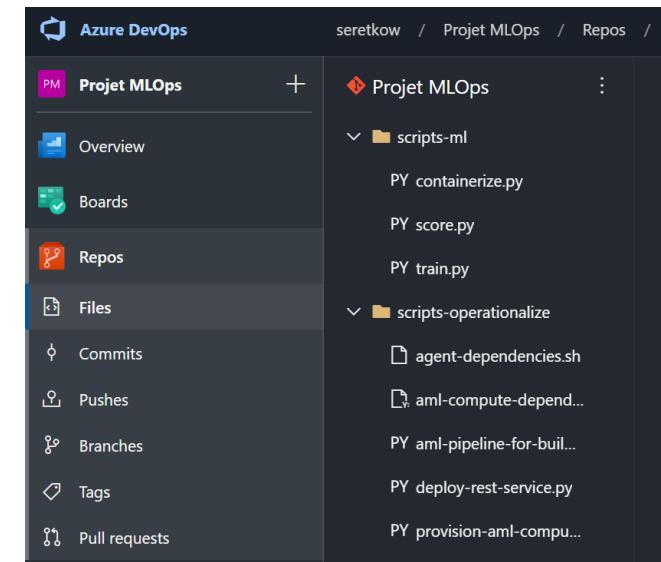
- This script is authored by the data scientist
- It trains the model and checks it into the AML model registry.
- It is called in the build pipeline

## **3. containerize.py (Container image creation script)**

- This script is authored by the data scientist
- It creates a docker container with dependencies and model packed in and checks it into the AML container registry
- It is called in the build pipeline

## **4. score.py (REST API based scoring script)**

- This script is authored by the data scientist
- It processes a scoring request by calling the REST service and returns a prediction
- It is deployed in the release pipeline



# Détails des fichiers et scripts

## 5. agent-dependencies.sh

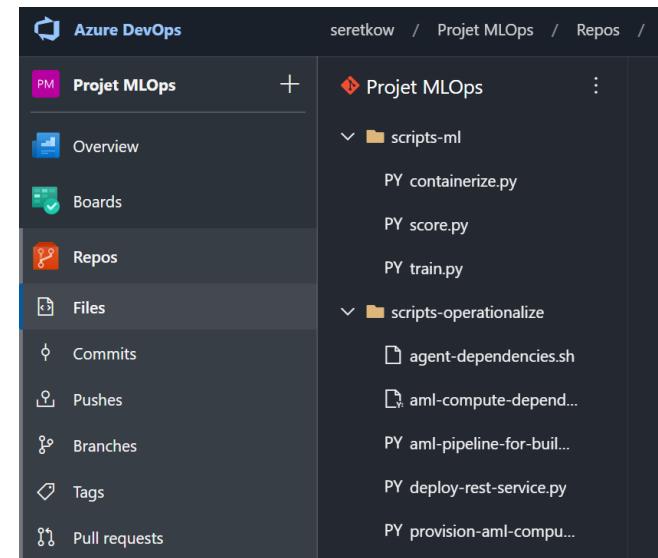
- This script is authored by the devops engineer in consultation with the data scientist
- It installs software dependencies for MLOps on the agents
- It is leveraged in the build and release pipelines

## 6. aml-compute-dependencies.yml

- This script is authored by the devops engineer in consultation with the data scientist
- It installs software dependencies for MLOps on training compute instances
- It is leveraged in the build pipeline

## 7. aml-pipeline-for-build.py

- This script is authored by the devops engineer in consultation with the data scientist
- It creates an AML pipeline with stages in it and executes it
- It is leveraged in the build pipeline



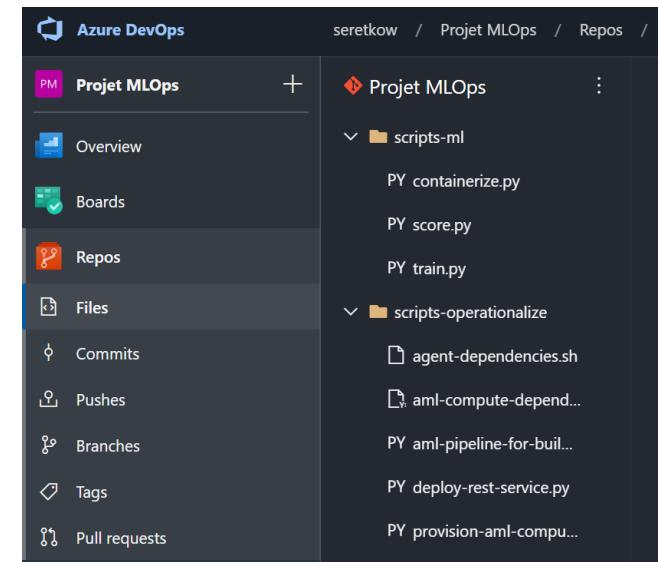
# Détails des fichiers et scripts

## 8. provision-aml-compute.py

- This script is authored by the devops engineer in consultation with the data scientist
- It provisions transient AML compute for model training
- It is leveraged in the build pipeline

## 9. deploy-rest-service.py

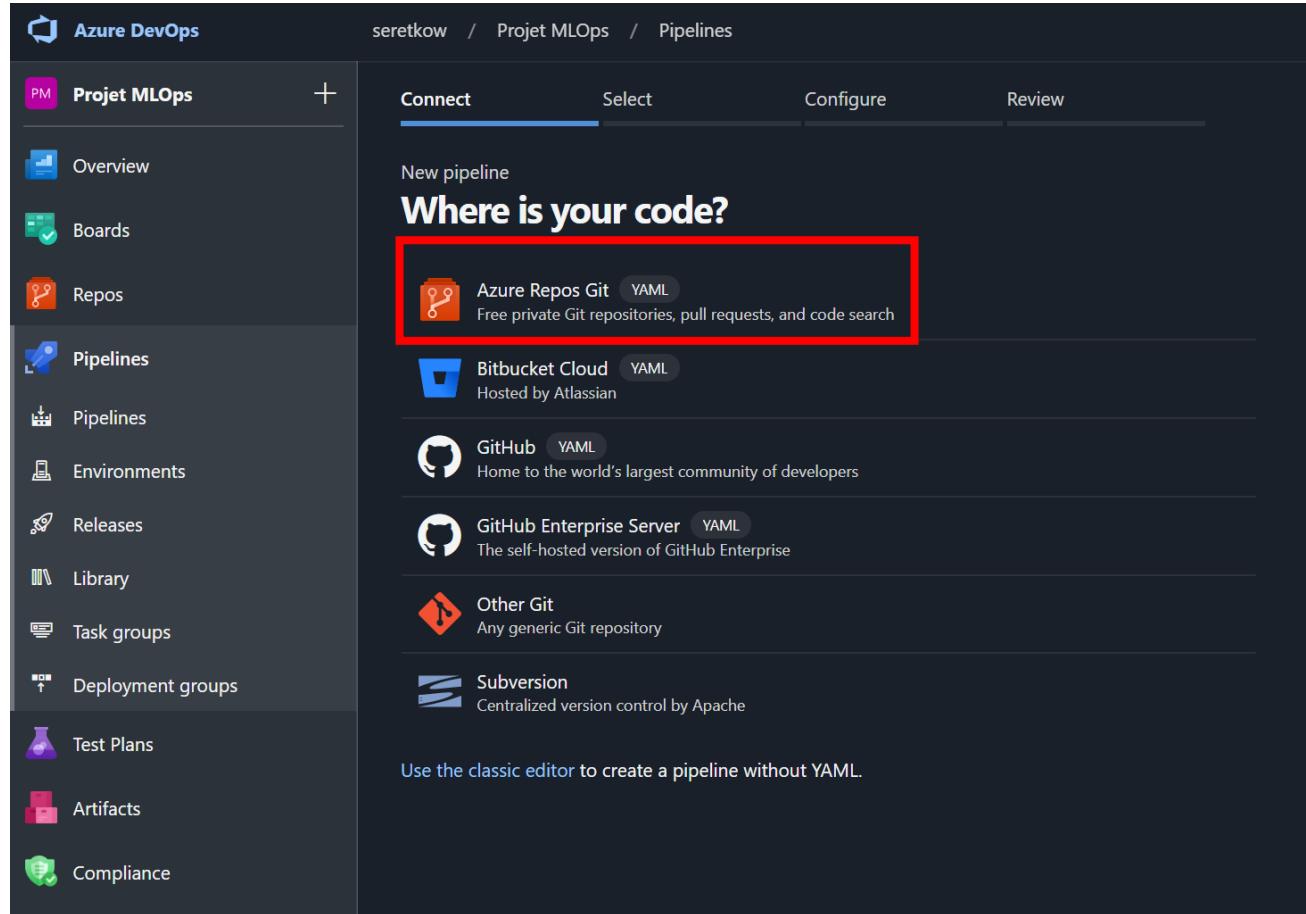
- This script is authored by the devops engineer in consultation with the data scientist
- It provisions an AKS cluster and a REST service on the same leveraging the container image and scoring file
- It is leveraged in the release pipeline



# Création d'un Pipeline

The screenshot shows the Azure DevOps interface for creating a pipeline. The left sidebar lists project navigation options: Overview, Boards, Repos, Pipelines (highlighted with a red box), Environments, Releases, Library, Task groups, Deployment groups, Test Plans, Artifacts, and Compliance. The main area features a central illustration of a person working at a laptop, a cat, and a robot, with clouds in the background. Below the illustration, the text "Create your first Pipeline" is displayed in bold. A subtitle explains: "Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes." A prominent blue button labeled "Create Pipeline" is centered at the bottom of the main area, also highlighted with a red box.

# Sélectionner Azure Repos Git



The screenshot shows the Azure DevOps interface for creating a new pipeline. The left sidebar lists various project management and development tools. The main area is titled "New pipeline" and features a prominent heading "Where is your code?". Below this, a list of integration options is displayed, each with a small icon, the provider name, "YAML" badge, and a brief description. The first item, "Azure Repos Git", is highlighted with a red rectangular box. The other items listed are Bitbucket Cloud, GitHub, GitHub Enterprise Server, Other Git, and Subversion.

Provider	YAML	Description
Azure Repos Git	YAML	Free private Git repositories, pull requests, and code search
Bitbucket Cloud	YAML	Hosted by Atlassian
GitHub	YAML	Home to the world's largest community of developers
GitHub Enterprise Server	YAML	The self-hosted version of GitHub Enterprise
Other Git		Any generic Git repository
Subversion		Centralized version control by Apache

# Sélectionner le repo

The screenshot shows the Azure DevOps interface for creating a new pipeline. The top navigation bar includes the Azure DevOps logo, the organization name 'seretkov', the project 'Projet MLOps', and the current page 'Pipelines'. Below this, a progress bar indicates the steps: 'Connect' (with a checkmark), 'Select' (which is highlighted in blue), 'Configure', and 'Review'. The main area is titled 'New pipeline' and features a large heading 'Select a repository'. A search bar at the top of this section has the text 'Projet MLOps' and a clear button. Below the search bar, a list of repositories is shown, with the first item, 'Projet MLOps', highlighted by a red rectangular box. The left sidebar contains links for 'Overview', 'Boards', 'Repos' (which is currently selected and highlighted in pink), 'Pipelines', 'Environments', and 'Pipelines' again.

# Configuration du Pipeline

The screenshot shows the Azure DevOps interface for configuring a new pipeline. The top navigation bar includes the Azure DevOps logo, the project name "Projet MLOps", and the path "seretkow / Projet MLOps / Pipelines". Below this, a progress bar indicates the steps: "✓ Connect", "✓ Select", "Configure" (which is underlined in blue), and "Review". The main content area is titled "New pipeline" and "Configure your pipeline". It lists several pipeline types:

- Python package**: Create and test a Python package on multiple Python versions.
- Python to Linux Web App on Azure**: Build your Python project and deploy it to Azure as a Linux Web App.
- Starter pipeline**: Start with a minimal pipeline that you can customize to build and deploy your code.
- Existing Azure Pipelines YAML file**: Select an Azure Pipelines YAML file in any branch of the repository. This option is highlighted with a red rectangular box.

A "Show more" button is located at the bottom of the list.

# Chemin du fichier YAML

The screenshot shows the Azure Pipelines interface for creating a new pipeline. The top navigation bar indicates the user is in the 'seretkow / Projet MLOps / Pipelines' section, and the 'Configure' step is active.

**Configure your pipeline**

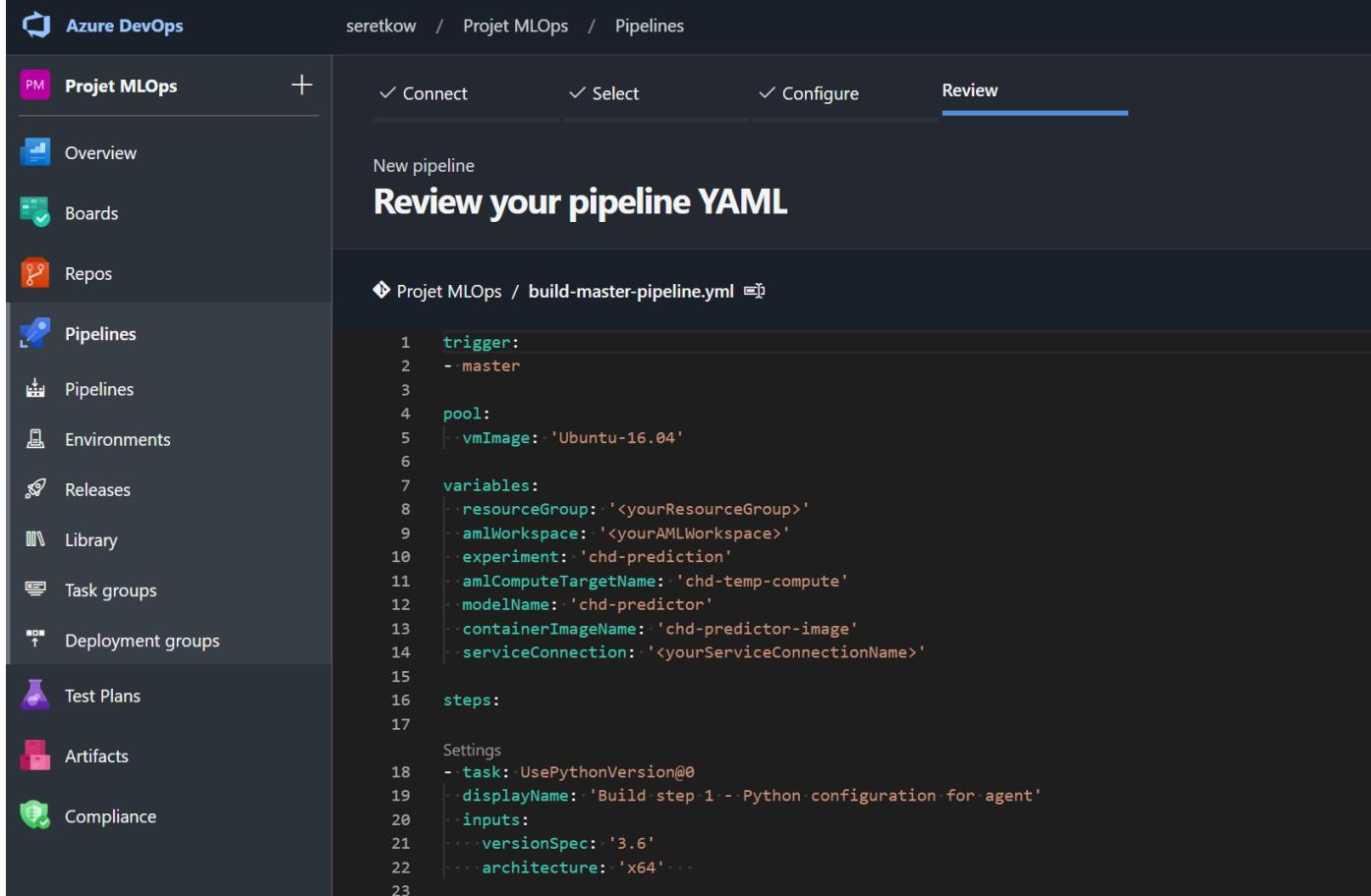
- Python package**: Create and test a Python package on multiple Python versions.
- Python to Linux Web App on Azure**: Build your Python project and deploy it to Azure as a Linux Web App.
- Starter pipeline**: Start with a minimal pipeline that you can customize to build and deploy your code.
- Existing Azure Pipelines YAML file**: Select an Azure Pipelines YAML file in any branch of the repository.

**Show more**

A modal dialog titled "Select an existing YAML file" is open, overlaid on the main configuration screen. It shows the "Branch" dropdown set to "master" and the "Path" dropdown set to "/build-master-pipeline.yml". The "Path" dropdown is highlighted with a red border.

Select an existing YAML file  
Select an Azure Pipelines YAML file in any branch of the repository.  
Branch master  
Path /build-master-pipeline.yml  
Select a file from the dropdown or type in the path to your file  
Projet MLOps

# Visualisation du fichier YAML



The screenshot shows the Azure DevOps interface for creating a new pipeline. The left sidebar lists various project management and development tools: Overview, Boards, Repos, Pipelines, Releases, Library, Task groups, Deployment groups, Test Plans, Artifacts, and Compliance. The main area is titled "Review your pipeline YAML" and displays the following YAML code:

```
trigger:
- master

pool:
  vmImage: 'Ubuntu-16.04'

variables:
  resourceGroup: '<yourResourceGroup>'
  amlWorkspace: '<yourAMLWorkspace>'
  experiment: 'chd-prediction'
  amlComputeTargetName: 'chd-temp-compute'
  modelName: 'chd-predictor'
  containerImageName: 'chd-predictor-image'
  serviceConnection: '<yourServiceConnectionName>'

steps:
  - task: UsePythonVersion@0
    displayName: 'Build step 1 -- Python configuration for agent'
    inputs:
      versionSpec: '3.6'
      architecture: 'x64'
```

# Modifier les variables du fichier YAML avec vos informations

New pipeline

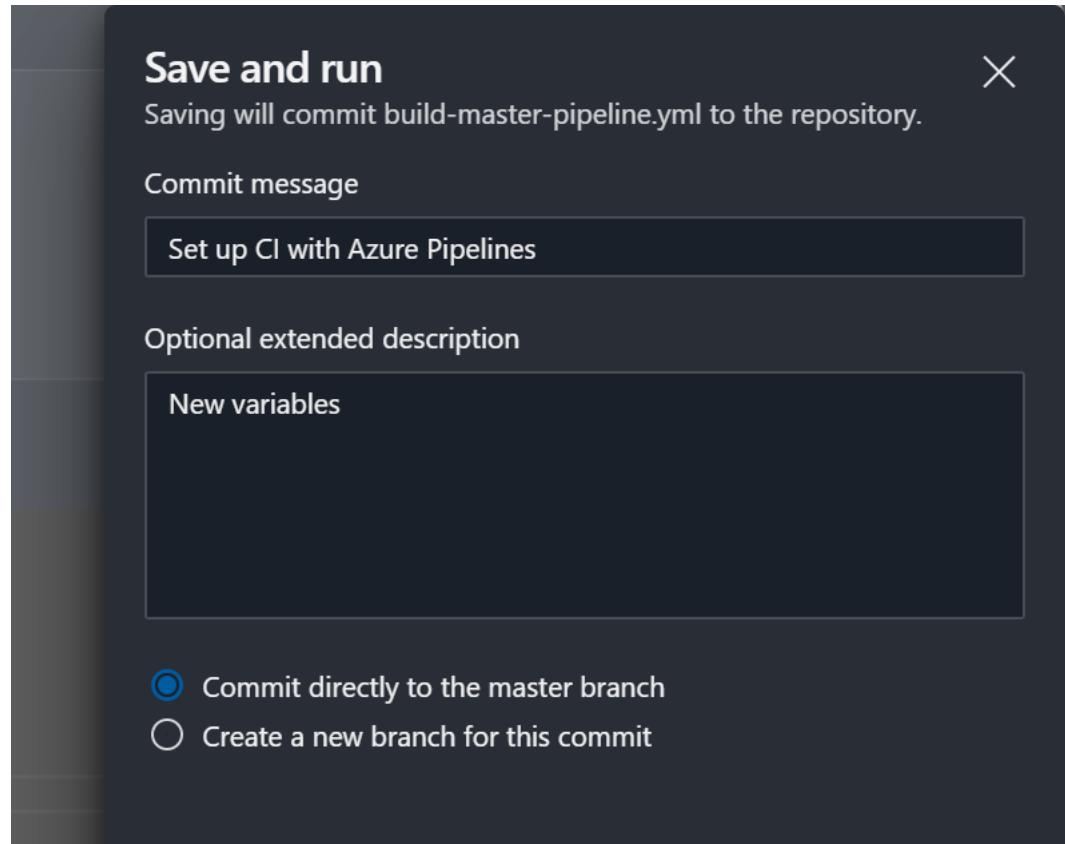
## Review your pipeline YAML

◆ Projet MLOps / build-master-pipeline.yml \* ☰

```
1 trigger:
2   - master
3
4 pool:
5   vmImage: 'Ubuntu-16.04'
6
7 variables:
8   resourceGroup: 'MLOpsworkshopRG'
9   amlWorkspace: 'MLOpsWorkshop'
10  experiment: 'chd-prediction'
11  amlComputeTargetName: 'chd-temp-compute'
12  modelName: 'chd-predictor'
13  containerImageName: 'chd-predictor-image'
14  serviceConnection: 'ncrMlopsServiceConnection'
15
16 steps:
17
18   Settings
19     - task: UsePythonVersion@0
20       displayName: 'Build step 1 - Python configuration for agent'
21       inputs:
22         versionSpec: '3.6'
23         architecture: 'x64' ...
```

Settings

# Sauvegarde du YAML



# Mise à jour automatique du Pipeline

The screenshot shows the Azure DevOps interface for a project named "Projet MLOps". The left sidebar has a "Pipelines" section selected. The main area displays a pipeline run titled "#20200312.1 Set up CI with Azure Pipelines" triggered by a commit from the "master" branch. The "Jobs" table lists one job named "Job" which is currently in the "Queued" status. A red box highlights the "Job" row in the table.

Azure DevOps

seretkov / Projet MLOps / Pipelines / Projet MLOps / 20200312.1

Search

Cancel

Projet MLOps

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Releases

Library

Task groups

Deployment groups

Test Plans

Artifacts

Compliance

#20200312.1 Set up CI with Azure Pipelines  
on Projet MLOps

Triggered by Serge Retkowsky

Projet MLOps master dc052d5  
Just now

Duration: - Tests: - Changes: 27 commits Work items: - Artifacts: -

Name	Status	Duration
Job	Queued	-

# Pipeline en cours d'exécution

**Jobs**

- Job (8s)
  - Initialize job (3s) ✓
  - Checkout Projet MLOps... (1s) ✓
  - Build step 1 - Python ... (<1s) ✓
  - Build step 2 - Install de... (3s) ⓘ
  - Build step 3 - Install the A... ⓘ
  - Build step 4 - Attach direc... ⓘ
  - Build step 5 - Provision A... ⓘ
  - Build step 6 - Define and I... ⓘ
  - Build step 7 - Copy build ... ⓘ
  - Build step 8 - Publish buil... ⓘ
  - Component Detection (au... ⓘ
  - Post-job: Checkout Projet... ⓘ

**Build step 2 - Install dependencies on agent**

```
1 Starting: Build step 2 - Install dependencies on agent
2 =====
3 Task      : Bash
4 Description: Run a Bash script on macOS, Linux, or Windows
5 Version   : 3.163.1
6 Author    : Microsoft Corporation
7 Help      : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/bash
8 =====
9 Generating script.
10 ##[warning]Executable bit is not set on target script, sourcing instead of executing. More info at https://github.com/Microsoft/azure-pipelines-tasks/blob/master/Tasks/Bash@3/README.md#script-execution
11 Formatted command: . '/home/vsts/work/1/s/scripts-operationalize/agent-dependencies.sh'
12 ===== Starting Command Output =====
13 /bin/bash --noprofile --norc /home/vsts/work/_temp/4b041792-3e86-4927-a36b-ccb6924ea0af.sh
14 Python 3.6.10
15 Collecting azure-cli==2.0.72
16   Downloading azure_cli-2.0.72-py2.py3-none-any.whl (1.3 MB)
17 Collecting azure-mgmt-datamigration==0.1.0
18   Downloading azure_mgmt_datamigration-0.1.0-py2.py3-none-any.whl (219 kB)
```

# Fin du traitement

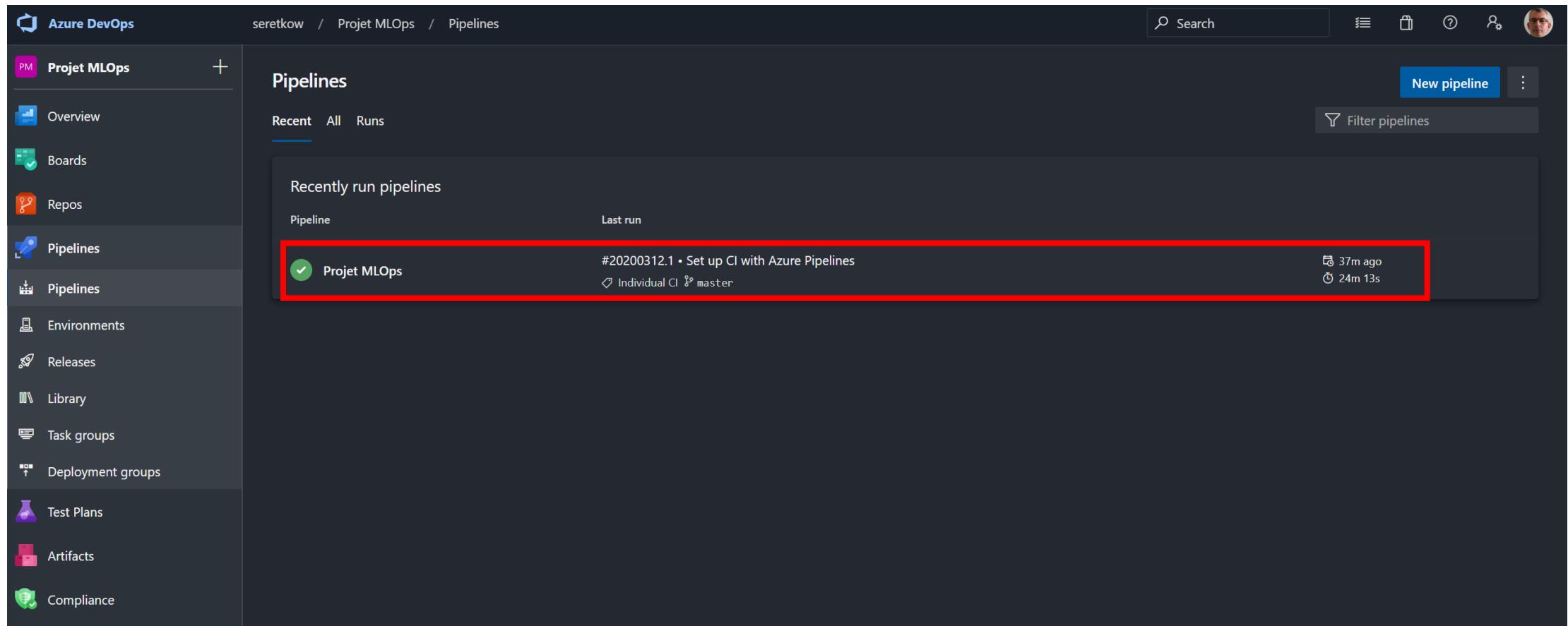
The screenshot shows the Azure DevOps interface for a pipeline named "Projet MLOps". The left sidebar lists various project management and development tools. The main area displays the "Jobs in run #20200312.1" for the "Projet MLOps" pipeline. A red box highlights the list of jobs, which includes:

- Job (24m 7s)
- Initialize job (3s)
- Checkout Projet MLOps... (1s)
- Build step 1 - Python ... (<1s)
- Build step 2 - Install... (1m 2s)
- Build step 3 - Install th... (23s)
- Build step 4 - Attach d... (12s)
- Build step 5 - Provis... (2m 2s)
- Build step 6 - Def... (19m 42s)
- Build step 7 - Copy b... (<1s)
- Build step 8 - Publish b... (4s)
- Component Detection... (34s)
- Post-job: Checkout Pr... (<1s)
- Finalize Job (<1s)
- Report build status (<1s)

The right panel shows the detailed log for the final build step, titled "Build step 6 - Define and launch build pipeline". The log output is as follows:

```
1 Starting: Build step 6 - Define and launch build pipeline
2 -----
3 Task      : Azure CLI
4 Description : Run Azure CLI commands against an Azure subscription in a Shell
5 Version   : 1.163.0
6 Author    : Microsoft Corporation
7 Help      : https://docs.microsoft.com/azure/devops/pipelines/tasks/deploy/az
8 -----
9 /opt/hostedtoolcache/Python/3.6.10/x64/bin/az --version
10 azure-cli           2.0.72 *
11 WARNING: You have 2 updates available. Consider updating your CLI installation.
12 -----
13 command-modules-nspkg      2.0.3
14 core                  2.0.72 *
15 nspkg                 3.0.4
16 telemetry              1.0.4
17 -----
18 Extensions:
19 azure-cli-ml            1.1.5
20 azure-devops             0.17.0
21 -----
22 Python location '/opt/hostedtoolcache/Python/3.6.10/x64/bin/python'
23 Extensions directory '/opt/az/azclixextensions'
24 -----
25 Python (Linux) 3.6.10 (default, Feb 14 2020, 11:07:53)
26 [GCC 5.4.0 20160609]
27 -----
28 Legal docs and information: aka.ms/AzureCliLegal
29 -----
30 -----
31 Setting AZURE_CONFIG_DIR env variable to: /home/vsts/work/_temp/.azclitask
32 Setting active cloud to: AzureCloud
33 /opt/hostedtoolcache/Python/3.6.10/x64/bin/az cloud set -n AzureCloud
```

# Visualisation des pipelines exécutés



The screenshot shows the Azure DevOps Pipelines interface for the project "Projet MLOps". The left sidebar navigation bar includes links for Overview, Boards, Repos, Pipelines (selected), Environments, Releases, Library, Task groups, Deployment groups, Test Plans, Artifacts, and Compliance. The main content area displays a table titled "Recently run pipelines" with columns for "Pipeline" and "Last run". A single row is shown, highlighted with a red box: "Projet MLOps" (Pipeline) - "#20200312.1 • Set up CI with Azure Pipelines" (Last run). The last run was triggered by "Individual CI" on branch "master" at 37m ago, with a duration of 24m 13s.

Pipeline	Last run
Projet MLOps	#20200312.1 • Set up CI with Azure Pipelines ⌚ Individual CI ⚖ master 🕒 37m ago 🕒 24m 13s

# Modification du fichier train.py

The screenshot shows the Azure DevOps interface for a project named "Projet MLOps". The left sidebar contains navigation links for Overview, Boards, Repos, Files, Commits, Pushes, Branches, Tags, Pull requests, Pipelines, Test Plans, Artifacts, and Compliance. The main area shows a file structure under "Repos / Projet MLOps / master / scripts-ml / train.py". The "train.py" file is open in the code editor, displaying Python code for data preprocessing and training a Random Forest classifier. A red box highlights the line of code where the classifier's maximum depth is set to 3.

```
54 df['glucose'].fillna(df['glucose'].mean(), inplace = True)
55 df['totChol'].fillna(df['totChol'].mean(), inplace = True)
56 df['education'].fillna(1, inplace = True)
57 df['BMI'].fillna(df['BMI'].mean(), inplace = True)
58 df['heartRate'].fillna(df['heartRate'].mean(), inplace = True)
59 print(..3. completed')
60 print('')
61 print('')
62
63 print("4. Train model")
64 print('.....')
65 # Features and label
66 features = df.iloc[:, :-1]
67 result = df.iloc[:, -1] # the last column is what we are about to forecast
68
69 # Train & Test split
70 X_train, X_test, y_train, y_test = train_test_split(features, result, test_size=0.2, random_state=42)
71
72 # RandomForest classifier
73+clf = RandomForestClassifier(n_estimators=100, max_depth=3, random_state=42)
74 clf.fit(X_train, y_train)
75
76 # Create a selector object that will use the random forest classifier to identify
77 # features that have an importance of more than 0.12
78 sfm = SelectFromModel(clf, threshold=0.12)
79
80 # Train the selector
81 sfm.fit(X_train, y_train)
82
83 # Features selected
84 featureNames = list(features.columns.values) # creating a list with features
85 print("Feature names:")
86 for featureNameListIndex in sfm.get_support(indices=True):
87 | print(featureNames[featureNameListIndex])
88
89 # Feature importance
90 importances = clf.feature_importances_
91 std = np.std([tree.feature_importances_ for tree in clf.estimators_],
```

# Sauvegarde du fichier train.py

The screenshot shows a GitHub commit interface for a file named 'train.py' in a repository 'Projet MLOps'. The commit message is highlighted with a red border and contains the text: "Updated train.py (Change max\_depth=3 vs 2)".

```
seretkow / Projet MLOps / Repos / Files / Projet MLOps

Projet MLOps : master / scripts-ml / train.py

train.py

Contents Highlight changes

54 df['glucose'].fillna(df['glucose'].mean(), inplace = True)
55 df['totChol'].fillna(df['totChol'].mean(), inplace = True)
56 df['education'].fillna(1, inplace = True)
57 df['BMI'].fillna(df['BMI'].mean(), inplace = True)
58 df['heartRate'].fillna(df['heartRate'].mean(), inplace = True)
59 print('..3. completed')
60 print('')
61 print('')
62
63 print("4. Train model")
64 print('.....')
65 # Features and label
66 features = df.iloc[:, :-1]
67 result = df.iloc[:, -1] # the last column is what we are about to forecast
68
69 # Train & Test split
70 x_train, x_test, y_train, y_test = train_test_split(features, result, test_size=0.2)
71
72 # RandomForest classifier
73 clf = RandomForestClassifier(n_estimators=100, max_depth=2, random_state=0)
74 clf.fit(x_train, y_train)
75
76 # Create a selector object that will use the random forest classifier to identify
77 # features that have an importance of more than 0.12
78 sfm = SelectFromModel(clf, threshold=0.12)
79
80 # Train the selector
81 sfm.fit(x_train, y_train)

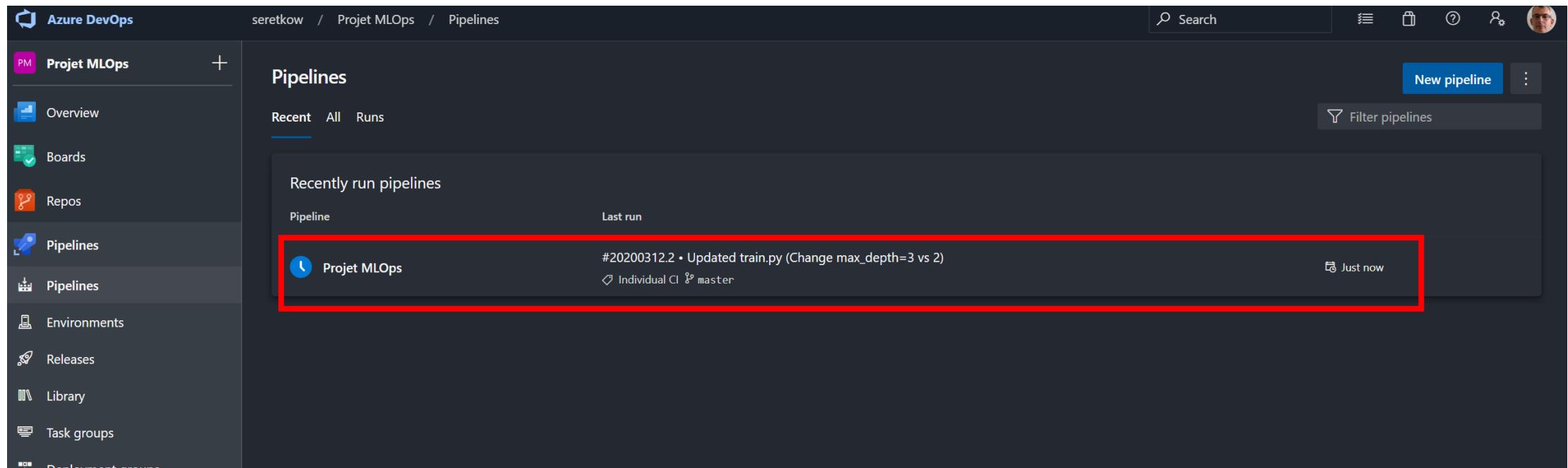
Commit

Comment
Updated train.py (Change max_depth=3 vs 2)

Branch name
master

Work items to link
Search work items by ID or title
```

# Le Pipeline s'exécute automatiquement



The screenshot shows the Azure DevOps Pipelines interface for the project "Projet MLOps". The left sidebar has "Pipelines" selected. The main area displays the "Recently run pipelines" section. A single pipeline entry is shown, highlighted with a red rectangle:

Pipeline	Last run
Projet MLOps	#20200312.2 • Updated train.py (Change max_depth=3 vs 2) ⌚ Individual CI ↗ master Just now

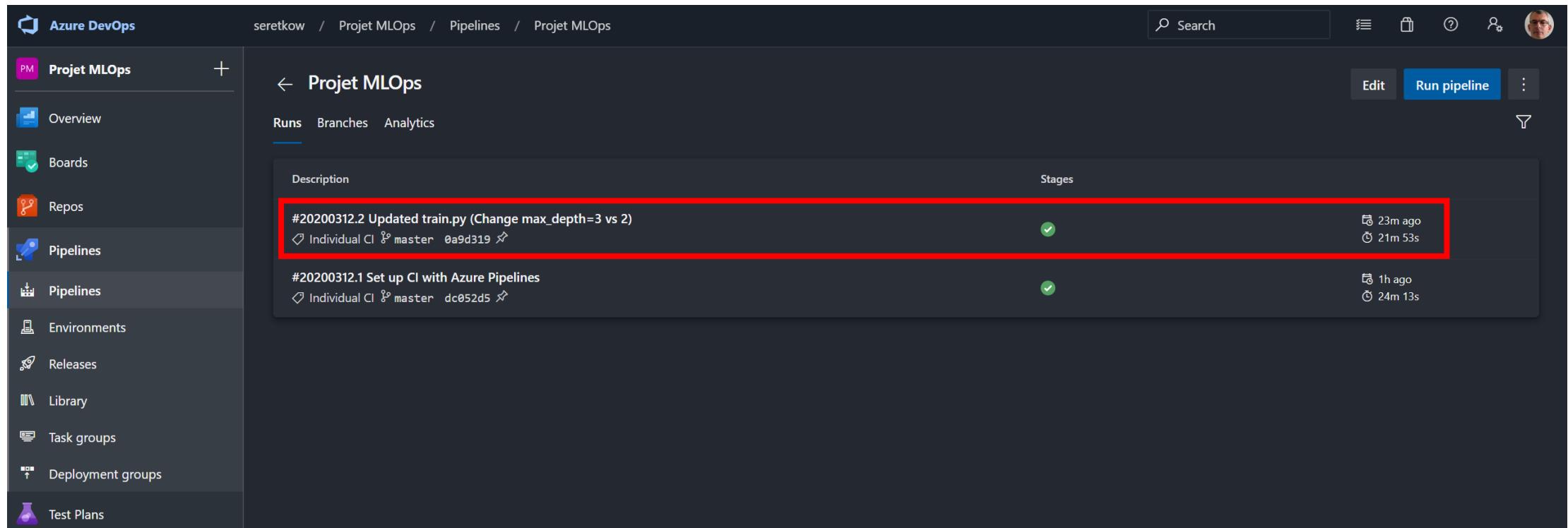
The pipeline name is "Projet MLOps". The last run was at "#20200312.2", with the message "Updated train.py (Change max\_depth=3 vs 2)". It was triggered by "Individual CI" and ran on the "master" branch, completed "Just now".

# Exécution en cours

The screenshot shows the Azure DevOps Pipelines interface for the 'Projet MLOps' project. The left sidebar lists various project management and pipeline-related tabs. The 'Pipelines' tab is selected, and the main area displays a list of recent pipeline runs.

Description	Stages	Created	Last Run
#20200312.2 Updated train.py (Change max_depth=3 vs 2) ↳ Individual CI ⚙ master 0a9d319	C	Just now	13s
#20200312.1 Set up CI with Azure Pipelines ↳ Individual CI ⚙ master dc052d5 ✘	✓	39m ago	24m 13s

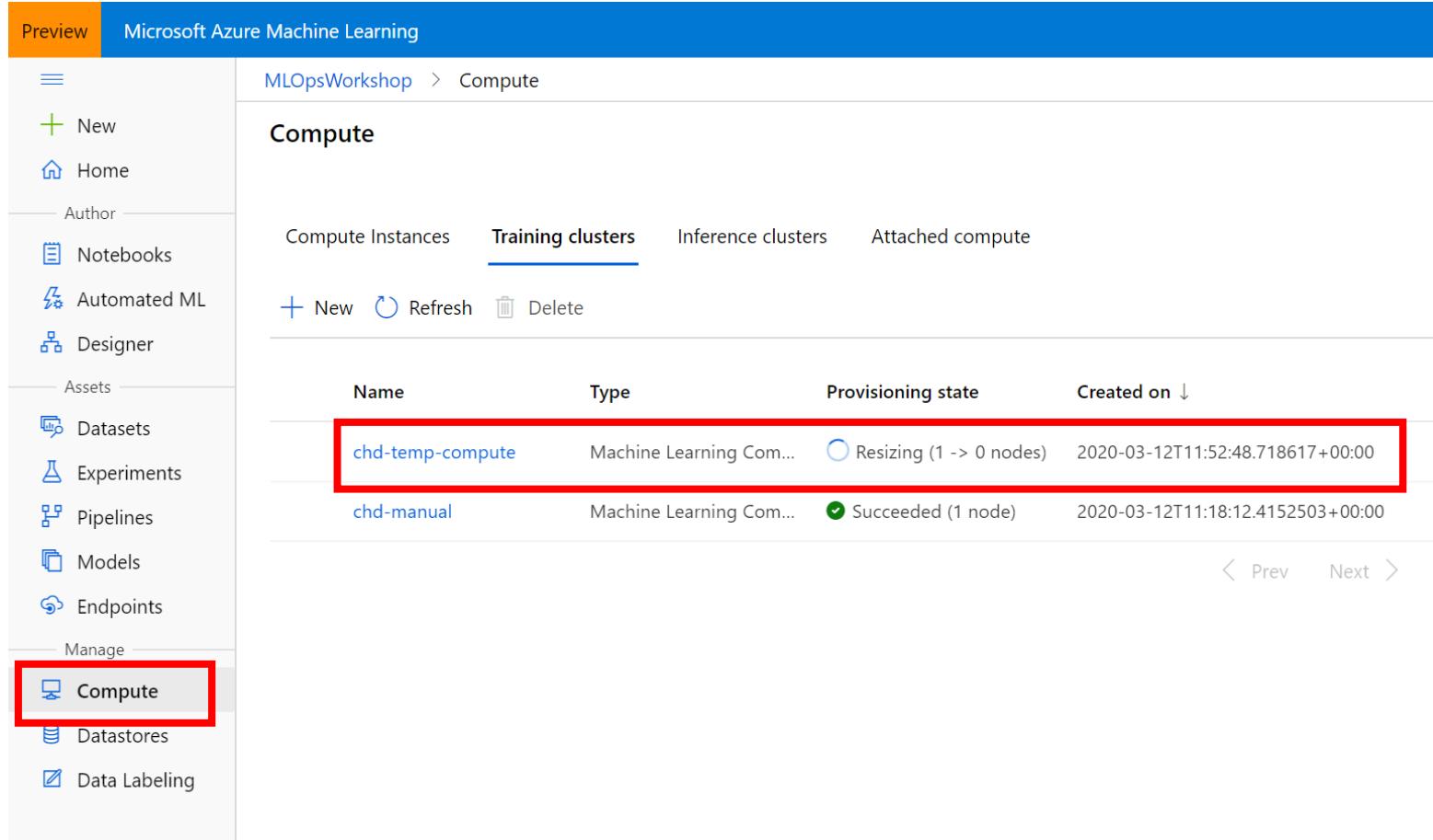
# Fin de l'exécution du pipeline



The screenshot shows the Azure DevOps Pipelines interface for the project "Projet MLOps". The left sidebar lists various project management and development tools: Overview, Boards, Repos, Pipelines (selected), Pipelines, Environments, Releases, Library, Task groups, Deployment groups, and Test Plans. The main area displays two completed pipeline runs under the "Runs" tab:

Description	Stages	Created	Last Run
#20200312.2 Updated train.py (Change max_depth=3 vs 2) ↳ Individual CI ↳ master 0a9d319 ✅	✓	23m ago	21m 53s
#20200312.1 Set up CI with Azure Pipelines ↳ Individual CI ↳ master dc052d5 ✅	✓	1h ago	24m 13s

# Visualisation depuis Azure ML Studio du training cluster créé par le pipeline



The screenshot shows the Microsoft Azure Machine Learning Studio interface. The top navigation bar has a blue header with the text "Microsoft Azure Machine Learning". On the far left, there is a vertical sidebar with several sections: "Author" (New, Home, Notebooks, Automated ML, Designer), "Assets" (Datasets, Experiments, Pipelines, Models, Endpoints), and "Manage" (Compute, Datastores, Data Labeling). The "Compute" item in the "Manage" section is highlighted with a red box. The main content area is titled "Compute" and shows four tabs: "Compute Instances", "Training clusters" (which is underlined in blue, indicating it is active), "Inference clusters", and "Attached compute". Below the tabs are buttons for "+ New", "Refresh", and "Delete". The main table lists two entries:

Name	Type	Provisioning state	Created on
chd-temp-compute	Machine Learning Com...	<input type="radio"/> Resizing (1 -> 0 nodes)	2020-03-12T11:52:48.718617+00:00
chd-manual	Machine Learning Com...	<input checked="" type="radio"/> Succeeded (1 node)	2020-03-12T11:18:12.4152503+00:00

At the bottom right of the table, there are navigation arrows for "Prev" and "Next".

# Visualisation de l'expérimentation créée par le pipeline

The screenshot shows the Microsoft Azure Machine Learning Studio interface. The left sidebar has a navigation menu with the following items:

- Preview
- New
- Home
- Author
- Notebooks
- Automated ML
- Designer
- Assets
- Datasets
- Experiments** (highlighted with a red box)
- Pipelines
- Models
- Endpoints
- Manage
- Compute
- Datastores
- Data Labeling

The main content area displays the "Experiments" page for the "MLOpsWorkshop" workspace. The page title is "Experiments". There are buttons for Refresh, Archive experiment, and View archived experiments. A "Add filter" button is also present.

The table lists two experiments:

Experiment	Latest run	Last submitted	Created	Created by	Run types
chd-prediction	8	Mar 12, 2020 1:32 PM	Mar 12, 2020 12:51 PM	Service principal	Script, Pipeline
chd-prediction-manual	2	Mar 12, 2020 12:27 PM	Mar 12, 2020 12:18 PM	Serge Retkowsky	Script, Automated ML

At the bottom, there are navigation links for Prev and Next.

# Visualisation de l'expérimentation créée par le pipeline

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Experiments > chd-prediction

chd-prediction

Switch to old experience ?

Edit table Refresh Reset to default view Include child runs

Add filter

Run status

Status	Count
Running	0
Completed	6
Failed	0
Other	0

Runs by compute

Duration

Count of runs

local null

Compute target

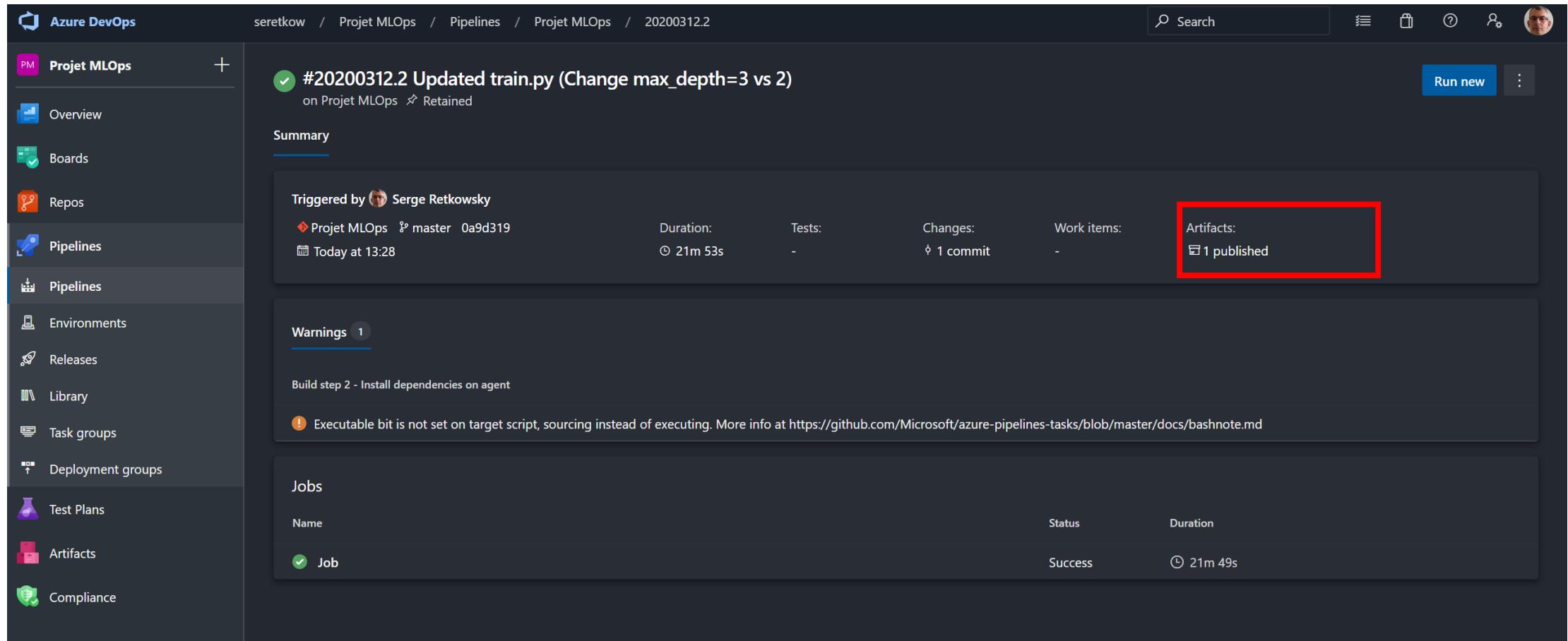
Duration (s)

Run number

Run	Run ID	Status	Submitted time	Duration	Submitted by	Compute target	Run type	Tags
Run 8	89b09638-0be0-4cbc-9fdc-6e06462776...	Completed	Mar 12, 2020 1:32 PM	16m 38s	Service principal	Pipeline	azureml.pipelineComponent: pipelinerun	

New Home Author Notebooks Automated ML Designer Datasets Experiments Pipelines Models Endpoints Compute Datastores Data Labeling

# Accès aux artifacts



The screenshot shows the Azure DevOps Pipelines summary page for a build named #20200312.2. The build was triggered by Serge Retkowsky on the master branch (commit 0a9d319) at 13:28 today. The build duration was 21m 53s, with one commit and no tests or work items. One artifact was published. The 'Artifacts' section is highlighted with a red box.

#20200312.2 Updated train.py (Change max\_depth=3 vs 2)  
on Projet MLOps Retained

Triggered by  Serge Retkowsky

Projet MLOps master 0a9d319 Duration: 21m 53s Tests: - Changes: 1 commit Work items: -

Artifacts: 1 published

Warnings 1

Build step 2 - Install dependencies on agent

Executable bit is not set on target script, sourcing instead of executing. More info at <https://github.com/Microsoft/azure-pipelines-tasks/blob/master/docs/bashnote.md>

Jobs

Name	Status	Duration
Job	Success	21m 49s

# Visualisation des artifacts du pipeline

The screenshot shows the Azure DevOps interface for the 'Projet MLOps' project. The left sidebar navigation bar is visible, with 'Pipelines' selected. The main content area displays the 'Artifacts' page for a specific pipeline run. The URL in the browser bar indicates the run ID: '20200312.2 / publishedArtifacts'. The page title is 'Artifacts'.

The 'Published' section lists the contents of the artifact 'chd-predictor-build-artifact'. The table shows the following files:

Name	Size
chd-predictor-build-artifact	2 MB
.amlignore	73 B
.azureml	22 KB
.git	501 KB
01 Analyse ML Python.ipynb	50 KB
02 Modélisation avec Azure ML.ipynb	168 KB
03 Test déploiement AKS.ipynb	4 KB
04 Télémétrie AKS.ipynb	9 KB
build-master-pipeline.yml	3 KB
framingham.csv	188 KB
outputs	232 B
README.md	317 B
scripts-ml	15 KB
scripts-operationalize	16 KB

# Création d'un Pipeline Release

The screenshot shows the Azure DevOps interface for the 'Projet MLOps' project. The left sidebar menu is visible, with the 'Releases' item highlighted by a red box. The main content area displays a message: 'No release pipelines found'. Below this message is a call-to-action: 'Automate your release process in a few easy steps with a new pipeline'. A blue button labeled 'New pipeline' is also highlighted with a red box.

Azure DevOps

seretkow / Projet MLOps / Pipelines / Releases

Projet MLOps +

- Overview
- Boards
- Repos
- Pipelines
- Pipelines
- Environments
- Releases**
- Library
- Task groups
- Deployment groups
- Test Plans
- Artifacts
- Compliance

No release pipelines found

Automate your release process in a few easy steps with a new pipeline

New pipeline

# Création Empty Job

The screenshot shows the Azure DevOps interface for creating a new release pipeline. On the left, the navigation bar includes 'Projet MLOps' and 'Pipelines'. The main area displays 'All pipelines > New release pipeline'. The pipeline structure is shown with 'Artifacts' and 'Stages' sections. A modal window titled 'Stage 1 Select a template' is open, with a red box highlighting the 'Empty job' option in the search bar.

**Select a template**  
Or start with an **Empty job**

**Featured**

- Azure App Service deployment**  
Deploy your application to Azure App Service. Choose from Web App on Windows, Linux, containers, Function Apps, or WebJobs.
- Deploy a Java app to Azure App Service**  
Deploy a Java application to an Azure Web App.
- Deploy a Node.js app to Azure App Service**  
Deploy a Node.js application to an Azure Web App.
- Deploy a PHP app to Azure App Service and Azure Database for MySQL**  
Deploy a PHP application to an Azure Web App and database to Azure Database for MySQL.
- Deploy a Python app to Azure App Service and Azure database for MySQL**  
Deploy a Python Django, Bottle, or Flask application to an Azure Web App and database to Azure Database for MySQL.
- Deploy to a Kubernetes cluster**  
Deploy, configure, update your containerized applications to a Kubernetes cluster.
- IIS website and SQL database deployment**  
Deployment Group: Deploy ASP.NET or ASP.NET Core web applications to IIS Websites and SQL databases.

# Pipeline

Azure DevOps    seretkov / Projet MLOps / Pipelines / Releases    Search    Save    Create release    View releases    ...

All pipelines > New release pipeline

Pipeline Tasks Variables Retention Options History

Artifacts | + Add    Stages | + Add ▾

+ Add an artifact    Stage 1 1 job, 0 task

Schedule not set

Stage Stage 1

Properties Name and owners of the stage

Stage name Stage 1

Stage owner Serge Retkowsky

The screenshot shows the Azure DevOps interface for creating a new release pipeline. On the left, the navigation bar includes 'Project MLOps', 'Overview', 'Boards', 'Repos', 'Pipelines' (selected), 'Environments', 'Releases', 'Library', 'Task groups', 'Deployment groups', 'Test Plans', 'Artifacts', and 'Compliance'. The main area displays 'All pipelines > New release pipeline'. It has tabs for 'Pipeline', 'Tasks', 'Variables', 'Retention', 'Options', and 'History'. Below these are sections for 'Artifacts' and 'Stages'. The 'Artifacts' section has a '+ Add' button and a note 'Schedule not set'. The 'Stages' section shows a single stage named 'Stage 1' with '1 job, 0 task'. A modal window on the right provides details for 'Stage 1', including its name, owner (Serge Retkowsky), and options to 'Delete' or 'Move'. The top navigation bar also includes a search bar, save button, create release button, view releases button, and more options.

# Nouveau nom du pipeline

The screenshot shows the Azure DevOps interface for managing pipelines. On the left, there's a sidebar with navigation links: Overview, Boards, Repos, Pipelines (which is selected), Environments, Releases, and Library. The main area displays the 'Projet MLOps' project under 'All pipelines'. A pipeline named 'Deploy REST Service & Test' is highlighted with a red box. Below the pipeline name, there are tabs for Pipeline, Tasks, Variables, Retention, Options, and History. The Pipeline tab is active. The pipeline interface includes sections for 'Artifacts' (with a '+ Add' button) and 'Stages' (with a '+ Add' dropdown). A single stage named 'Stage 1' is shown, which contains '1 job, 0 task'. A 'Schedule not set' message is also visible.

# Ajout Artifact

The screenshot shows the Azure DevOps interface for managing pipelines. On the left, the navigation bar includes 'Project MLOps' (selected), 'Overview', 'Boards', 'Repos', 'Pipelines' (selected), 'Environments', 'Releases', 'Library', 'Task groups', 'Deployment groups', 'Test Plans', 'Artifacts' (highlighted with a red box), and 'Compliance'. The main area displays 'All pipelines > Deploy REST Service & Test'. The pipeline interface shows 'Artifacts' and 'Stages'. A red box highlights the '+ Add' button under 'Artifacts'. To the right, a modal window titled 'Add an artifact' is open, showing the 'Source type' section with 'Build' selected (also highlighted with a red box). Other options include 'Azure Repos ...', 'GitHub', and 'TFVC'. The 'Project' dropdown is set to 'Project MLOps'. The 'Source (build pipeline)' dropdown is also set to 'Project MLOps'. The 'Default version' is set to 'Latest'. The 'Source alias' field contains '\_Project MLOPs'. A note at the bottom states: 'The artifacts published by each version will be available for deployment in release pipelines. The latest successful build of Project MLOps published the following artifacts: chd-predictor-build-artifact.' A blue 'Add' button is at the bottom of the modal.

# Ajout Artifact

The screenshot shows the Azure DevOps interface for managing pipelines. The left sidebar lists various project management and development tools, with 'Pipelines' selected. The main area displays a pipeline named 'Deploy REST Service & Test'. The pipeline structure is divided into two main sections: 'Artifacts' and 'Stages'.

**Artifacts:** This section contains a single artifact named '\_Projet MLOps'. A tooltip indicates that the schedule is not set. An 'Add' button is available for creating more artifacts.

**Stages:** This section contains one stage named 'Stage 1'. It is described as having '1 job, 0 task'. A tooltip indicates that users can add or remove users from the stage. An 'Add' button is available for creating more stages.

The top navigation bar shows the current path: All pipelines > Deploy REST Service & Test. The pipeline tab is active, along with other tabs for Tasks, Variables, Retention, Options, and History.

# Paramétrage

The screenshot shows the Azure Pipelines interface for the 'Deploy REST Service & Test' pipeline. On the left, the pipeline structure is displayed with an 'Artifacts' section containing '\_Projet MLOps' and a 'Stages' section with 'Stage 1'. A red box highlights the '\_Projet MLOps' artifact. Stage 1 contains a task icon and a user icon. Below the pipeline view, there are two trigger configurations:

- Continuous deployment trigger**: Build: '\_Projet MLOps'. Enabled. Creates a release every time a new build is available.
- Pull request trigger**: Build: '\_Projet MLOps'. Disabled. Enabling this will create a release every time a selected artifact is available as part of a pull request workflow.

# Ajout Agent Job

The screenshot shows the Azure DevOps Pipelines interface for the 'Projet MLOps' project. The pipeline is named 'Deploy REST Service & Test' and consists of a single stage named 'Stage 1: Deployment process'. Within this stage, there is an 'Agent job' task, which is currently selected. A modal dialog is open on the right side of the screen, titled 'Agent job', used for configuring the job settings.

**Agent job configuration fields:**

- Display name \***: Agent job
- Agent selection ^**
- Agent pool**: Azure Pipelines
- Agent Specification \***: ubuntu-18.04
- Demands**: (empty)
- Execution plan ^**
- Parallelism**: None

**Navigation and other UI elements:**

- Left sidebar: Overview, Boards, Repos, Pipelines (selected), Pipelines, Environments, Releases, Library, Task groups, Deployment groups (selected), Test Plans, Artifacts, Compliance.
- Top navigation bar: seretkov / Projet MLOps / Pipelines / Releases, Search, Save, Create release, View releases, More options.

# Ajout Python version

The screenshot shows the Azure Pipelines interface for a pipeline named "Deploy REST Service & Test". The pipeline has one stage, "Stage 1: Deployment process", which contains an "Agent job" step. The "Tasks" tab is selected. A search bar at the top right shows the query "python". The results list includes:

- Python script**: Run a Python file or inline script
- Use Python version**: Use the specified version of Python from the tool cache, optionally adding it to the PATH (this item is highlighted with a red box)
- Python pip authenticate**: Authentication task for the pip client used for installing Python distributions
- Python twine upload authenticate**: Authenticate for uploading Python distributions using twine. Add '-r FeedName/EndpointName --config-file \${PYPIRC\_PATH}' to your twine upload command. For feeds present in this organization, use the feed name as the repository (-r). Otherwise, use the endpoint name defined in the service connection.
- Azure Functions**: Update a function app with .NET, Python, JavaScript, PowerShell, Java based web applications

# Paramétrage Python version

The screenshot shows the Azure Pipelines interface for a pipeline named "Deploy REST Service & Test". The "Tasks" tab is selected. On the left, there is a stage named "Stage 1" with an "Agent job" step containing a task named "Install Python 3.6". This task is highlighted with a blue background. On the right, the configuration details for this task are displayed:

- Use Python version**: Task version is set to "0.\*".
- Display name \***: Set to "Install Python 3.6".
- Version spec \***: Set to "3.6".
- Add to PATH**: A checked checkbox.
- Advanced**, **Control Options**, and **Output Variables** sections are also visible.

# Ajout Bash

The screenshot shows the Azure Pipelines interface for a pipeline named "Deploy REST Service & Test". The pipeline has one stage, "Stage 1", which contains an "Agent job" step. This step includes a task named "Install Python 3.6". On the right side of the screen, the "Tasks" blade is open, showing a search bar with "bash" typed in. A red box highlights the "Bash" task card, which is described as "Run a Bash script on macOS, Linux, or Windows". Below it is the "Command line" task, which runs a command-line script on Linux, macOS, and Windows. Further down are the "Script Retryer tool" and "Applitools Eyes Integration" tasks.

seretkov / Projet MLOps / Pipelines / Releases

All pipelines > Deploy REST Service & Test

Pipeline Tasks Variables Retention Options History

Stage 1 Deployment process

Agent job Run on agent

Install Python 3.6 Use Python version

Add tasks Refresh

bash

Bash

Run a Bash script on macOS, Linux, or Windows

Command line

Run a command line script using Bash on Linux and macOS and cmd.exe on Windows

Marketplace

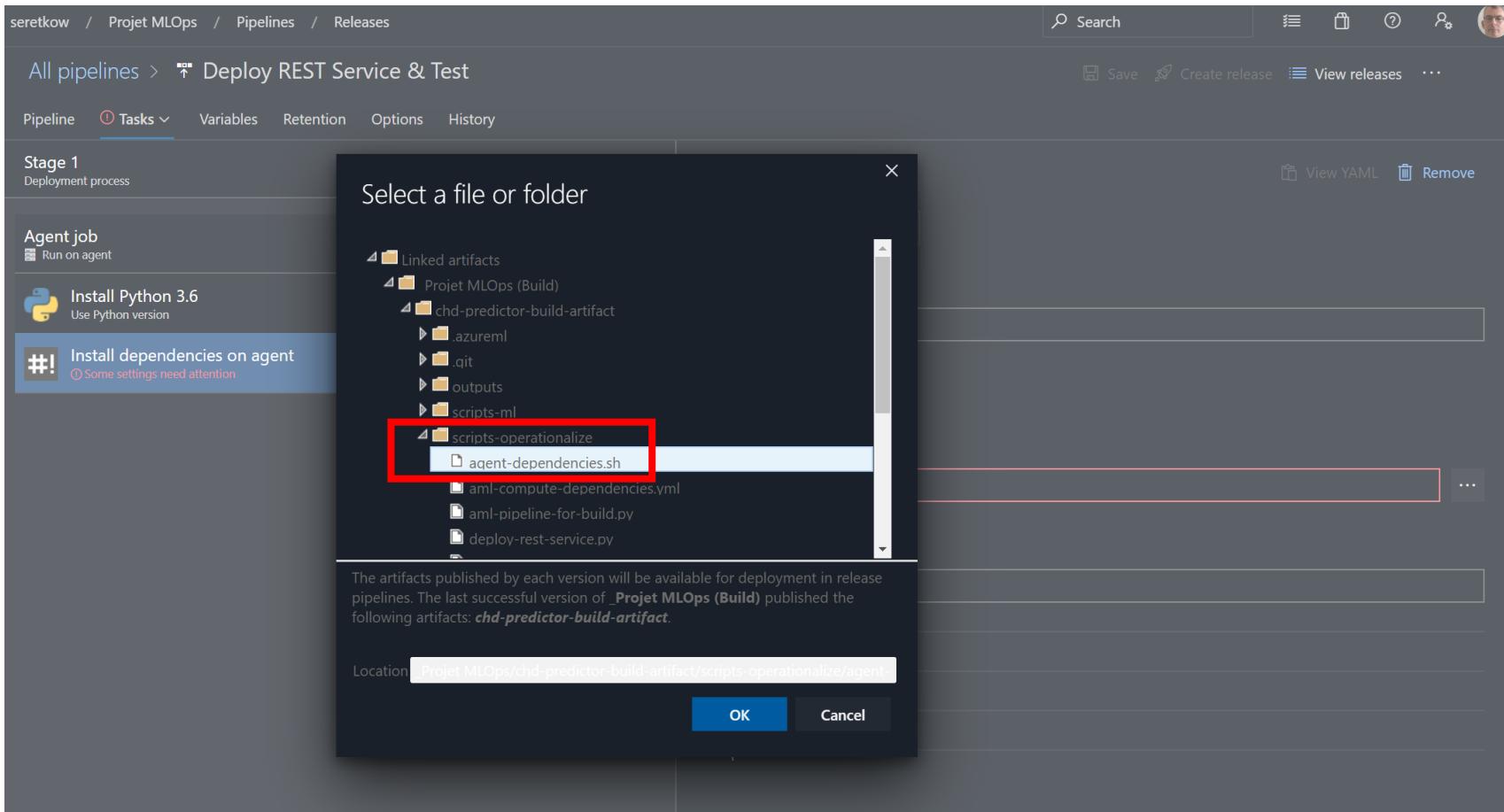
Script Retryer tool

That task can be used to retry several times with some delay specified script. Inline script supported. Supported languages: bash, powershell, python. It works fine on Unix and Windows agents.

Applitools Eyes Integration

Displays the Applitools Dashboard in the build summary page.

# Paramétrage Bash



# Paramétrage Bash

The screenshot shows the Azure DevOps Pipeline interface for a project named "Projet MLOps". The pipeline is titled "Deploy REST Service & Test" and consists of three stages: "Stage 1" (Deployment process), "Agent job" (Run on agent), and "Install dependencies on agent". The third stage is currently selected and highlighted in blue. On the right side, the configuration for the "Install dependencies on agent" task is displayed. The task is of type "Bash" (version 3.\*). The "Display name" is set to "Install dependencies on agent". The "Type" is set to "File Path". The "Script Path" field contains the value `$(System.DefaultWorkingDirectory)/_Projet MLOps/chd-predictor-build-artifact/scripts-operationalize/agent-dependencies.sh`. This entire section is highlighted with a red box.

# Ajout Azure CLI

The screenshot shows the Azure DevOps Pipeline interface for a project named "Projet MLOps". The pipeline is titled "Deploy REST Service & Test". The "Tasks" tab is selected, showing the current stages and their tasks:

- Stage 1: Deployment process**
  - Agent job** (Run on agent)
  - Install Python 3.6** (Use Python version)
  - Install dependencies on agent** (Bash)

To the right, the "Add tasks" pane is open, displaying a search bar with "cli" typed in. A red box highlights the "Azure CLI" task, which is described as running Azure CLI commands against an Azure subscription in a Shell script on Linux or a Batch script on Windows. Other visible tasks include "Docker CLI installer" and "Python pip authenticate".

# Paramétrage Azure CLI

The screenshot shows the Azure DevOps Pipeline editor. On the left, the pipeline structure is visible with Stage 1 containing an Agent job with three tasks: Install Python 3.6, Install dependencies on agent, and Deploy and test the REST service. The Deploy and test the REST service task is currently selected, highlighted with a blue background. On the right, the task configuration pane is open, showing the following fields:

- Display name \*: Deploy and test the REST service
- Azure subscription \*: ncrMlopsServiceConnection
- Script Location \*: Inline script
- Inline Script \*:

```
python scripts-operationalize/deploy-rest-service.py --service_name $(service_name) --aks_name $(aks_name) --aks_region $(aks_region) --description $(description)
```

A red box highlights the inline script field.

**python scripts-operationalize/deploy-rest-service.py --service\_name \$(service\_name) --aks\_name \$(aks\_name) --aks\_region \$(aks\_region) --description \$(description)**

# Paramétrage Azure CLI

The screenshot shows the Azure DevOps Pipeline editor interface. The pipeline is named "Deploy REST Service & Test". The "Tasks" tab is selected. A single task, "Deploy and test the REST service" (Azure CLI), is visible in the list. The task configuration pane on the right contains the following details:

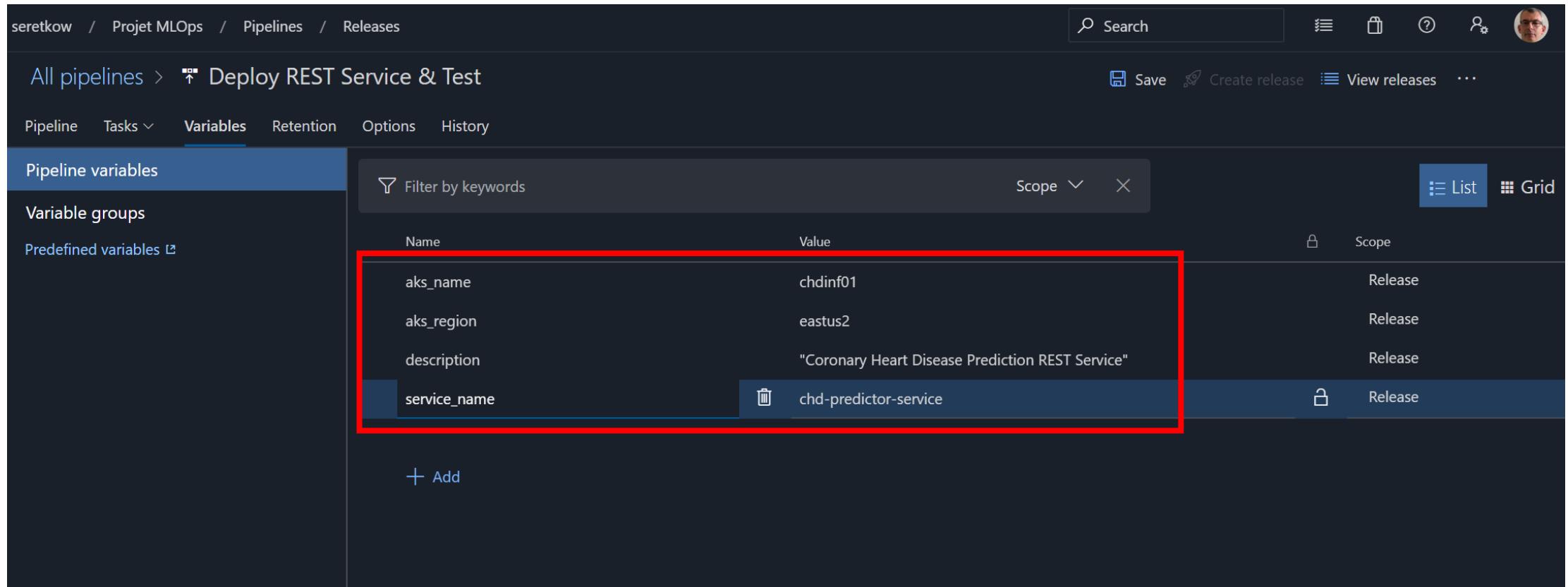
- Script:** python scripts-operationalize/deploy-rest-service.py --service\_name \$(service\_name) --aks\_name \$(aks\_name) --aks\_region \$(aks\_region) --description \$(description)
- Arguments:** An empty text input field.
- Advanced Options:**
  - Access service principal details in script
  - Use global Azure CLI configuration
- Working Directory:** \$(System.DefaultWorkingDirectory)/\_Projet MLOps/chd-predictor-build-artifact
- Control Options:** An expandable section.
- Environment Variables:** An expandable section.
- Output Variables:** An expandable section.

# Accès définition des variables

The screenshot shows the 'Variables' tab selected in the navigation bar of the Azure Pipelines interface. A red box highlights the 'Variables' tab. The page displays a list of pipeline variables with columns for Name and Value. There is a 'Scope' dropdown and a 'Filter by keywords' search bar at the top right. A blue 'List' button is also visible. On the left sidebar, there are sections for 'Pipeline variables', 'Variable groups', and 'Predefined variables'. The main title above the table is 'All pipelines > Deploy REST Service & Test'.

Name	Value	Scope	Scope
+ Add			

# Ajout des variables



The screenshot shows the 'Variables' tab in the Azure DevOps Pipelines interface. The pipeline is named 'Deploy REST Service & Test'. The 'Pipeline variables' section displays four variables:

Name	Value	Scope
aks_name	chdinf01	Release
aks_region	eastus2	Release
description	"Coronary Heart Disease Prediction REST Service"	Release
service_name	chd-predictor-service	Release

The last row, 'service\_name', is highlighted with a red box.

# Sauvegarde

The screenshot shows a software interface for managing pipeline variables and creating releases.

**Pipeline Variables Screen:**

- Path: seretkow / Projet MLOps / Pipelines / Releases
- Current Pipeline: Deploy REST Service & Test
- Variables Tab selected
- Table of Pipeline variables:

Name	Value
aks_name	chdinf01
aks_region	eastus2
description	"Coronary Heart Disease Prediction Model"
service_name	chd-predictor
- Add button (+ Add)

**Create a new release Dialog:**

- Title: Create a new release  
Deploy REST Service & Test
- Section: Pipeline ▲  
Click on a stage to change its trigger from automated to manual.  
Stage 1 (Automated trigger)
- Section: Artifacts ▲  
Select the version for the artifact sources for this release  
Source alias: \_Projet MLOps  
Version: 20200312.2
- Section: Release description (Empty text area)

# Accès Release-1

The screenshot shows the Azure DevOps interface for managing pipeline variables. The top navigation bar includes 'seretkow / Projet MLOps / Pipelines / Releases'. The main title is 'All pipelines > Deploy REST Service & Test'. A green notification bar at the top states 'Release Release-1 has been created'.

The left sidebar has tabs for 'Pipeline', 'Tasks', 'Variables' (which is selected), 'Retention', 'Options', and 'History'. The 'Variables' section contains a 'Pipeline variables' table:

Name	Value	Scope
aks_name	chdinf01	Release
aks_region	eastus2	Release
description	"Coronary Heart Disease Prediction REST Service"	Release
service_name	chd-predictor-service	Release

At the bottom of the table, there is a '+ Add' button.

# Release-1

seretkov / Projet MLOps / Pipelines / Releases / Deploy REST Service & Test / Release-1

↑ Deploy REST Service & Test > Release-1

Pipeline Variables History + Deploy Cancel Refresh Edit ...

**Release**

Manually triggered  
by  Serge Retkowsky  
12/03/2020 à 14:14

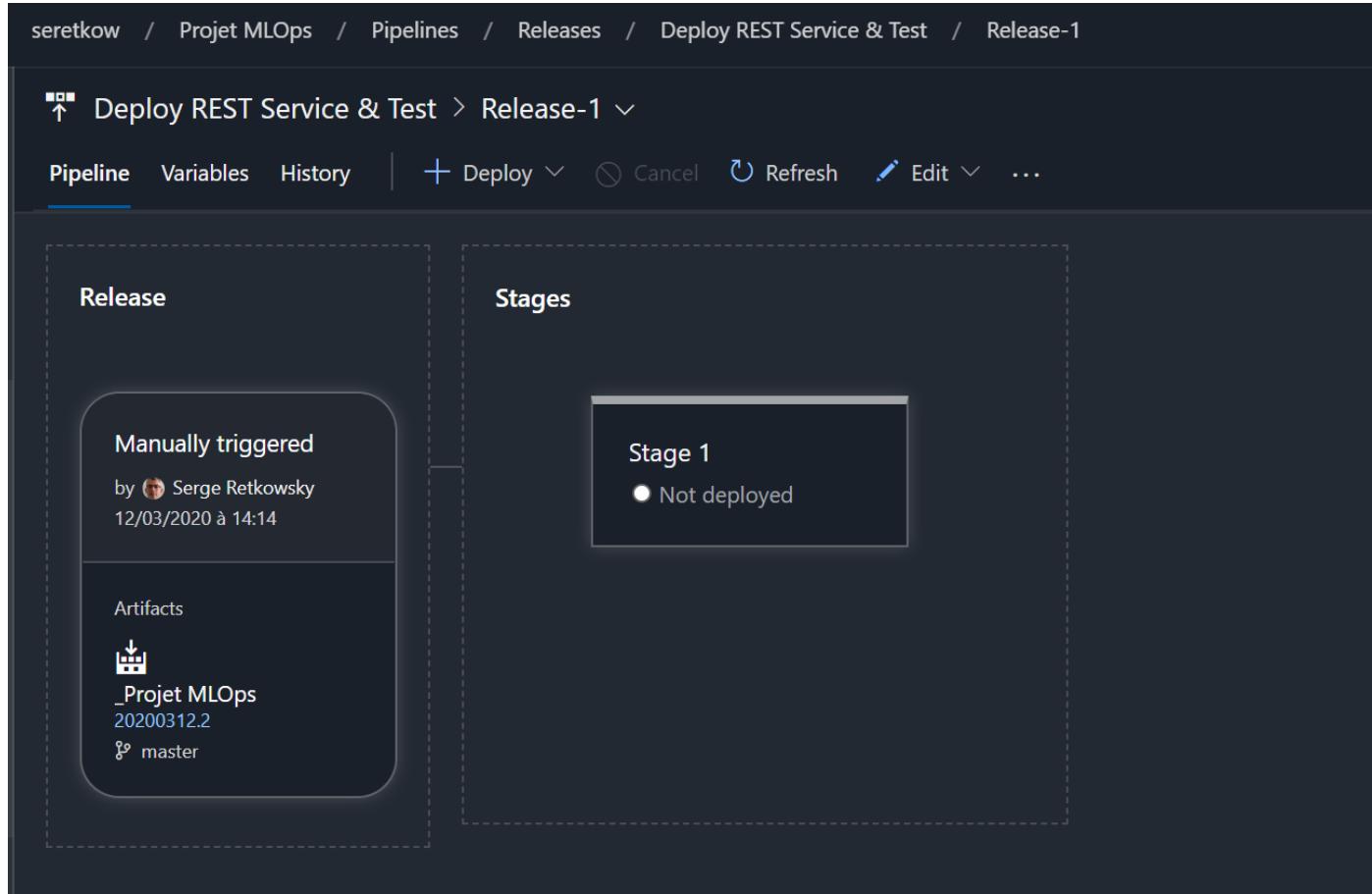
Artifacts

 \_Projet MLOps  
20200312.2  
g master

**Stages**

Stage 1

● Not deployed



# Deploy

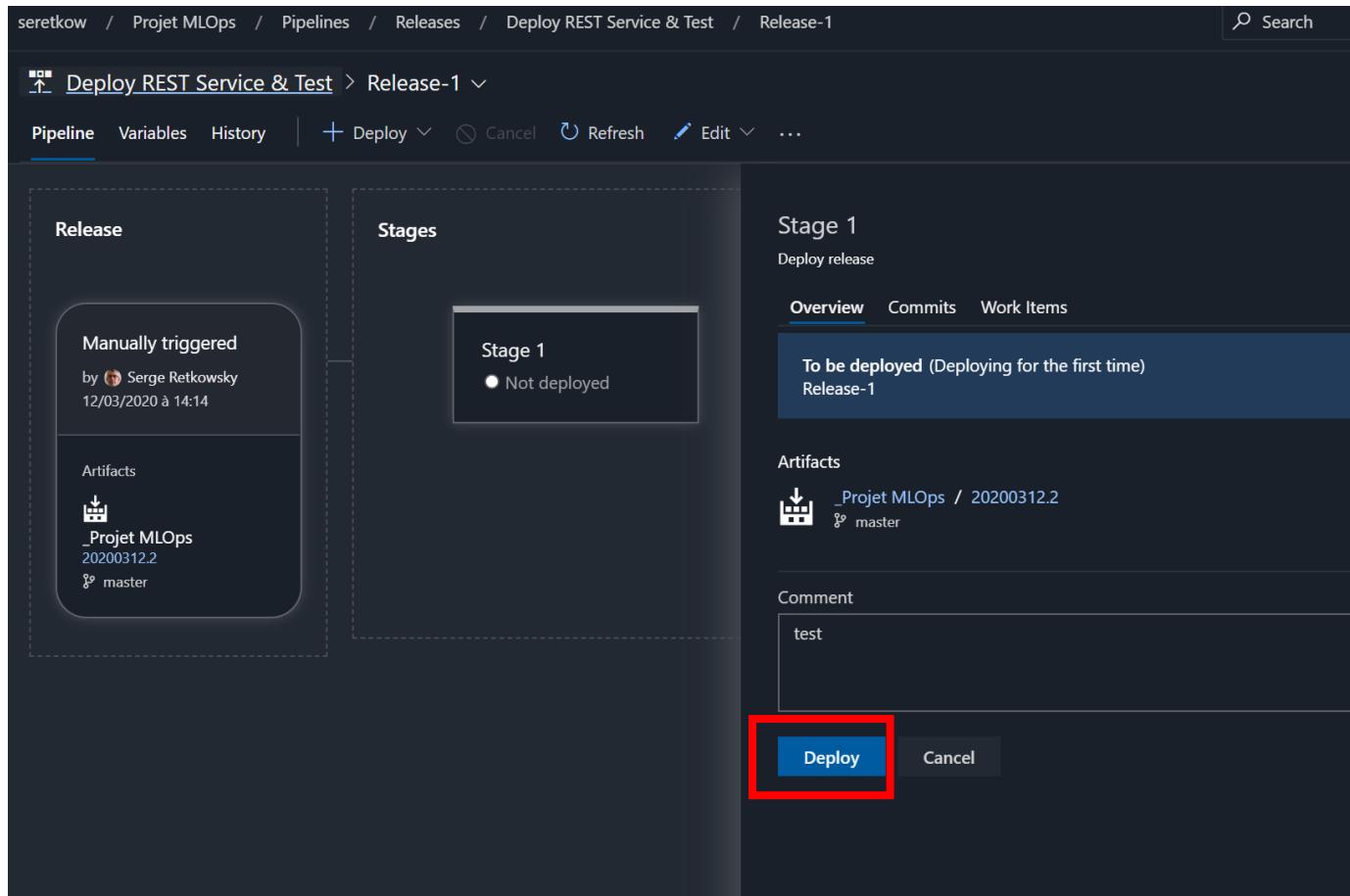
The screenshot shows the Azure DevOps interface for a pipeline named "Deploy REST Service & Test" under the "Release-1" release. The left sidebar lists various project management and development tools. The main area displays the release details and stages.

**Release:** Manually triggered by **Serge Retkowsky** on **12/03/2020 à 14:14**. Artifacts listed: **\_Projet MLOps 20200312.2** from branch **master**.

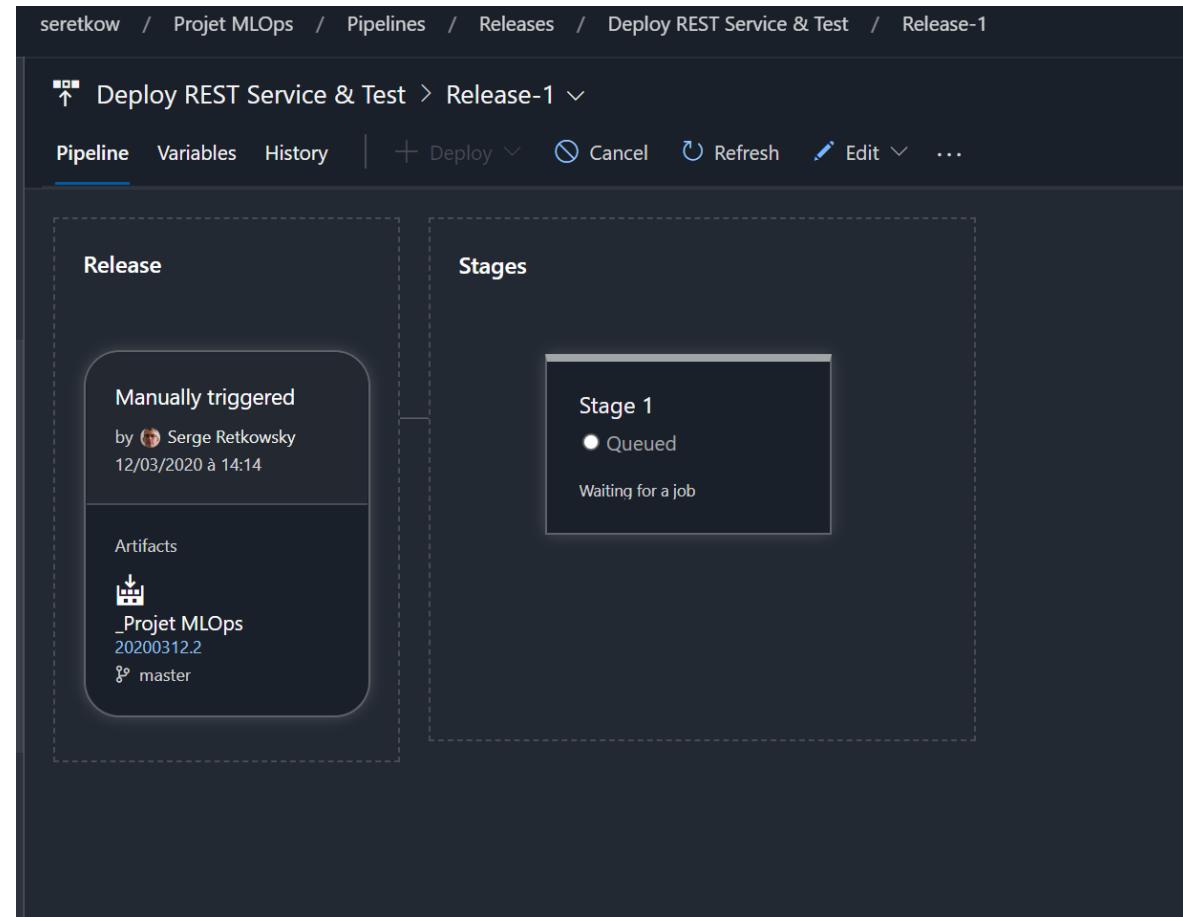
**Stages:** Stage 1 is currently **Not deployed**. A red box highlights the **Deploy** button associated with this stage.

Top navigation bar: seretkov / Projet MLOps / Pipelines / Releases / Deploy REST Service & Test / Release-1. Right side includes a search bar, ribbon icons, and user profile.

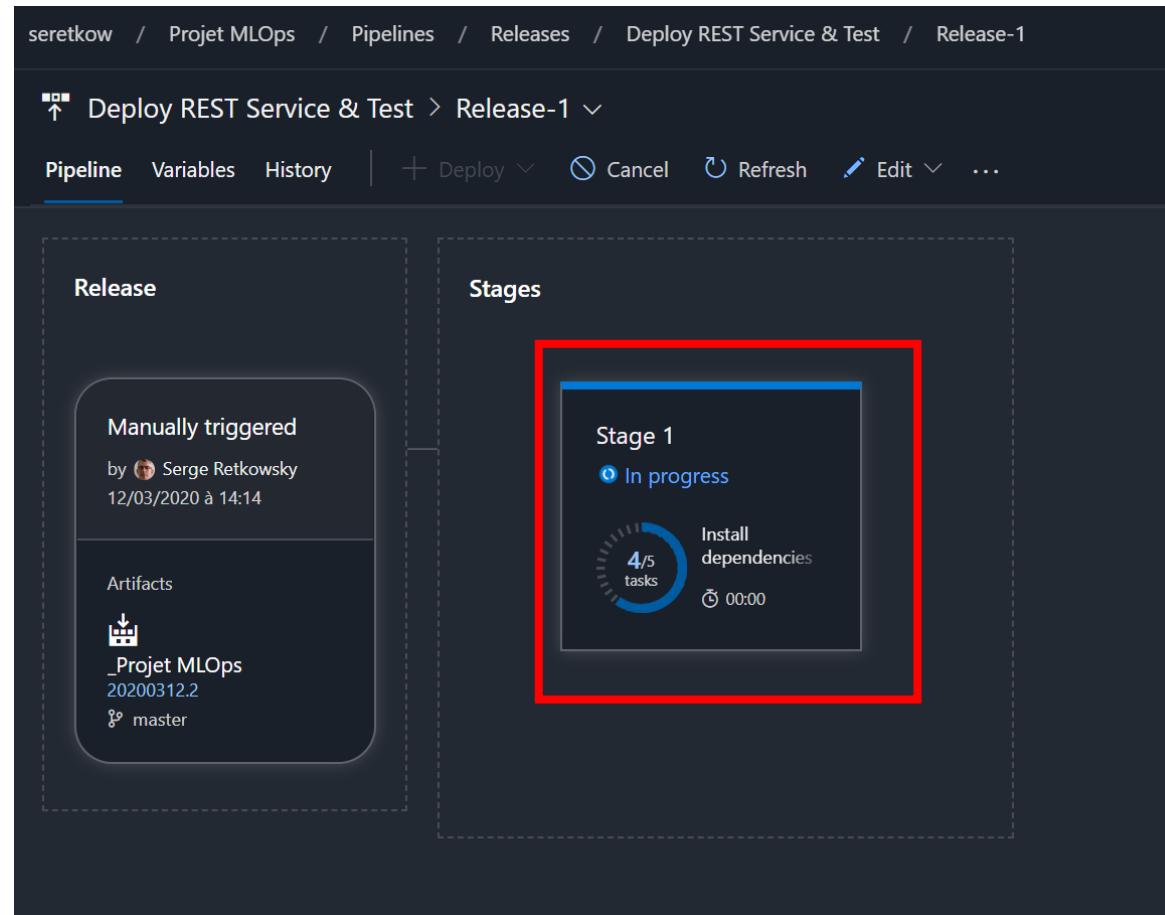
# Deploy



# Exécution du Release pipeline en cours



# Exécution du Release pipeline en cours



# Nous pouvons visualiser la création du cluster AKS par le Release Pipeline

The screenshot shows the Microsoft Azure Machine Learning interface. The left sidebar has a 'Preview' tab selected. Under 'Compute', the 'Inference clusters' tab is active. A red box highlights the first row of the table, which lists a single cluster named 'chdinf01'. The table columns are: Name, Type, Created/Attached, Provisioning state, and Created on. The cluster details are: Name - chdinf01, Type - Kubernetes Service, Created/Attached - Created, Provisioning state - Creating, and Created on - 2020-03-12T13:17:51.8327405+00:00.

Name	Type	Created/Attached	Provisioning state	Created on
chdinf01	Kubernetes Service	Created	Creating	2020-03-12T13:17:51.8327405+00:00

# Exécution du Release Pipeline en cours

The screenshot shows a release pipeline execution interface. The top navigation bar includes 'seretkow / Projet MLOps / Pipelines / Releases / Deploy REST Service & Test / Release-1'. The main title is 'Deploy REST Service & Test > Release-1 > Stage 1' with a status of 'In progress'. Below the title are buttons for 'Pipeline', 'Tasks', 'Variables', 'Logs' (which is selected), 'Tests', 'Deploy', 'Cancel', 'Refresh', 'Download all logs', 'Edit', and '...'. A search bar and user profile are also present.

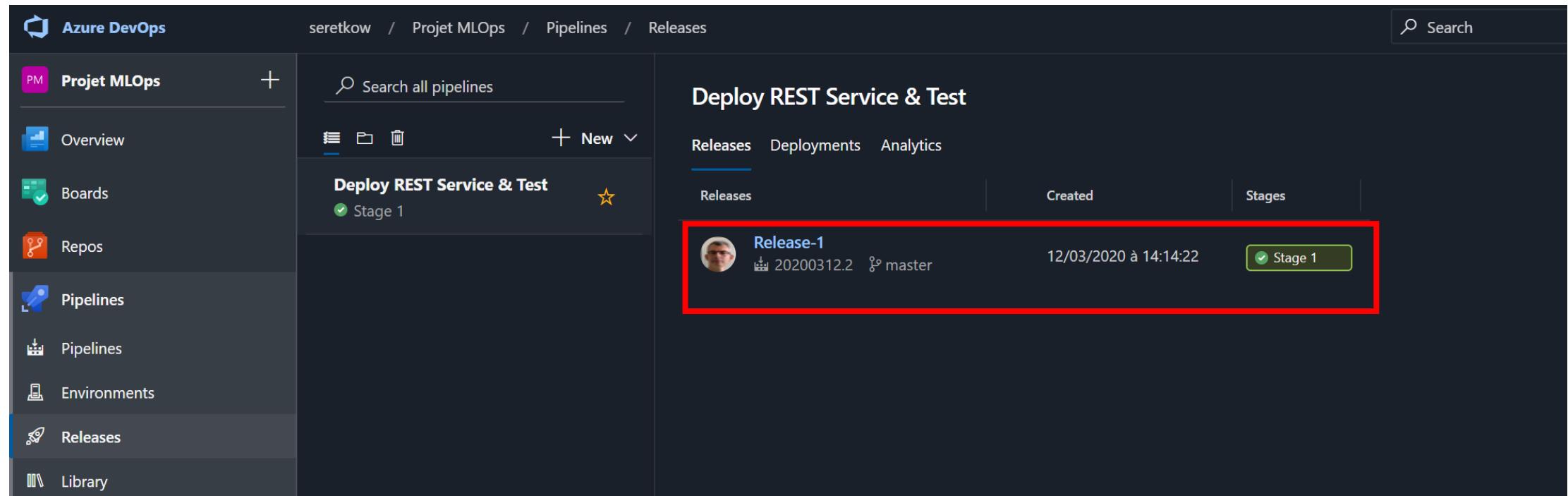
**Deployment process**  
In progress

**Agent job**  
In progress - 1 warning

Task	Status	Duration
Initialize job	succeeded	2s
Download artifact - _Projet MLOps - chd-predictor-build-artifact	succeeded	4s
Install Python 3.6	succeeded	<1s
Install dependencies on agent	succeeded 1 warning	1m 3s
Deploy and test the REST service	In progress	12m 48s

Waiting for console output from an agent...

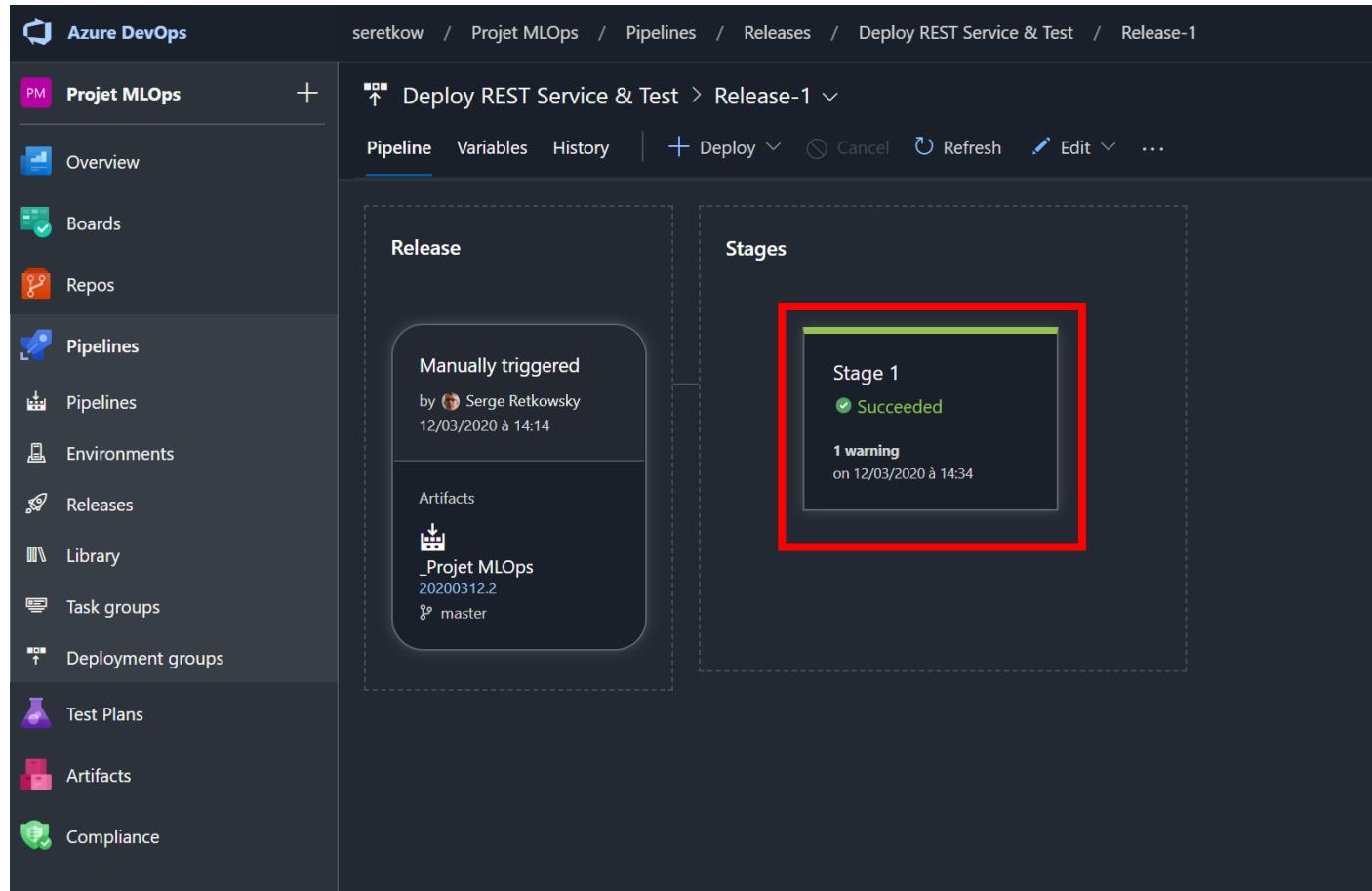
# Fin de l'exécution du Release Pipeline



The screenshot shows the Azure DevOps interface for the 'Projet MLOps' project. The left sidebar highlights the 'Pipelines' section. The main area displays the 'Deploy REST Service & Test' pipeline. A red box highlights the first release entry:

Release	Created	Stages
Release-1 20200312.2	12/03/2020 à 14:14:22	Stage 1

# Fin du traitement



# Visualisation AKS depuis le portail Azure

Home > Kubernetes services

## Kubernetes services

Microsoft

+ Add Manage view Refresh Export to CSV Assign tags Feedback

Filter by name... Subscription == Microsoft Azure Internal Consumption Type == all Resource group == all Location == all Add filter

Showing 1 to 3 of 3 records.

<input type="checkbox"/> Name ↑↓	Type ↑↓	Resource group ↑↓	Location ↑↓	Subscription ↑↓	Location ID ↑↓	Tags	Kuberne... ↑↓
<input type="checkbox"/>  aks-exemple42c702968f	Kubernetes service	AzureMLWorkshopRG	West Europe	Microsoft Azure Interna...	westeurope		1.14.8
<input type="checkbox"/>  akstitanic75344941095	Kubernetes service	titanicworkspaceRG	West Europe	Microsoft Azure Interna...	westeurope		1.14.8
<input type="checkbox"/>  chdinf0130d64001a0fc	Kubernetes service	MLOpsWorkshopRG	East US 2	Microsoft Azure Interna...	eastus2		1.14.8

# Visualisation du real-time endpoint déployé dans AKS

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Endpoints

Endpoints

Real-time endpoints Pipeline endpoints

Refresh Delete Search to filter items...

Name	Description	Created On	Created By	Updated On ↓	Compute Type	Compute Target
chd-predictor-service	Coronary Heart Di...	March 12, 2020 2:32 PM	84611bf7-80f8-41c9-ad30-3c43d...	March 12, 2020 2:34 PM	AKS	chdinf01

< Prev Next >

New Home Author Notebooks Automated ML Designer Assets Datasets Experiments Pipelines Models Endpoints Manage Compute Datastores Data Labeling

# Détails du endpoint

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Endpoints > chd-predictor-service

## chd-predictor-service

Details Consume

Description	Coronary Heart Disease Prediction REST Service	CPU	0.1
Deployment state	Healthy	Memory	0.5 GB
Compute type	AKS	Autoscale enabled	true
Service ID	chd-predictor-service	Min replicas	1
Tags	name: chdinf01 image_id: chd-predictor-image:2	Max replicas	10
Created on	3/12/2020 2:32:48 PM	Target utilization	70%
Last updated on	3/12/2020 2:34:40 PM	Refresh period	1 s
Compute target	chdinf01	App Insights enabled	true
Image ID	chd-predictor-image:2	Event Hubs enabled	false
REST endpoint	[REDACTED]	Storage enabled	true
Key-based authentication enabled	true	Region	eastus2
Token-based authentication enabled	false	Last edited by	N/A
Swagger URI	[REDACTED]	Created by	N/A

# Détails

Preview Microsoft Azure Machine Learning

MLOpsWorkshop > Endpoints > chd-predictor-service

## chd-predictor-service

Details Consume

**Basic consumption info**

REST endpoint  
http://13.77.100.100:5000/predict

Using key  Using token

Primary key  
n0BeJA [Regenerate](#)

Secondary key  
IbNadS [Regenerate](#)

# Workshop Tests et télémétrie



# Test du modèle déployé avec notebook 03 Test déploiement AKS

Ce notebook Python permet de tester le modèle de ML déployé par MLOps.

The screenshot shows a Jupyter Notebook interface with the title "jupyter 03 Test déploiement AKS Last Checkpoint: 3 hours ago (autosaved)". The toolbar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. Below the toolbar is a toolbar with icons for file operations, run, and nbdiff. The date "2020-03-12 09:57:00.588788" is displayed. The notebook content starts with imports:

```
In [6]: import requests  
import json
```

Section 2. Tests call API:

```
In [7]: scoring_REST_uri = "ARENSEIGNER"  
In [8]: REST_api_key = "ARENSEIGNER"  
In [9]: # Set the content type  
headers = {'Content-Type': 'application/json'}  
  
# If authentication is enabled, set the authorization header  
headers['Authorization'] = f'Bearer {REST_api_key}'  
In [10]: toBeScored = {"data": [[61, 1, 150, 103], [43, 1, 180, 99]]}  
resp = requests.post(scoring_REST_uri, json.dumps(toBeScored), headers=headers)  
print("Résultats : ", resp.text)
```

The output for In [10] is "Résultats : "[1, 0]"

Fin

```
In [ ]:
```

# Test du modèle déployé avec notebook 03 Test déploiement AKS

Il est nécessaire de renseigner les informations relatives au endpoint et à la clef.

The screenshot shows a Jupyter Notebook interface with the title "03 Test déploiement AKS". The notebook contains several code cells:

- 1. Infos**
  - In [1]: 

```
import sys
sys.version
```

Out[1]: '3.6.9 |Anaconda, Inc.| (default, Jul 30 2019, 19:07:31) \n[GCC 7.3.0]'
  - In [2]: 

```
import datetime
now = datetime.datetime.now()
print(now)
```

2020-03-12 13:55:44.865632
  - In [3]: 

```
import requests
import json
```
- 2. Tests call API**
  - In [4]: 

```
scoring_REST_uri = "http://[REDACTED]:[REDACTED]/score"
```
  - In [5]: 

```
REST_api_key = "[REDACTED]"
```
  - In [6]: 

```
# Set the content type
headers = {'Content-Type': 'application/json'}
```

# If authentication is enabled, set the authorization header
headers['Authorization'] = f'Bearer {REST\_api\_key}'
  - In [11]: 

```
toBeScored = {"data":[[11, 1, 150],[67, 1, 138]]}
resp = requests.post(scoring_REST_uri,json.dumps(toBeScored), headers=headers)
print("Résultats : ", resp.text)
```

Résultats : "[0, 1]"

# Notebook 04 Télémétrie AKS

Ce notebook Python permet de générer différents appels au service afin de créer de la télémétrie.

The screenshot shows a Jupyter Notebook interface with the title "jupyter 04 Télémétrie AKS Last Checkpoint: 3 hours ago (autosaved)". The notebook contains several code cells and their outputs:

- Cell 1:** Prints workspace configuration success.

```
ws.write_config()  
print('Workspace configuration succeeded')
```

Output: Workspace configuration succeeded
- Section 2: Call du modèle**

```
In [9]: # This is dummy data, just to test the call  
test_data = {"data": [[61, 1, 150], [43, 1, 180]]}
```

```
In [10]: headers = {'Content-Type': 'application/json', 'Authorization': ('Bearer ' + rest_service_key)}  
response = requests.post(rest_service_url, json.dumps(test_data), headers=headers)  
print('Predictions :')  
print(response.text)
```

Predictions :  
"[0, 0]"
- Section 3: Activation télémétrie**

```
In [11]: rest_service = Webservice(ws, rest_service_name)
```

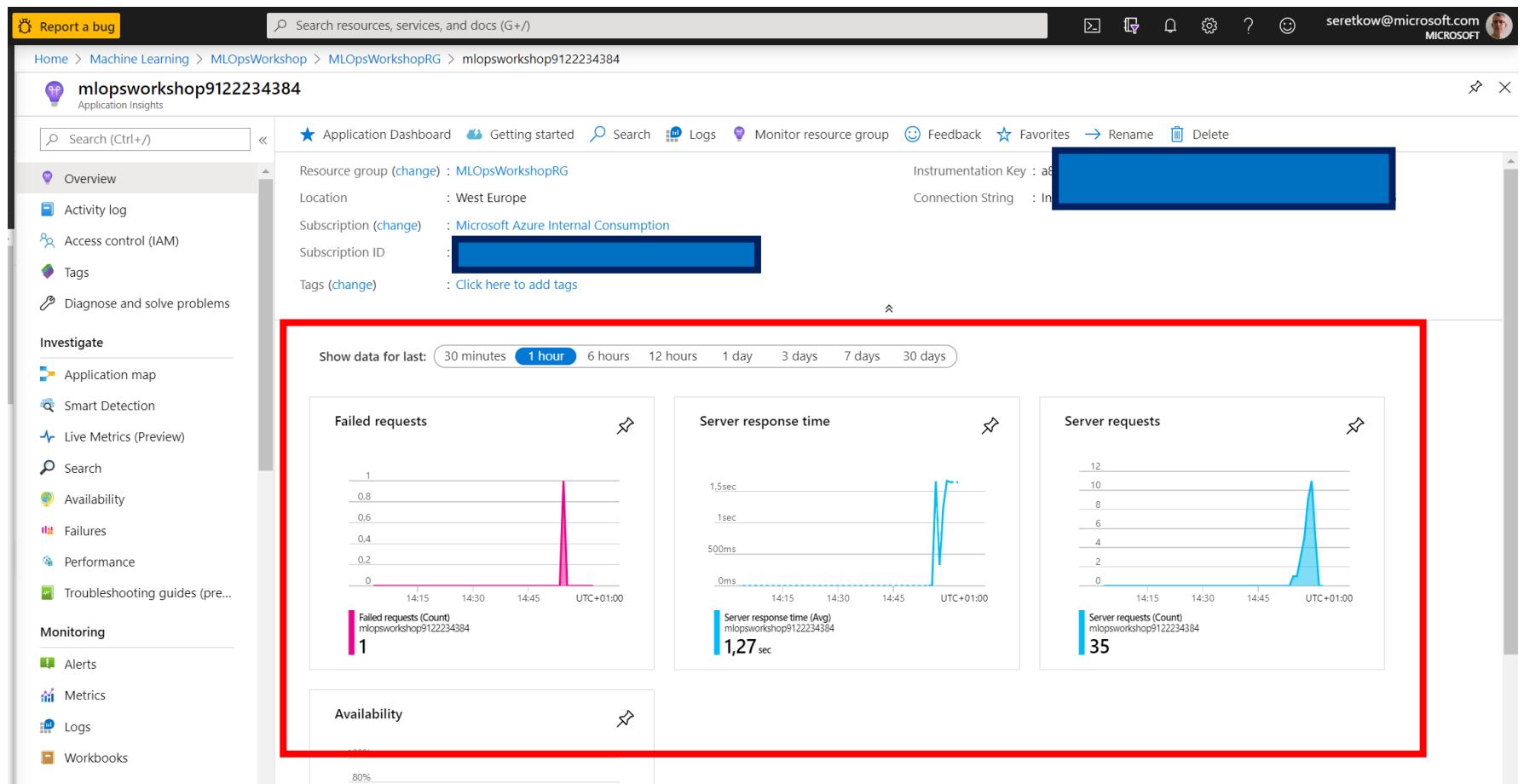
```
In [12]: rest_service.update(enable_app_insights=True, collect_model_data=True)
```
- Section 4: Génération d'appels pour simuler des données de télémétries**

```
In [28]: test_data = {"data": [[44, 1, 138], [43, 1, 150], [67, 1, 138]]}  
  
headers = {'Content-Type': 'application/json', 'Authorization': ('Bearer ' + rest_service_key)}  
response = requests.post(rest_service_url, json.dumps(test_data), headers=headers)  
print('Predictions :')  
print(response.text)
```

Predictions  
"[0, 0, 1]"

# Accès à AppInsights

Accès au service  
AppInsights du  
ressource group  
Azure ML.



# Détails ApplInsights

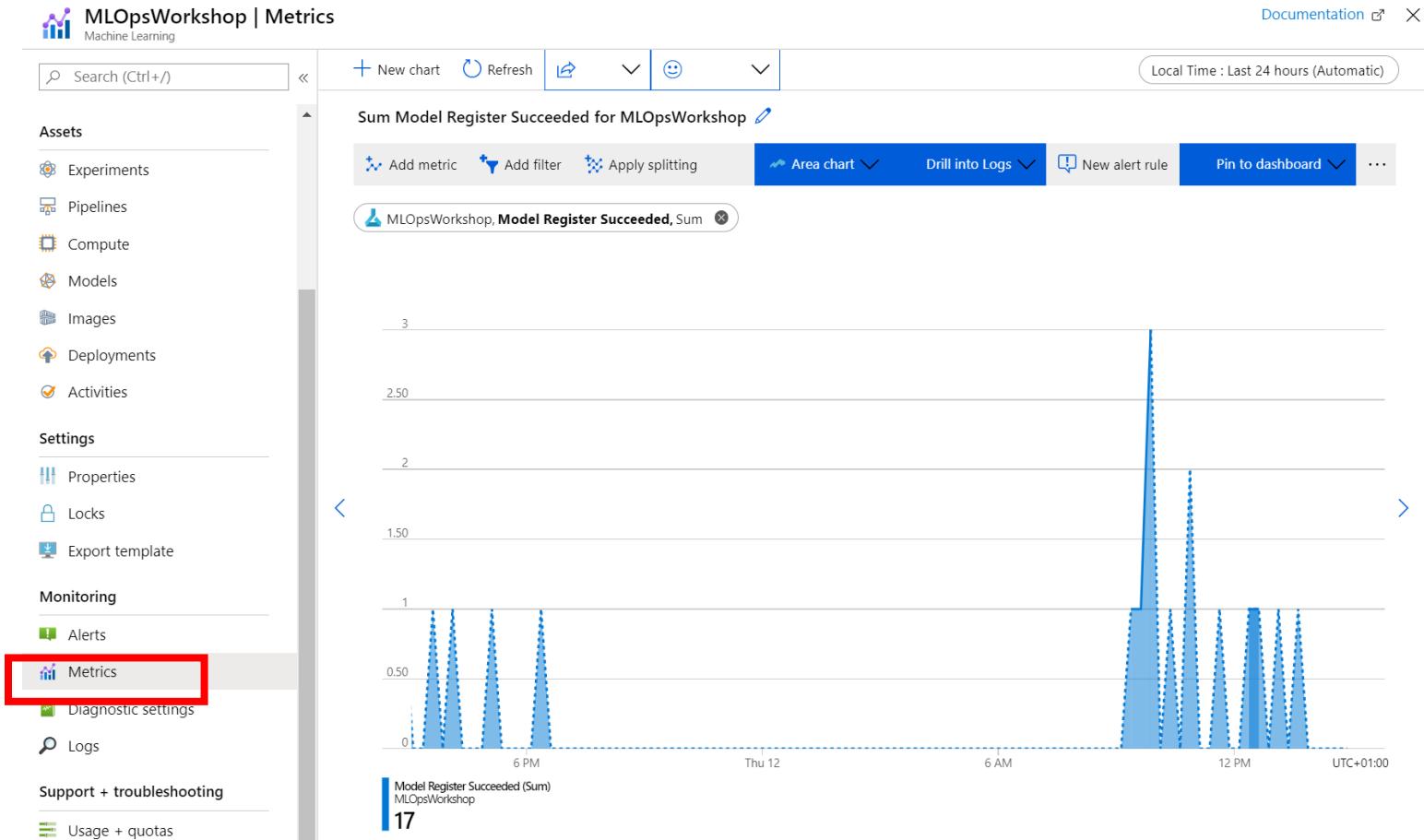
The screenshot shows the Microsoft Application Insights Logs blade for the resource group 'mlopsworkshop9122234384'. The left sidebar includes links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Investigate (Application map, Smart Detection, Live Metrics (Preview), Search, Availability, Failures, Performance, Troubleshooting guides), Monitoring (Alerts, Metrics, Logs, Workbooks), and Usage. The 'Logs' link is highlighted with a red box. The main area displays a query editor with the query 'traces | limit 50' and a results table showing log entries from the last 24 hours. The results table has columns: timestamp [UTC], message, severityLevel, itemType, and customDimensions. The first few rows of data are as follows:

timestamp [UTC]	message	severityLevel	itemType	customDimensions
12/03/2020 à 13:58:51.454	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:58:51.454"}
12/03/2020 à 13:59:08.784	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:59:08.784"}
12/03/2020 à 13:59:27.973	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:59:27.973"}
12/03/2020 à 13:59:38.909	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:59:38.909"}
12/03/2020 à 13:59:42.419	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:59:42.419"}
12/03/2020 à 13:59:51.799	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:59:51.799"}
12/03/2020 à 13:59:51.799	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:59:51.799"}
12/03/2020 à 13:59:53.509	model_data_collection	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Container Id": "chd-predictor", "Timestamp": "Mar 12 13:59:53.509"}
12/03/2020 à 13:59:53.510	STDOUT	trace	trace	{"Workspace Name": "MLOpsWorkshop", "Timestamp": "Mar 12 13:59:53.510"}

# Métriques Azure ML

The screenshot shows the Azure ML Metrics interface. On the left, there's a sidebar with a red box highlighting the 'Metrics' option under the 'Monitoring' section. The main area displays a chart titled 'MLOpsWorkshop | Metrics' with a scope of 'MLOpsWorkshop' and a metric namespace of 'Machine Learning S...'. The chart shows a single data series from 'Thu 12' to '12 PM UTC+01:00' with values ranging from 0 to 100. The chart title is 'Chart Title'. Below the chart, there are sections for 'MODEL' and 'QUOTA' metrics, each with several items listed. At the bottom, there are three cards: 'Filter + Split', 'Plot multiple metrics', and 'Build custom dashboards'. The top right corner shows the user's email (seretkow@microsoft.com) and Microsoft branding.

# Métriques Azure ML



# Documentation



# Documentation MLOps



MLOps

<https://azure.microsoft.com/fr-fr/services/machine-learning/mlops/>



Azure DevOps

<https://docs.microsoft.com/en-us/azure/devops/?view=azure-devops>



Blog

<https://devblogs.microsoft.com/devops/>



Pricing

<https://azure.microsoft.com/fr-fr/pricing/details/devops/azure-devops-services/>



Git MLOps

<https://github.com/microsoft/MLOps>

