



**Hewlett Packard**  
Enterprise

# MODULE 2: END-END MACHINE LEARNING

HPE-Official Not For Circulation

# AGENDA

---

DATA SCIENCE PIPELINES

MLOPS ECOSYSTEM

AUTOMATE COMPLEX DATA TRANSFORMATION HYDERM

SPARK PIPELINES

MLFLOW

MLOPS FRAMEWORK TO PACKAGE, DEPLOY, MONITOR AND MANAGE MODELS  
SELDON CORE

# LESSONS

---

## Pipeline

- What is Pipeline
- Training Pipeline
- Inference Pipeline

## Model Deployment

- Real-time Inferencing
- Microservices
- Event-Based
- Container
- Batch Inferencing

## Spark pipelines

- ML Model Lifecycle
- Training
- Hyperparameter Tuning

## Pachyderm



## MLflow

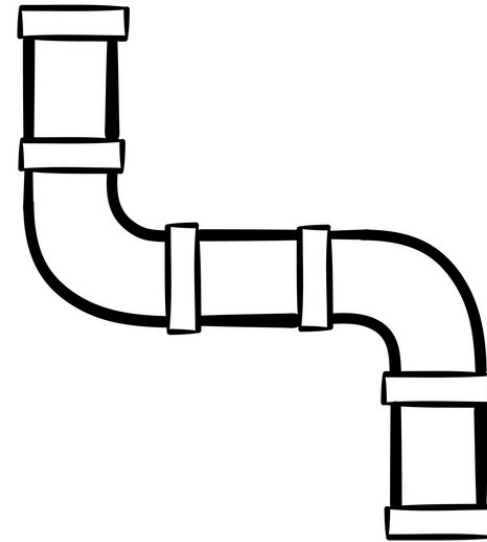


## Seldon



# LESSON 1: PIPELINE

---



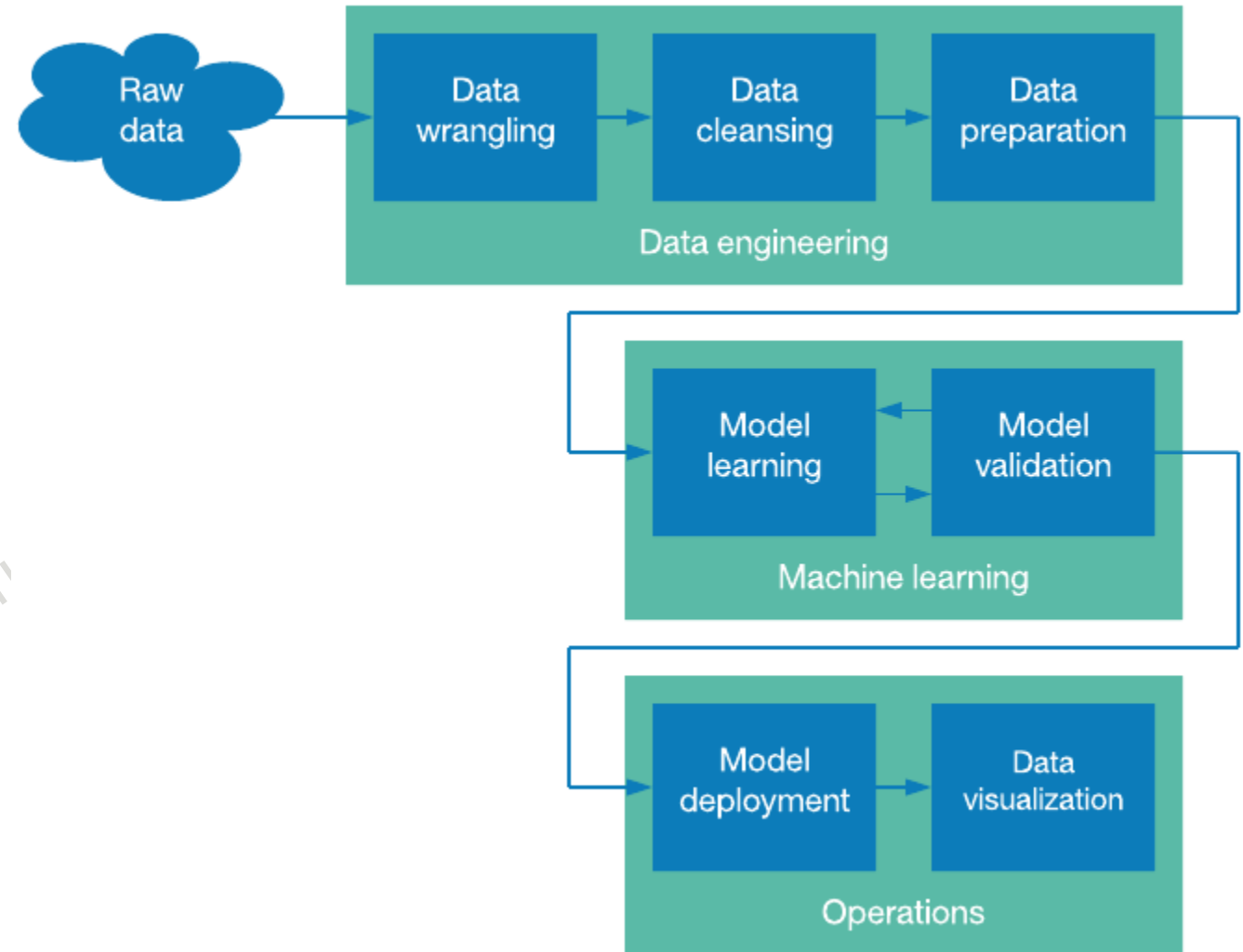
HPE-Official Not For Circulation



# DATA SCIENCE PIPELINES

A data science pipeline is a process collection that transforms raw data into useful solutions to business issues.

Pipelines for data science streamline data movement from source to destination, allowing you to make better business decisions.



# PIPELINE

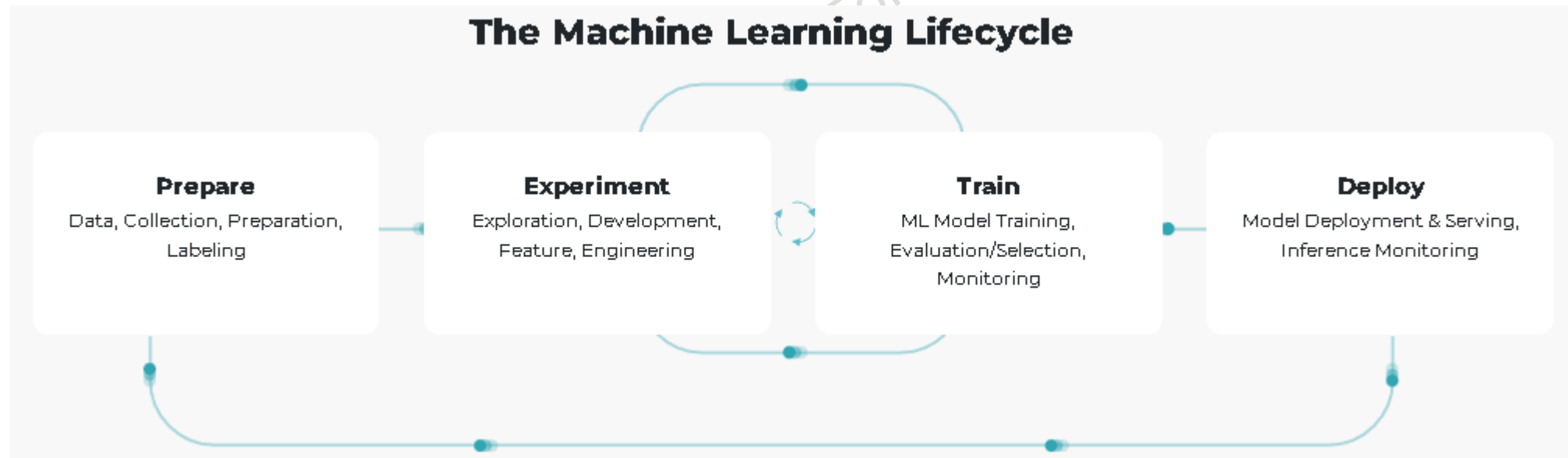
---

- For orchestrating the ML model lifecycle
- Series of steps or actions
- Each step consumes output of previous step and generate output provided to subsequent step
- Each step include actions related to data processing, model training, model testing



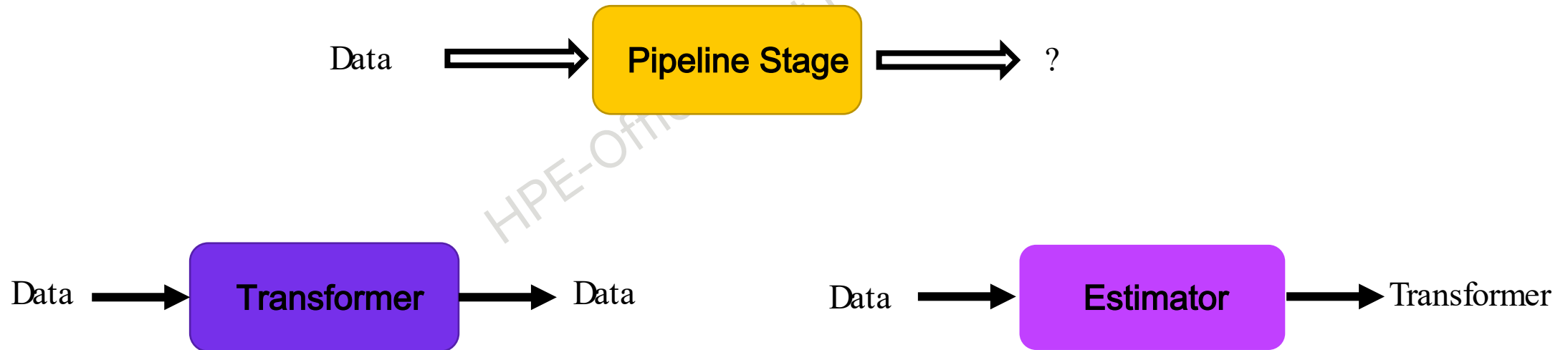
# MLOPS ECOSYSTEM

- When it comes to MLOps, no one tool can do it all.
- It takes an array of best-of-breed tools to truly automate the entire ML lifecycle.



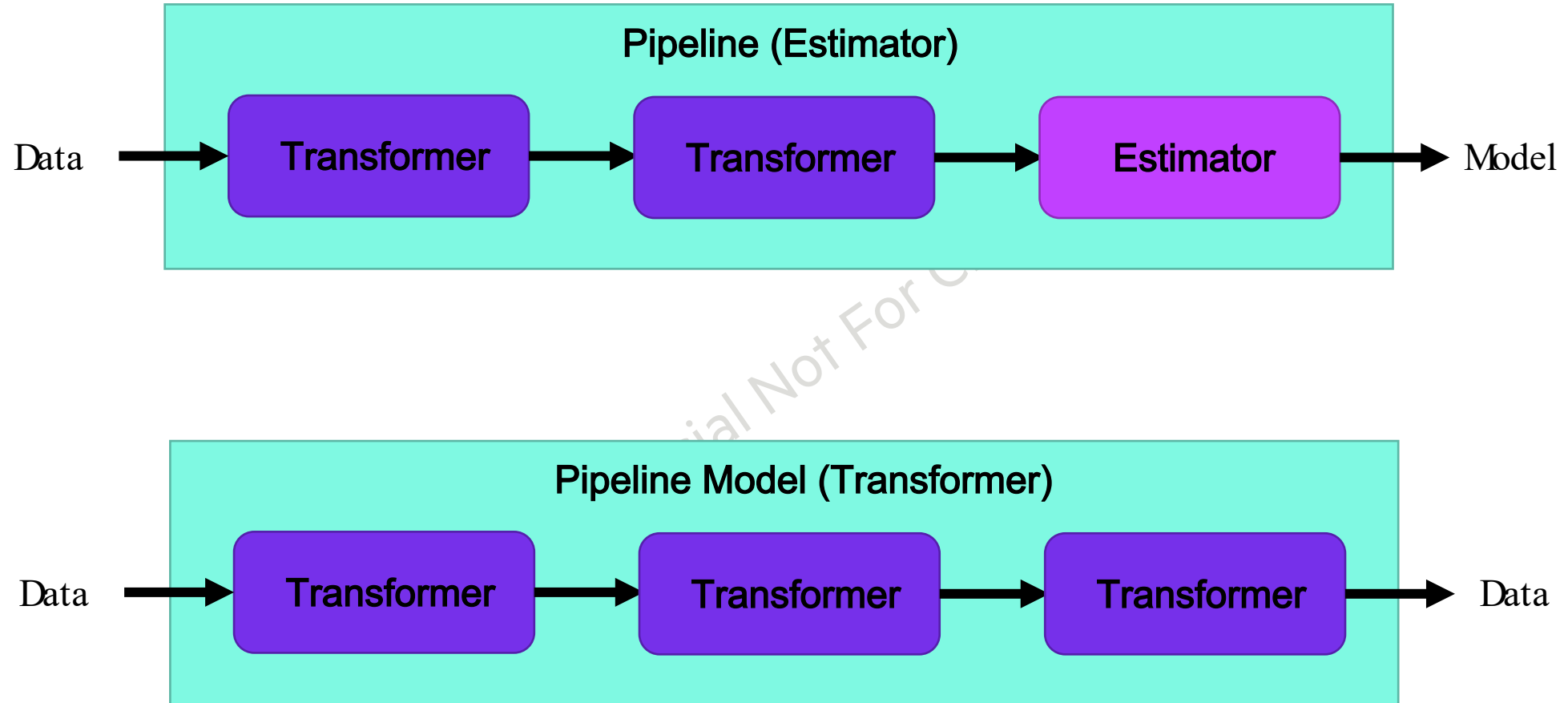
# SPARK PIPELINE

- A Spark pipeline comprises a series of Estimators or Transformers at each step.
- The Estimator uses an incoming data frame to produce a Transformer.
- The Transformer uses an incoming data frame to produce a new data frame.





# SPARK PIPELINE

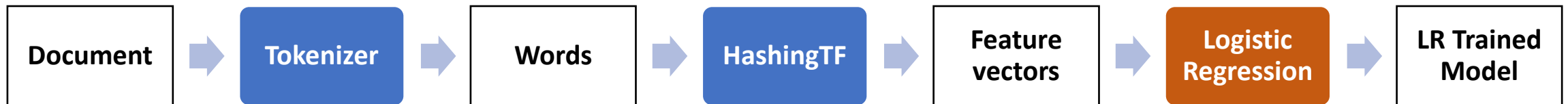


# TRAINING PIPELINE

- Process of creating a machine learning model.

Example: Processing a Text Document

- Step 1: Split the text in the document into words.
- Step 2: Generate a feature vector for each of the unique words in the document.
- Step 3: Train a prediction model using the feature vectors and the corresponding labels.



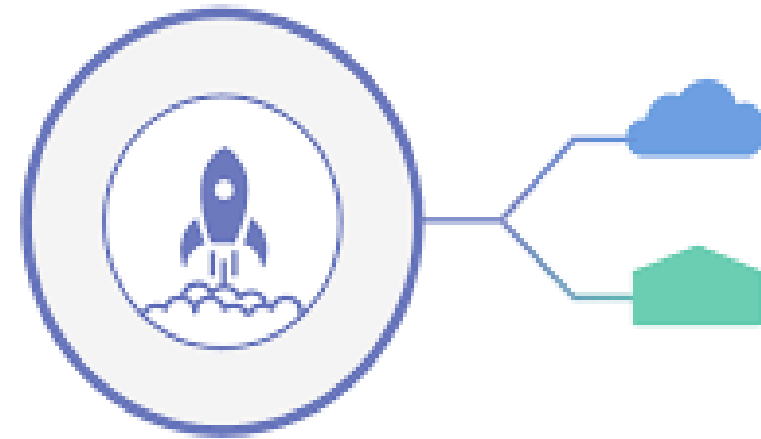
# INFERENCE PIPELINE

- Process of utilizing the trained machine learning model to make a prediction for the live data.
- Inference pipeline comprises only Transformers.

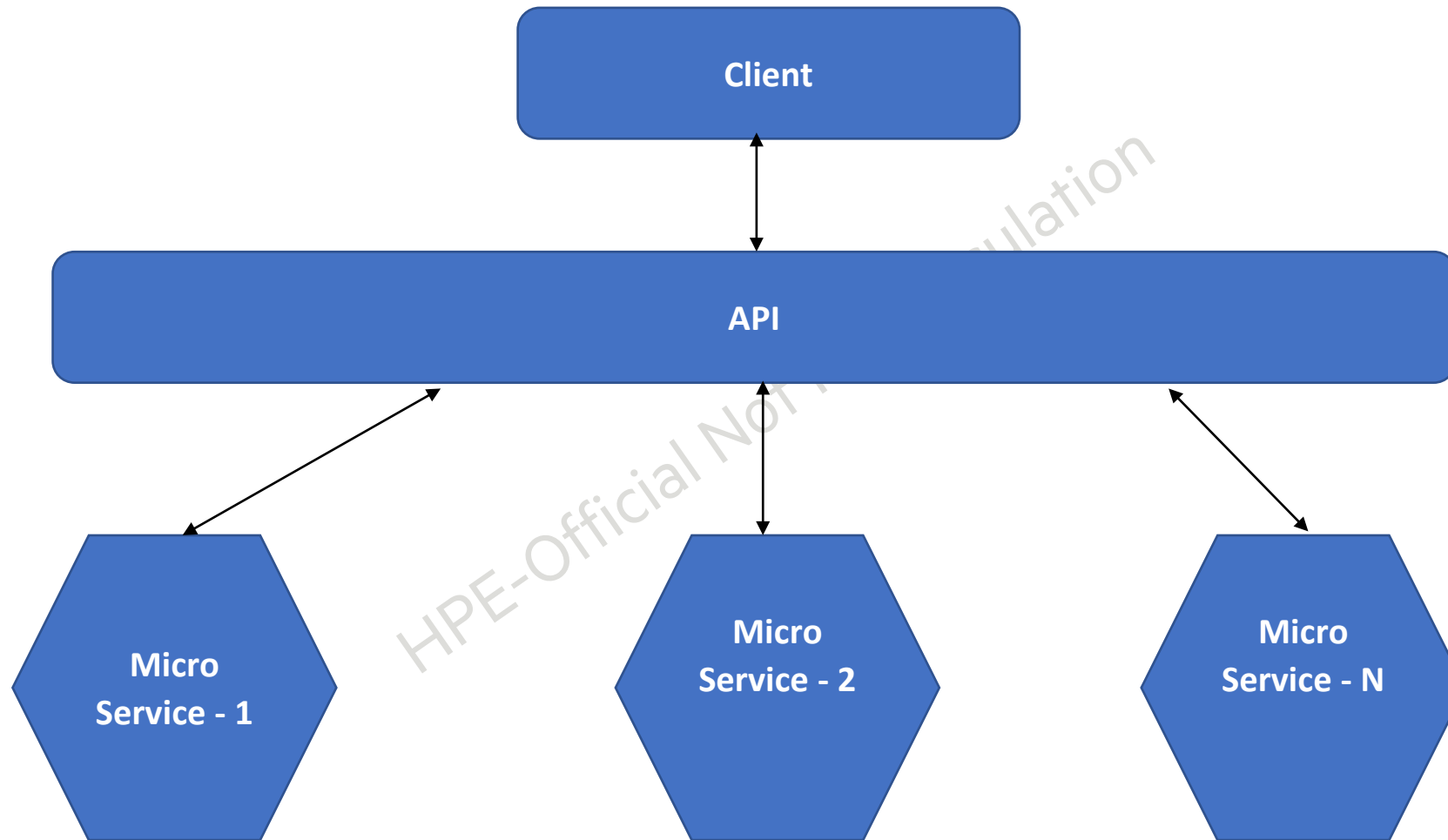


## LESSON 2: MODEL DEPLOYMENT

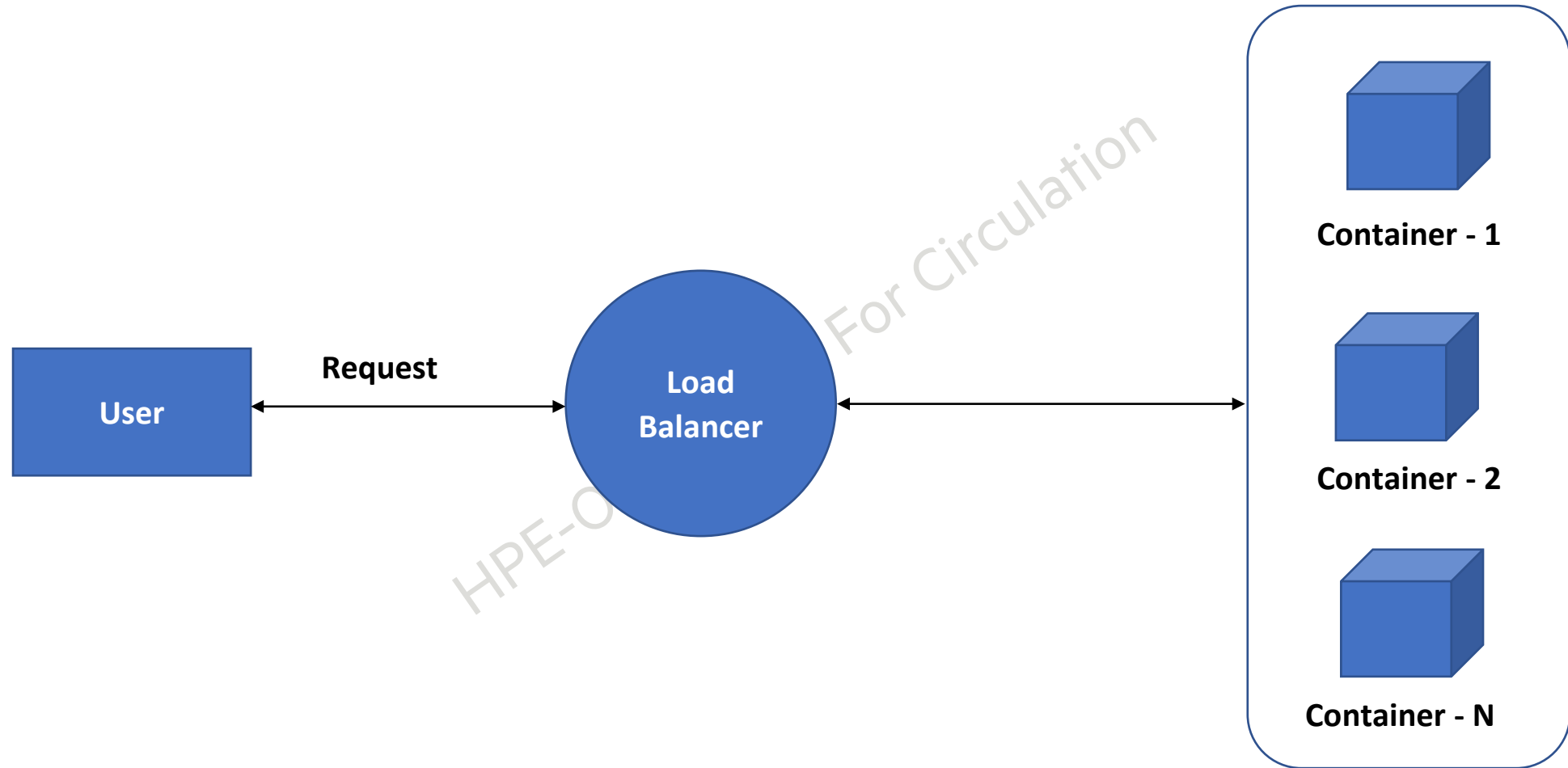
---



# MICROSERVICES



# CONTAINERIZATION



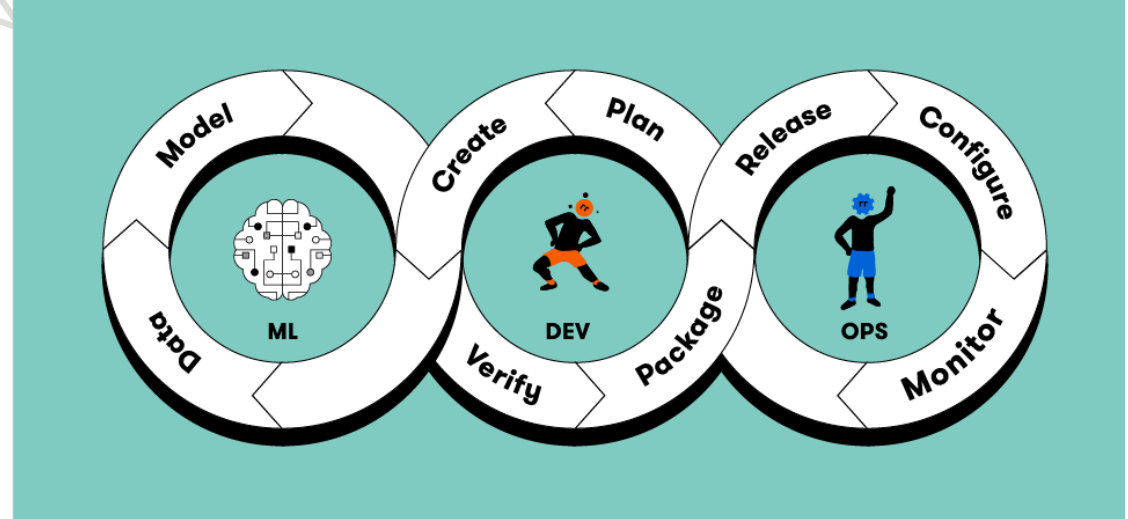
# ONNX MODEL



# LESSON 3: MLOPS

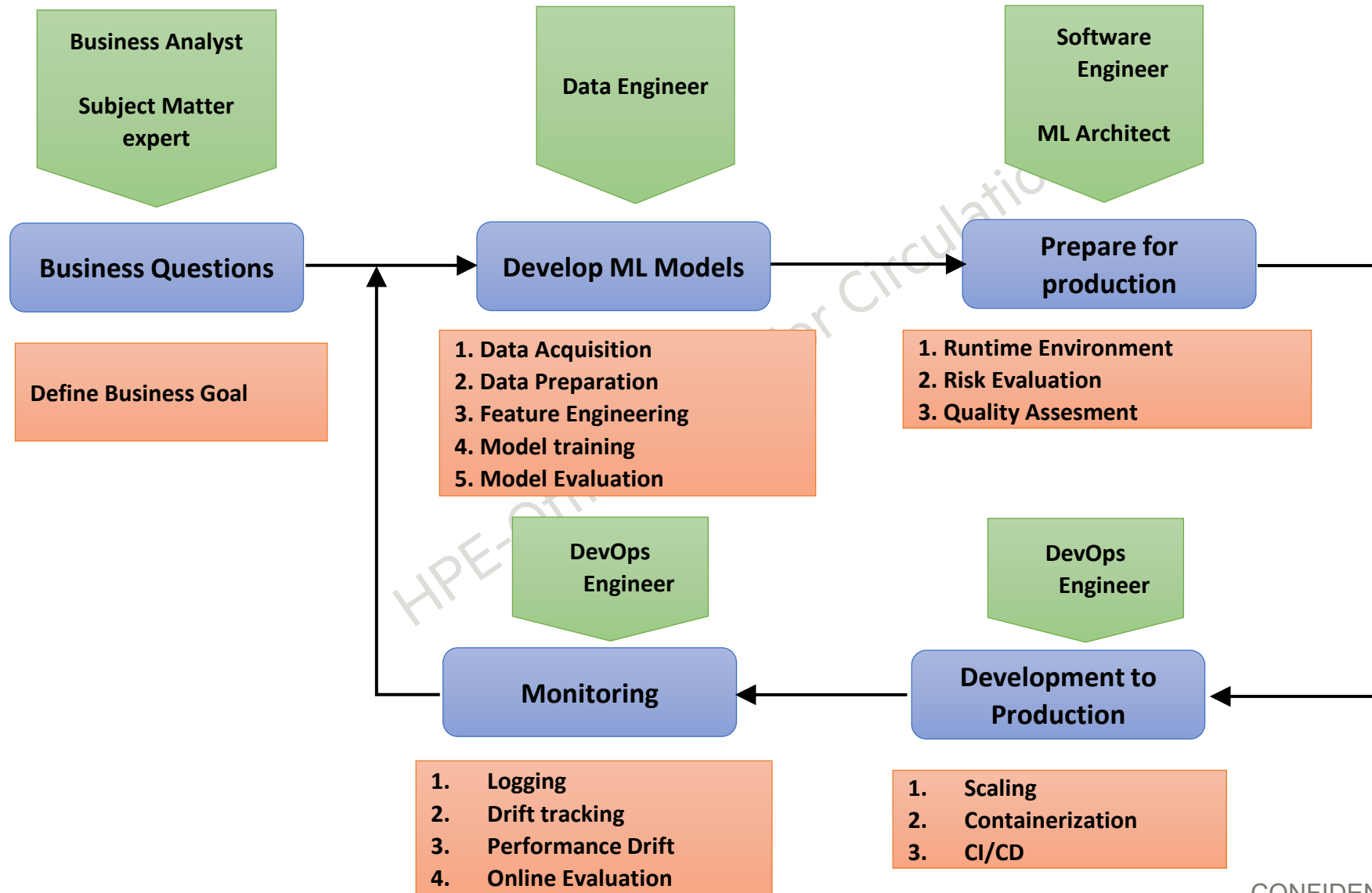
---

HPE-Official Not For Circulation

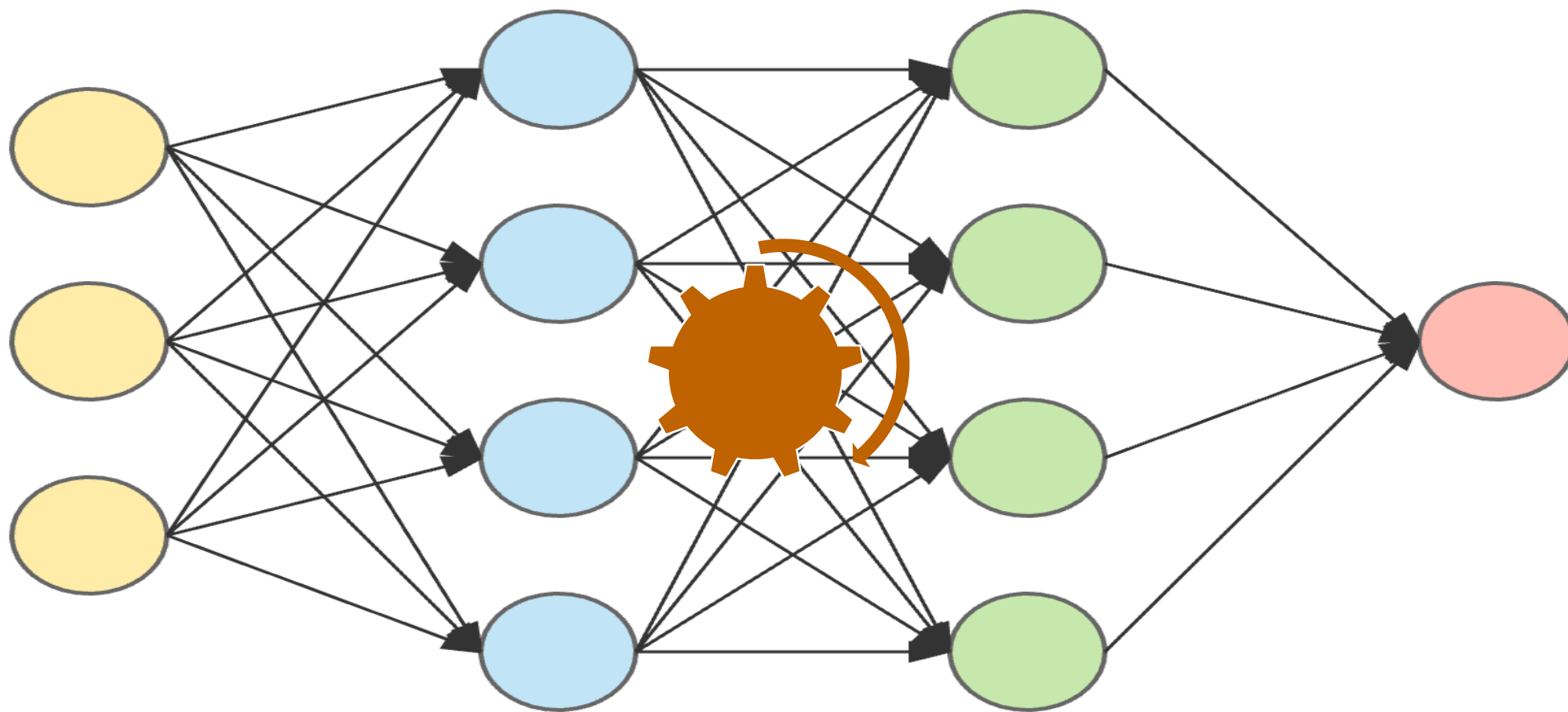




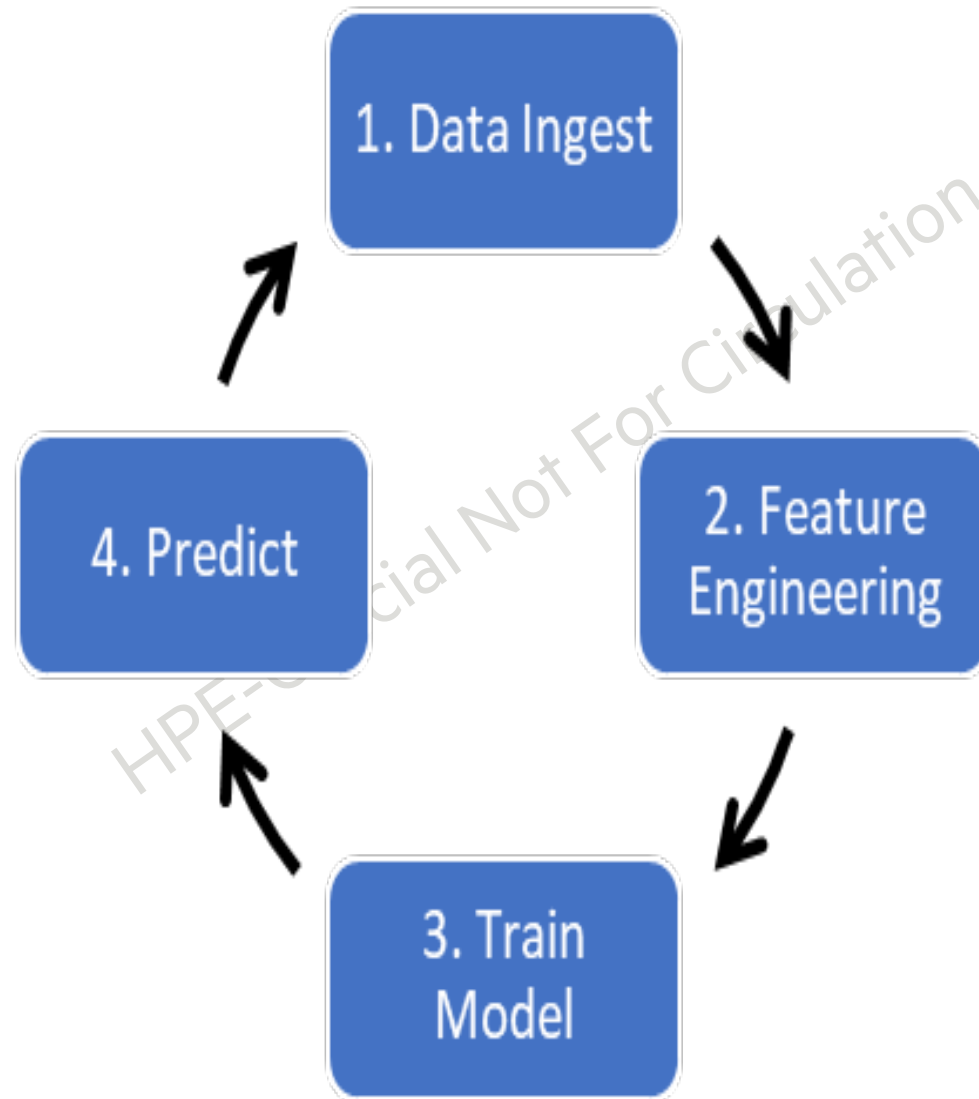
# LIFE CYCLE OF A MACHINE LEARNING MODEL



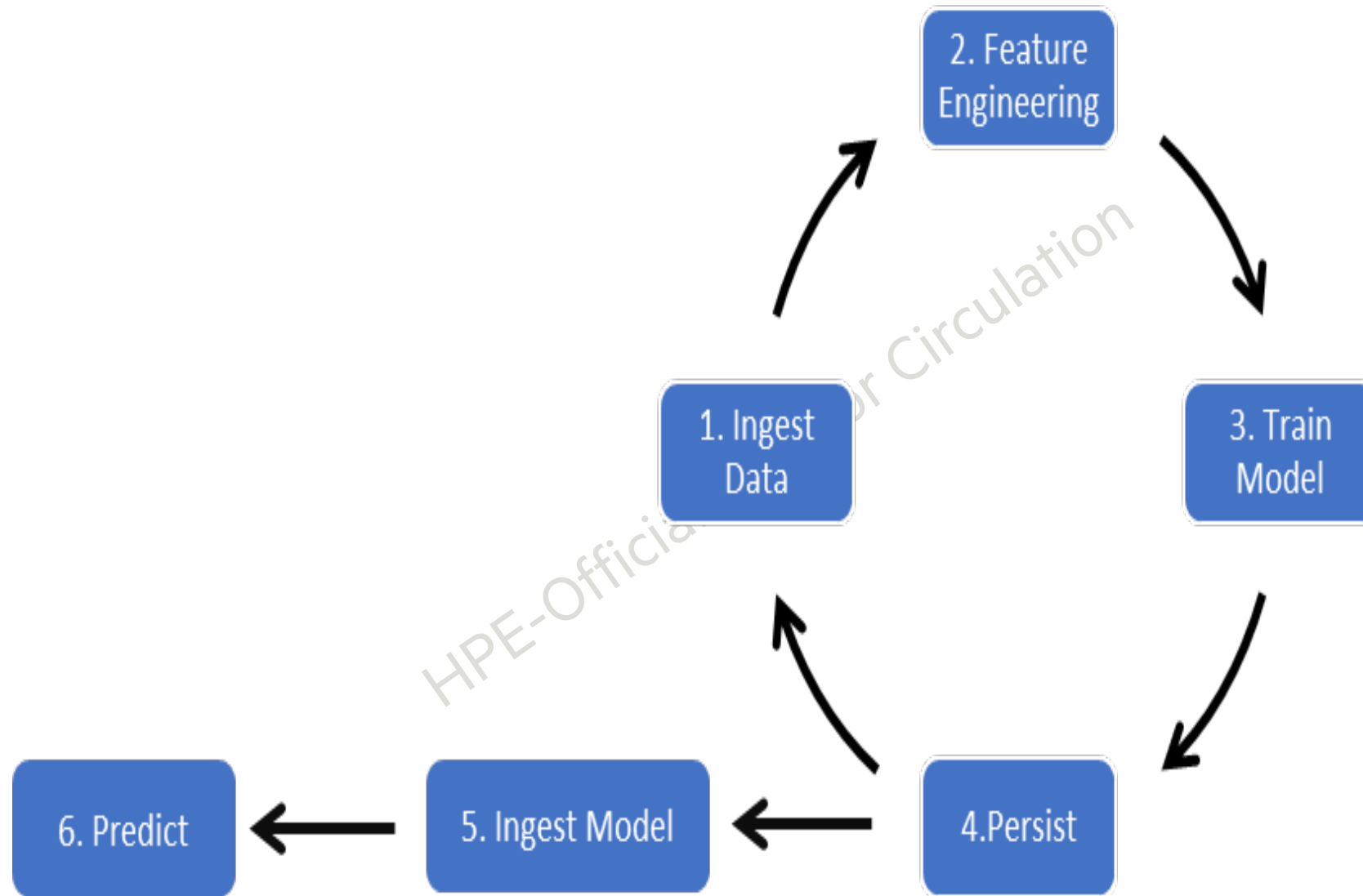
# TRAINING



# MODEL TRAINING

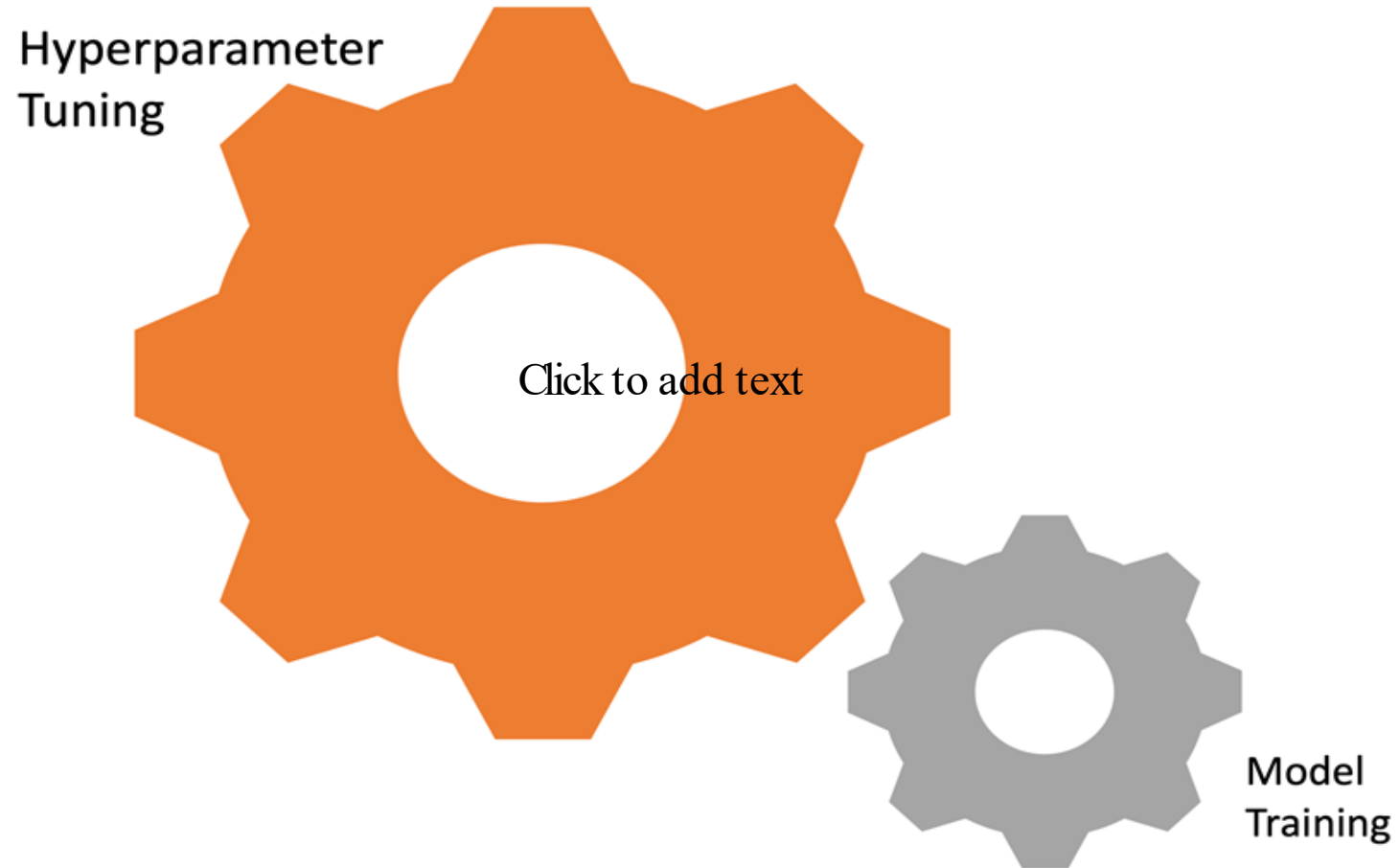


# MODEL TRAINING



# HYPERPARAMETER TUNING

---



# HYPERPARAMETER TUNING

---

- Process of finding the best set of parameters for the Estimator such as ML model.
- General process of hyperparameter tuning works as follows,
  - The input data is split into one/more training and testing sets.
  - For each <Train, Test> pair of sets, the Estimator is iterated over the set of parameters.
  - The performance of the model is evaluated on the test set.
  - The best performing model is selected based on the metric provided in the Evaluator.



# HYPERPARAMETER TUNING

---

- **Data Split:**

- Cross-Validation
- TrainValidationSplit

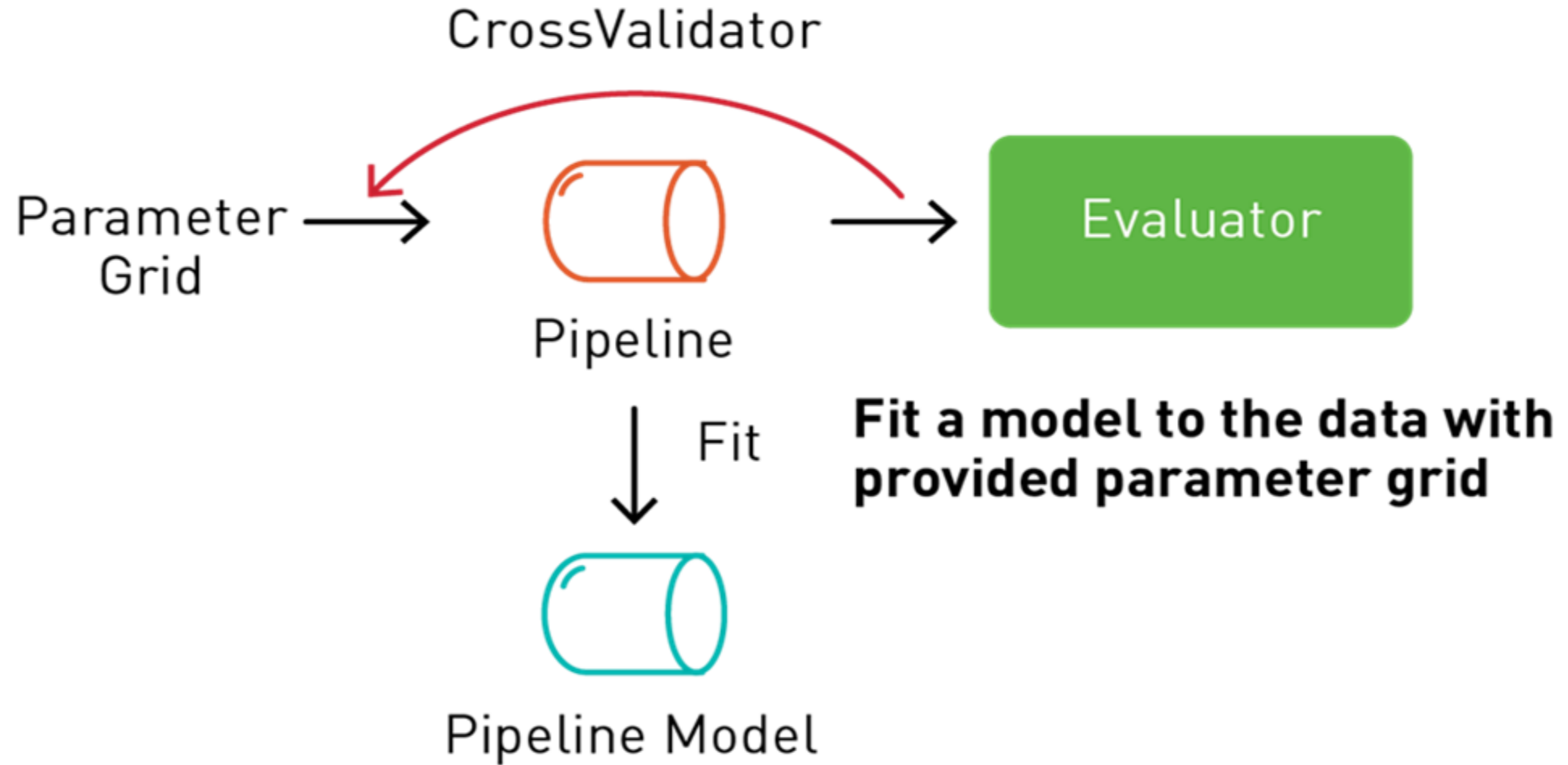
- **Iterations:**

- Parameter Grid

HPE-Official Not For Circulation

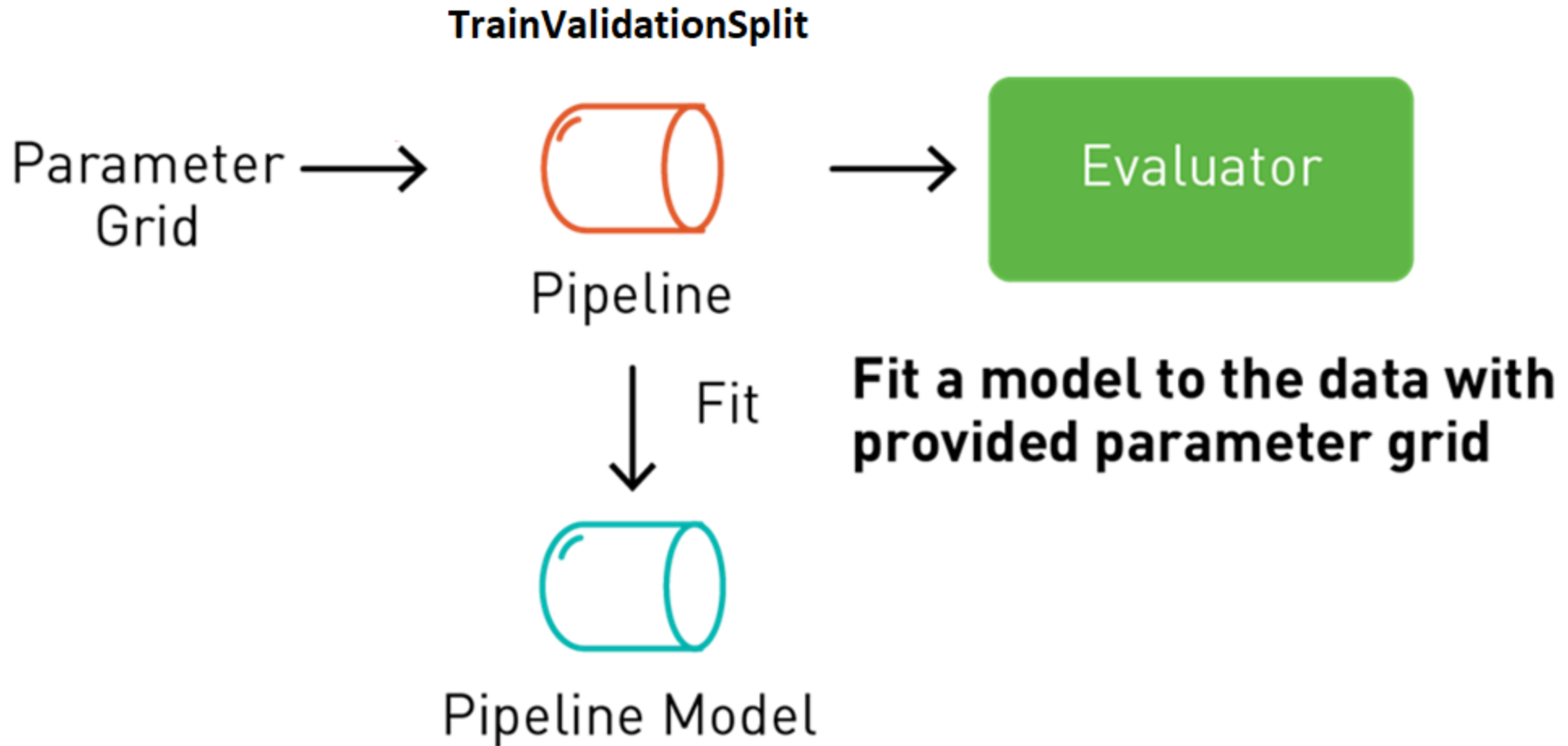


# CROSS VALIDATION





# TRAIN VALIDATION SPLIT



# HYPERPARAMETER TUNING

---

- **Evaluator:**

RegressionEvaluator

BinaryClassificationEvaluator

MulticlassClassificationEvaluator

MultilabelClassificationEvaluator

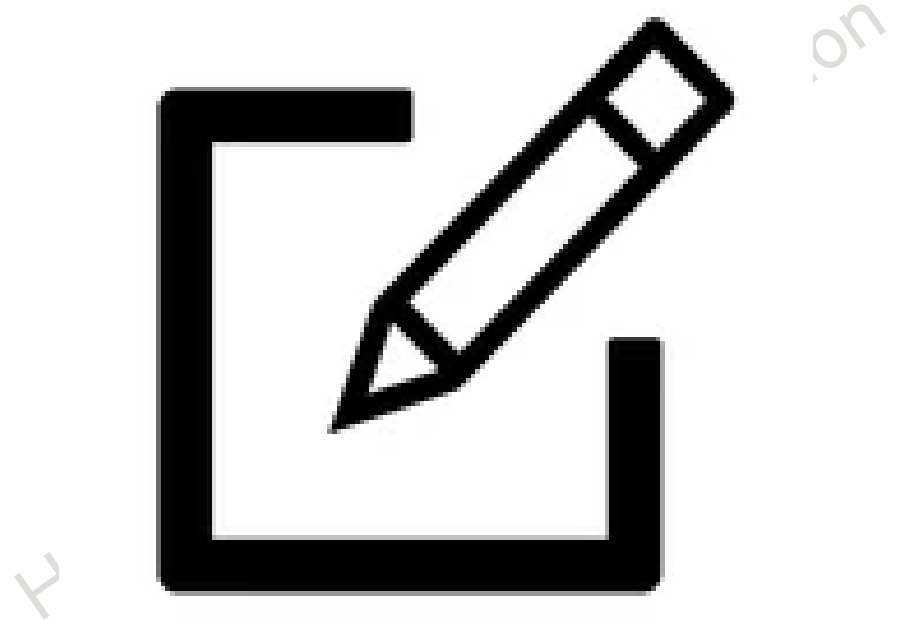
RankingEvaluator

HPE-Official Not For Circulation



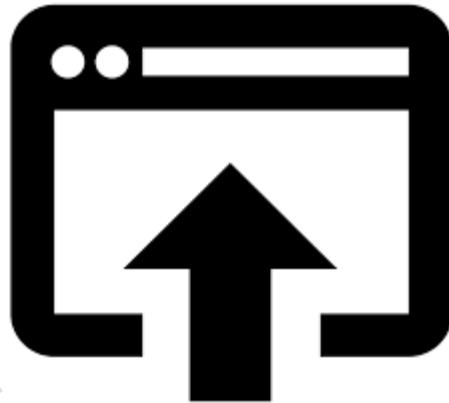
# REGISTER AN ML MODEL

---



# PUBLISHING AN ML MODEL

---



# WEB SERVICE

- Deploying the ML model as a HTTP endpoint:
  1. Set up a web service using Flask, Gunicorn, and Dash.
  2. Load the pre-trained ML models.
  3. Create the feature vector required to be provided as input to the ML model.
  4. Obtain and append the model output under the “response” key.



Server

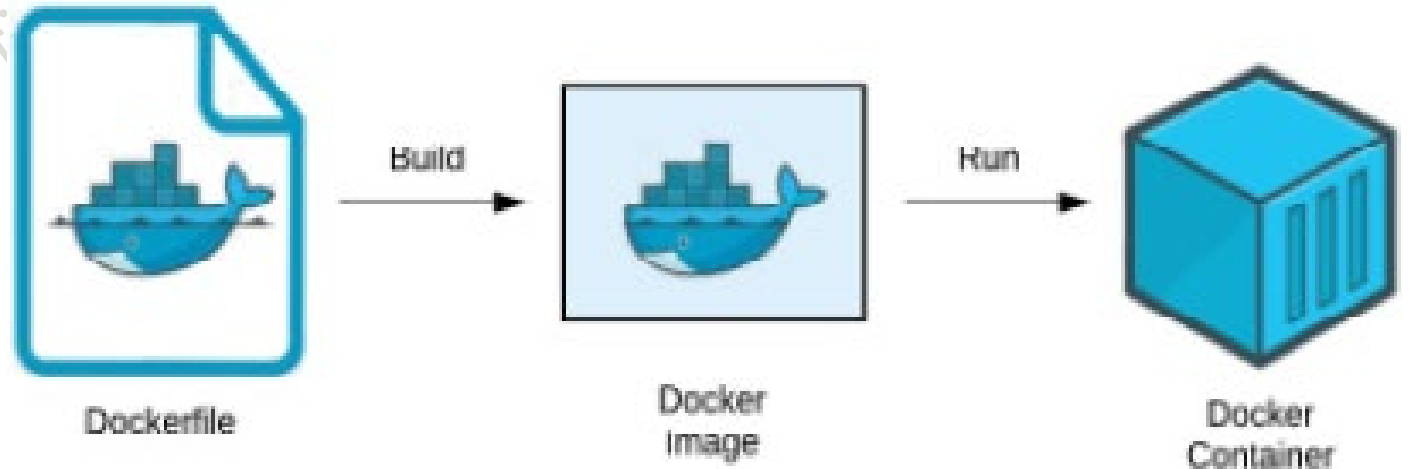
Server hosting  
the web service

Response from  
server to client

Response from  
client to server

# CONTAINER DEPLOYMENT

1. Create a requirements.txt file with the requires dependencies.
2. Create a Dockerfile to install the dependencies from the requirements file, mention the directory for the web application.
3. Place the web application files in the working directory along with the trained model files.
4. Build the container.
5. Verify that the image is created
6. Run the container.



# CHALLENGES IN MODEL SERVING

---











- Serving infrastructure
- Scaling up (and down!)
- Monitoring models
- Updating models

HPE-Official Not For Circulation



# AIML CANONICAL STACK

When it comes to MLOps, no one tool can do it all.

		IT Infrastructure	Data Labelling	Synthetic Data	Data Versioning & Management	Exploratory Data Analysis (EDA)	Feature Stores	Code Management	Model Development	Distributed Training	Hyperparameter Tuning	Experiment Tracking & Metadata Store	Model Repository	Model Inference	Model Deployment	Model Testing / Validation	Monitoring / Observability	Interpretation / Explainability	Team Management	Orchestration	Machine Learning Security	Dashboards	AutoML
 CLEAR ML		ITI	DL		DVM	EDA	FS	CM	MDV	DT	HT	ETMS	MR	MI	MDP	MTV	MO	IE	TM	O	MLS	D	AML
 Determined AI		ITI							DT	HT	ETMS											D	
 mlflow											ETMS	MR	MI										
 Pachyderm					DVM	EDA		CM	MDV	DT	HT	ETMS	MR		MDP	MTV			TM	O		D	
 SELDON												ETMS	MR	MI	MDP	MTV	MO	IE	TM	O	MLS	D	



# DATA DRIVEN PIPELINE

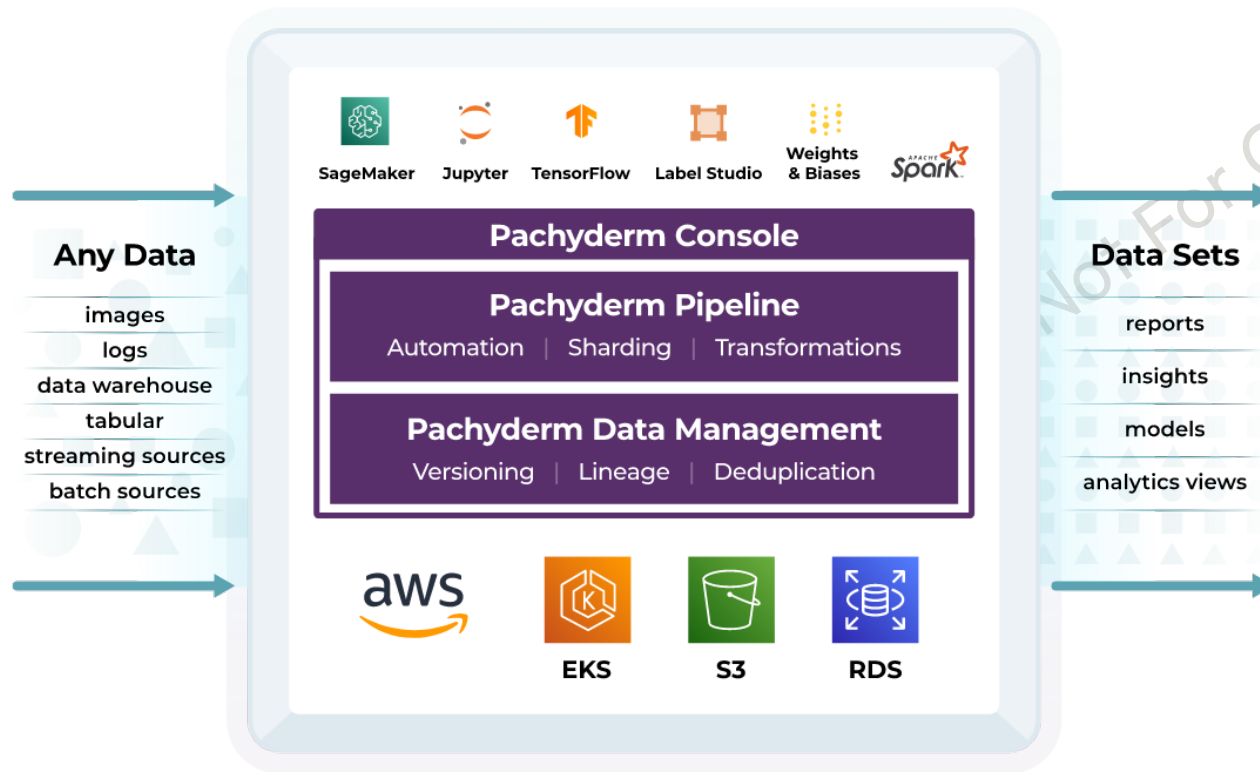
---

HPE-Official Not For Circulation



# PACHYDERM

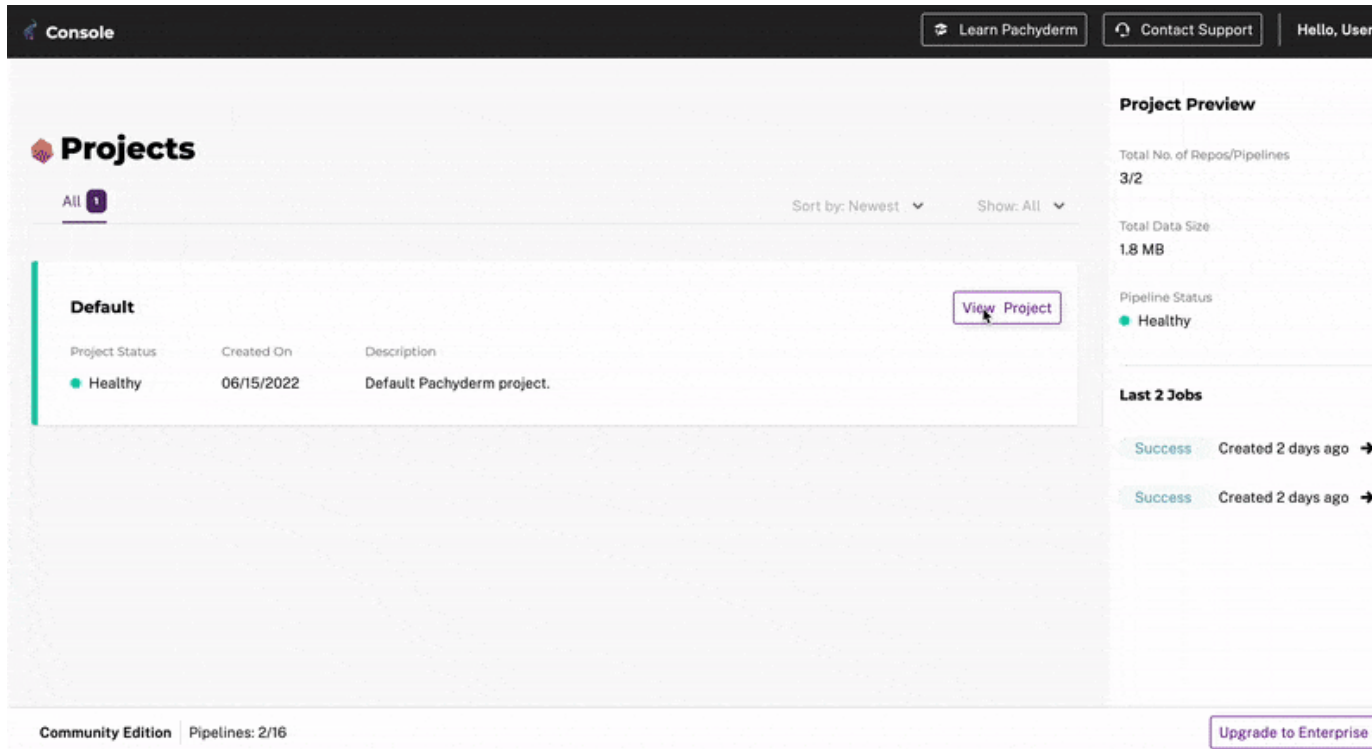
Pachyderm is cost-effective at scale, enabling data engineering teams to automate complex pipelines with sophisticated data transformations



- Data-driven pipelines are automatically triggered based on detecting data changes.
- Immutable data lineage with data versioning of any data type.
- Autoscaling and parallel processing built on Kubernetes for resource orchestration.
- Uses standard object stores for data storage with automatic deduplication.
- Runs across all major cloud providers and on-premises installations.

# PACHYDERM CONSOLE

Pachyderm Console is a complete web UI for visualizing running pipelines and exploring your data.



users can check their

- jobs' status,
- visualize their commits' content,
- access logs,
- and much more!

**It is a valuable companion when troubleshooting pipelines.**

# PACHYDERMPIPELINE

A pipeline is a Pachyderm primitive.

A minimum pipeline specification must include the following parameters:

name — The name of your data pipeline.

input — A location of the data that you want to process, such as a Pachyderm repository.

transform — Specifies the code that you want to run against your data.

Pipeline is responsible for

- reading data from a specified source, such as a Pachyderm repo,
- transforming it according to the pipeline configuration, and
- writing the result to an output repo.

HPE-Official Not For Circulation



# PACHYDERM DATA MANAGEMENT

## Version Control in Pachyderm

- Pachyderm handles version control by breaking down the jobs for your machine learning model at the datum level.
- This iterative approach to look what changed - data or code
- Pachyderm's data version control system allows for automated file tracking and complete audits

## Data lineage

it's the history of your data.

It tells us

- where that data comes from,
- where it lives and
- how it's transformed over time.

Why track lineage?

Changing data changes your experiments.

If your data changes after you've run an experiment, you can't reproduce that experiment.

Reproducibility is utterly essential in every data science project.



# SELDON

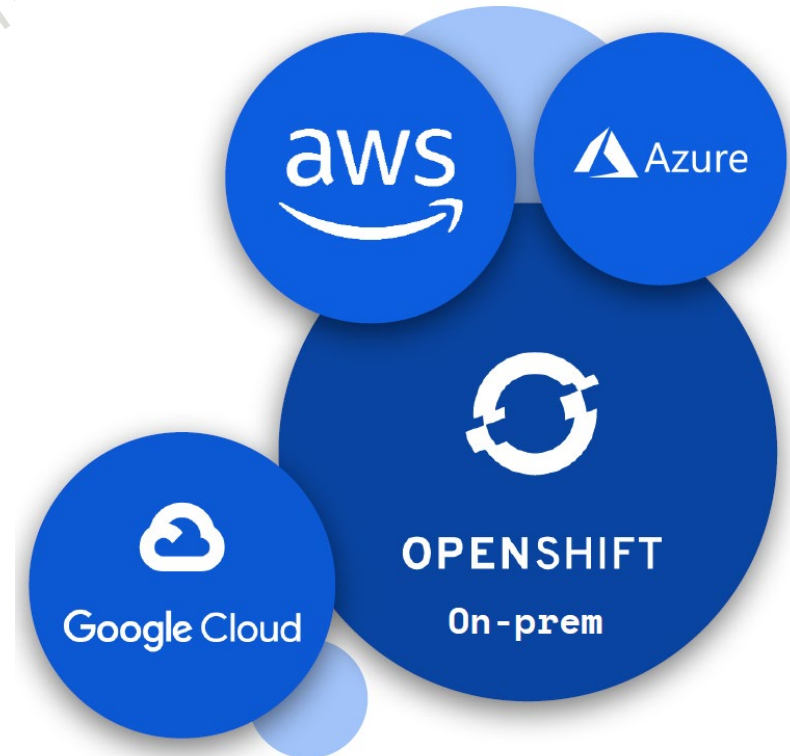
---

HPE-Official Not For Circulation



# SELDON CORE

- Open-Source project created by Seldon
- An MLOps framework to package, deploy, monitor and manage thousands of production machine learning models
- Built on top of Kubernetes Cloud Native APIs.
- All major cloud providers are supported.
- On-prem providers such as OpenShift are also supported.



# SELDON

Seldon provides a set of tools for

- Deploying machine learning models at scale.
- Deploy machine learning models in the cloud or on-premise.
- Get metrics and ensure proper governance and compliance for your running machine learning models

**Seldon Core** An MLOps framework to package, deploy, monitor and manage thousands of production machine learning models

**Alibi Explain** Algorithms for AI Explainability for machine learning models

**Alibi Detect** Algorithms for outlier detection, concept drift and metrics

**Tempo** MLOps SDK for accelerating data science experimentation with Seldon Core and KFserving

**MLServer** High performance async machine learning model server for Seldon and KFserving

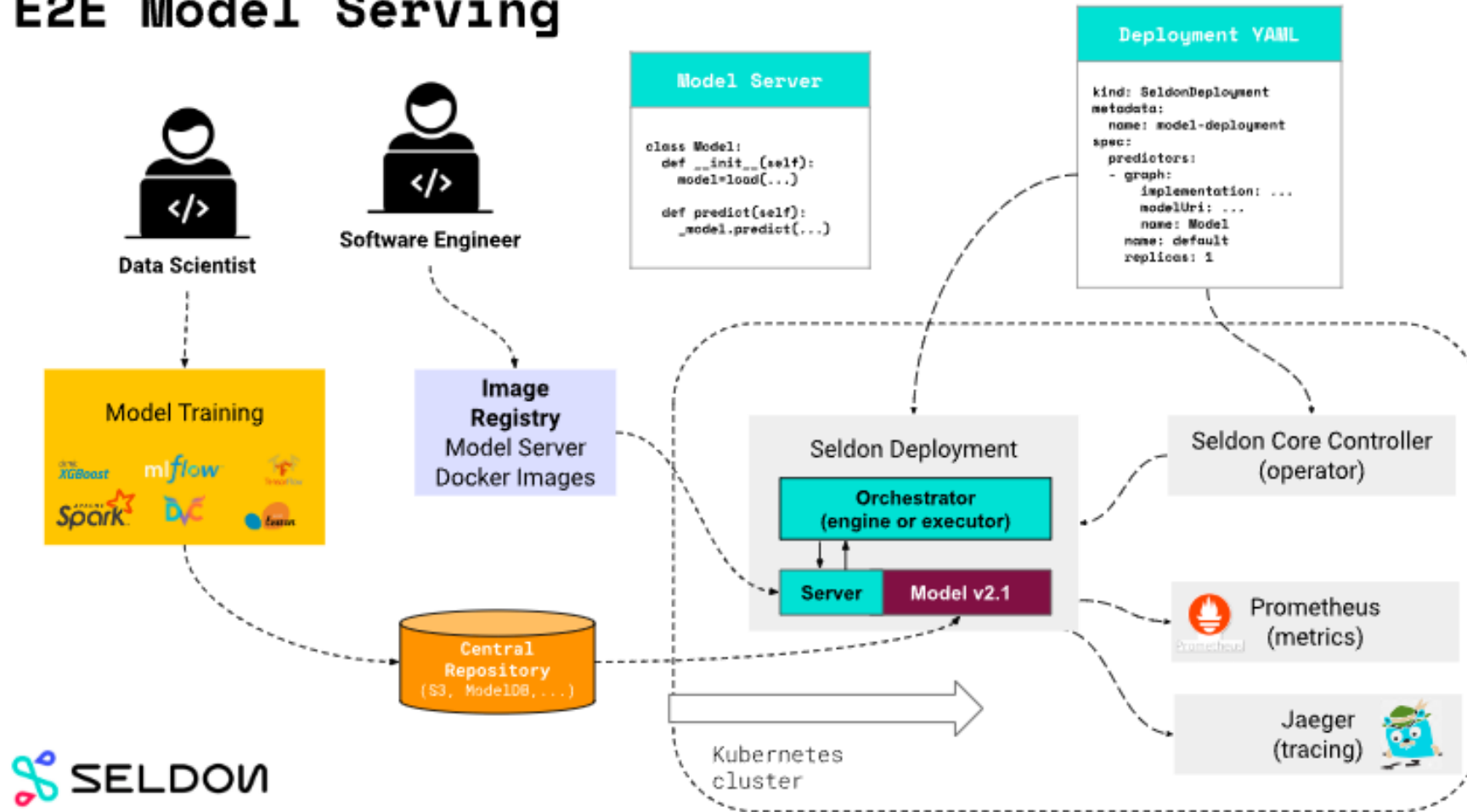
**Seldon Deploy** Enterprise platform that introduces advanced management, monitoring and explainability to open source stack

**Seldon Deploy SDK** Enterprise SDK that provides a Pythonic way to interact with the Seldon Deploy Enterprise API



# E2E MODEL SERVING

## E2E Model Serving



# MLFLOW

---

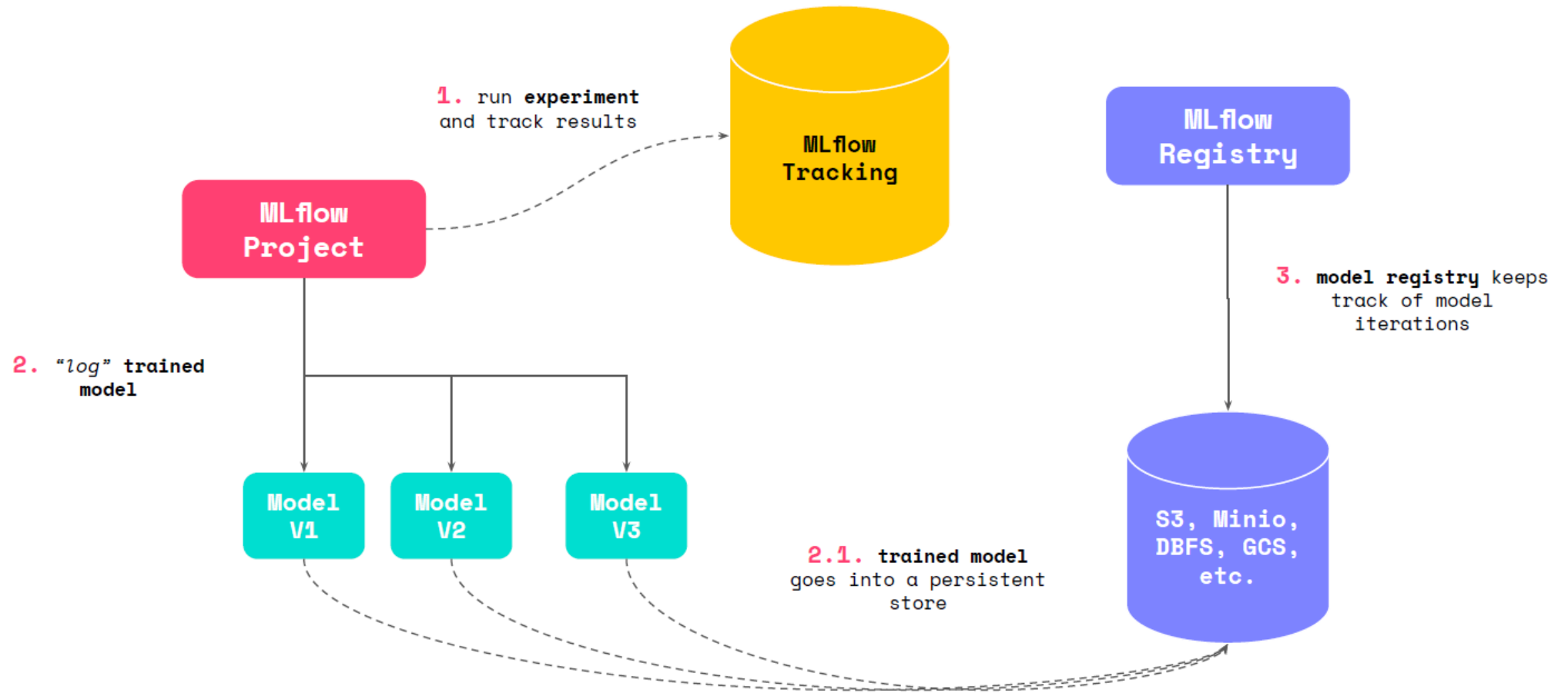
HPE-Official Not For Circulation

mlflow™



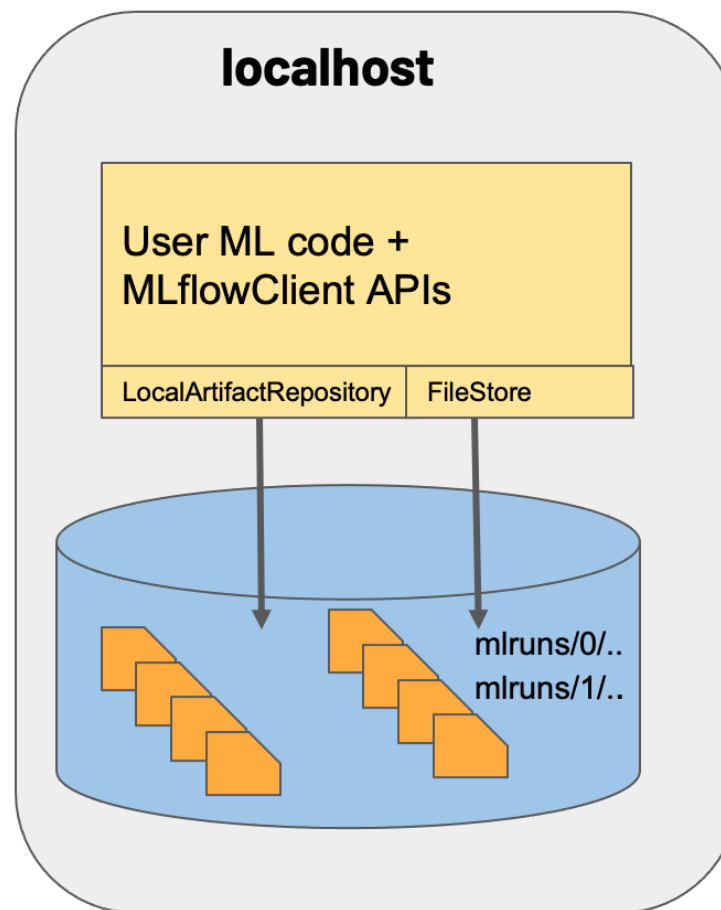
# MLFLOW

What is MLflow?

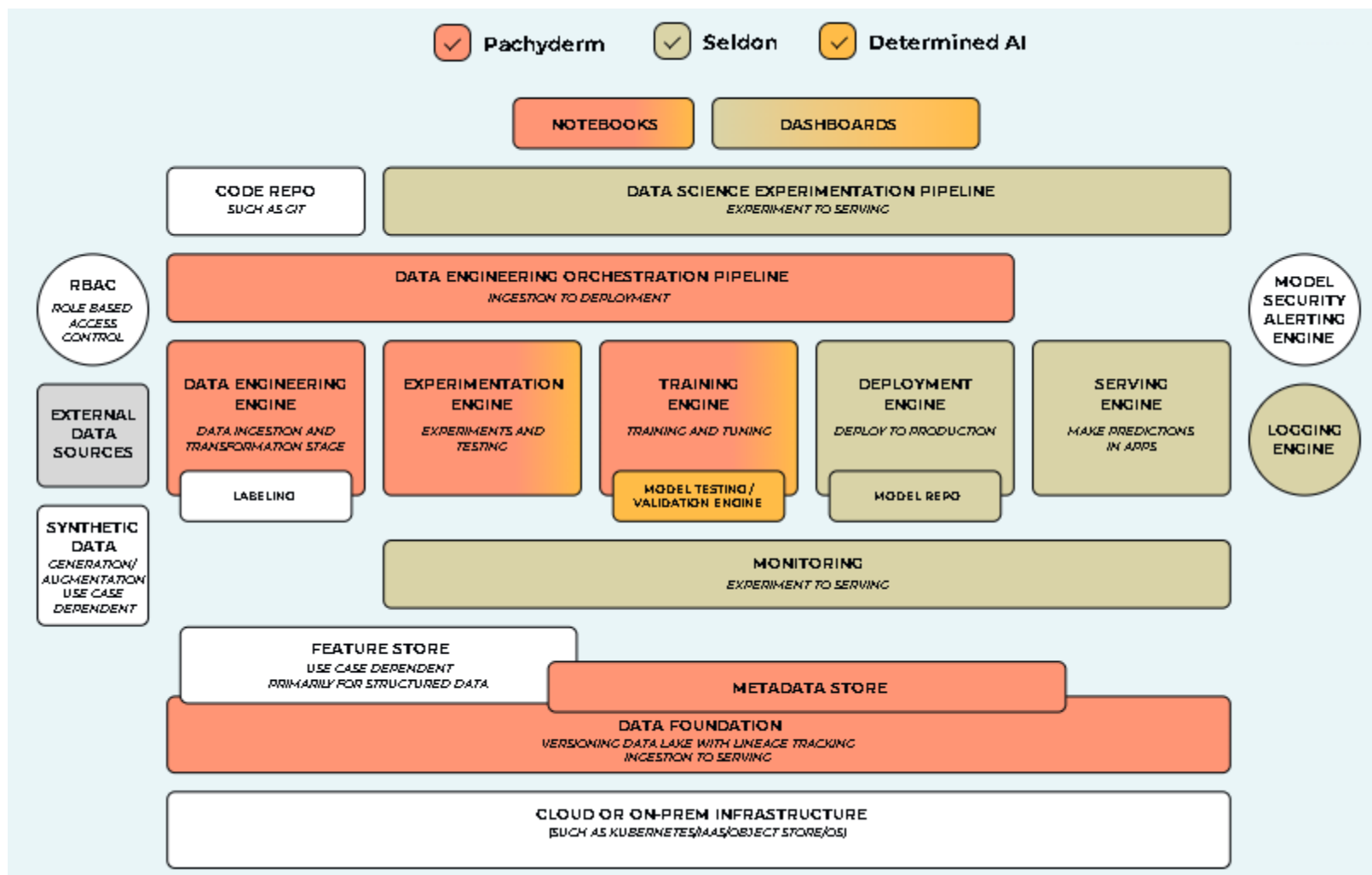


# MLFLOW

- A registered ML model has a unique name, contains versions, associated transitional stages, model lineage, and other metadata.
- **Registering**
  - `mlflow.spark.log_model`
  - `mlflow.register_model`
  - `create_registered_model`
- **Fetching**
  - `mlflow.spark.load_model`
- **Deleting**
  - `delete_registered_model`
  - `delete_model_version`



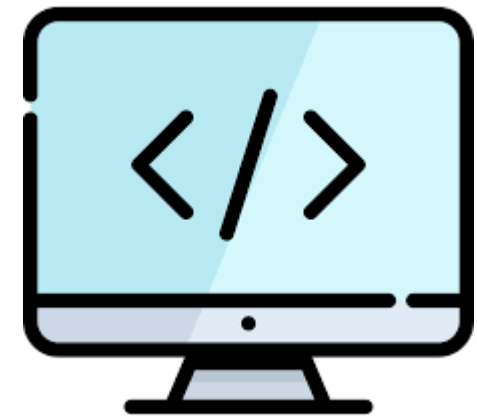
# INTEGRATION OF PACHYDERM+ DETERMINED AI+ SELDON



# DEMO

---

- Github link



# DATA DRIFT DETECTOR

---



HPE-Official Not For Circulation



# DATA DRIFT

---

- Historical dataset you used for model training reflects the new data your model will receive for inferencing.
- Patterns may emerge over time that alters the data profile, making your model less accurate.
- Consider a trained model to forecast an automobile's mileage based on the number of cylinders, engine size, weight, and other factors.
- The usual fuel economy of automobiles may increase considerably over time as car manufacturing and engine technology evolve.
- Forecasts generated by the model based on older data less accurate.

HPE-Official Not For Circulation





# TYPES OF MACHINE LEARNING ALGORITHMS

- Data drift is defined as the difference in data profiles between training and inferencing, and it may be a big problem for predictive models in production.
- To sustain forecast accuracy, it's critical to be able to track data drift over time and retrain models as needed.
- Need two datasets to monitor data drift using registered datasets:

1. **A reference dataset** The original training data is generally the case.

2. **The test datasets** Typically, the dataset generated after deployment.



# ALIBI DETECT BY SELDON.IO FOR DATA DRIFT MONITOR

- Outlier, adversarial, and drift detection are all covered by Alibi Detect, an open-source Python package.
- For drift detection, both TensorFlow and PyTorch backends are supported.

Detector	Tabular	Image	Time Series	Text	Categorical Features	Online	Feature Level
Kolmogorov-Smirnov	✓	✓		✓	✓		✓
Cramér-von Mises	✓	✓				✓	✓
Fisher's Exact Test	✓				✓	✓	✓
Least-Squares Density Difference	✓	✓		✓	✓	✓	
Maximum Mean Discrepancy (MMD)	✓	✓		✓	✓	✓	
Learned Kernel MMD	✓	✓	✓	✓	✓		
Context-aware MMD	✓	✓	✓	✓	✓		
Chi-Squared	✓				✓		✓
Mixed-type tabular	✓				✓		✓
Classifier	✓	✓	✓	✓	✓		
Spot-the-diff	✓	✓	✓	✓	✓		✓
Classifier Uncertainty	✓	✓	✓	✓	✓		
Regressor Uncertainty	✓	✓	✓	✓	✓		



# THANK YOU

---

HPE-Official Not For Circulation

