

ORACLE

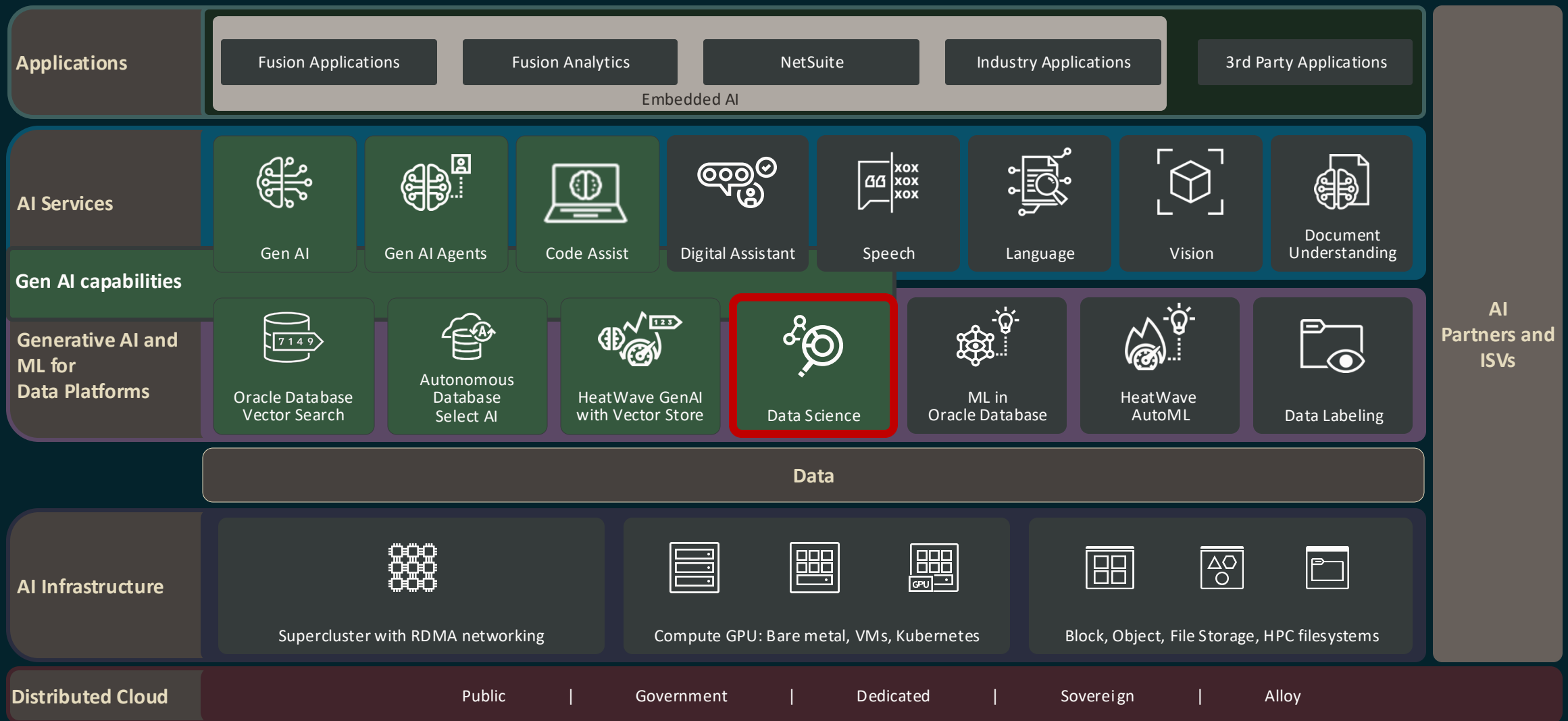
# OCI Data Science

**Oracle Data Science in action**

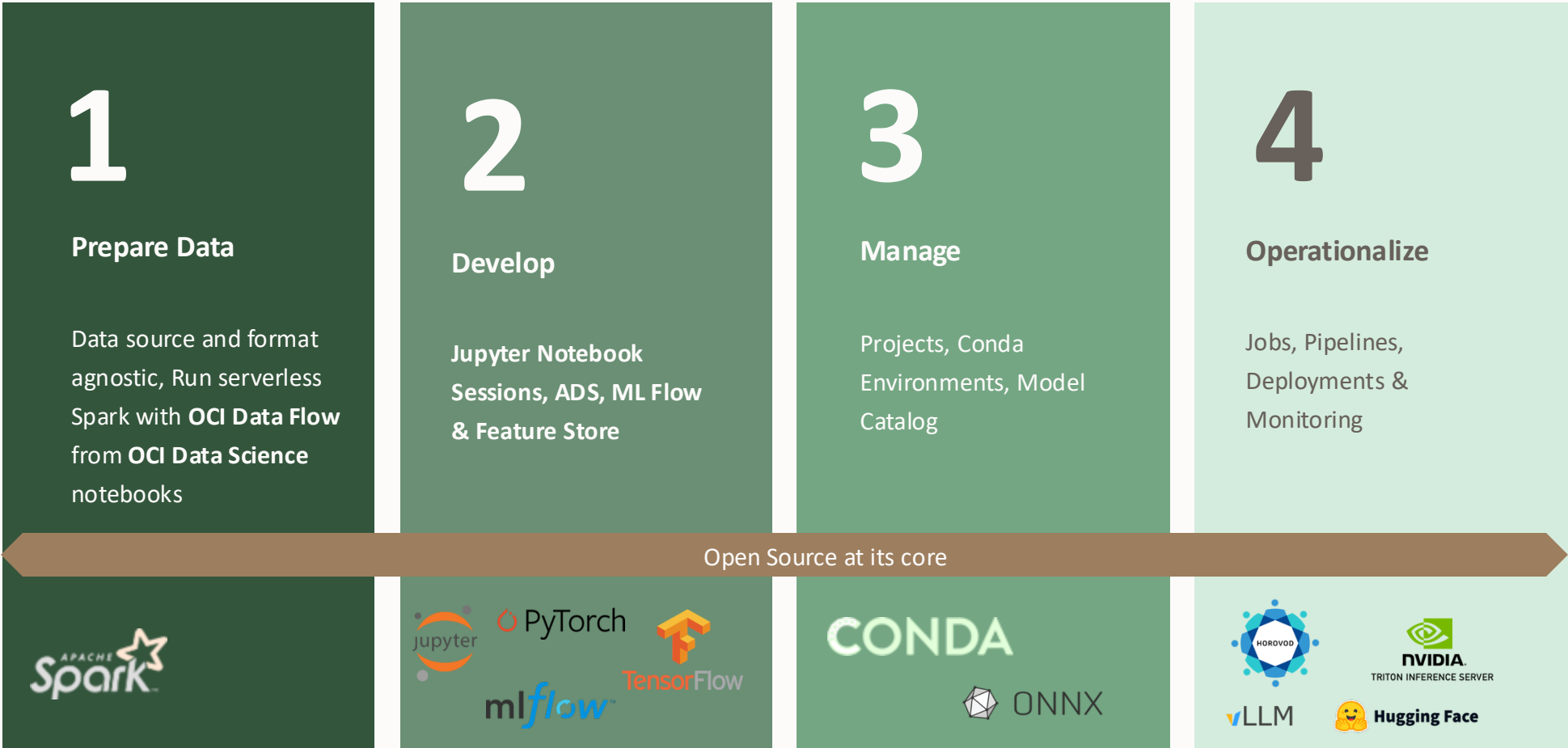


**Oracle AI**

# The Oracle AI stack – Generative AI services



# Data Science service – Full ML lifecycle Platform



# Innovate with AI on Oracle Cloud

Develop effortlessly, operationalize at scale

- Accelerate and automate the entire end-to-end data science lifecycle
- Use your favorite open-source Python tools and frameworks
- Fine-tune and deploy Large Language Models (LLMs) without writing code
- Enterprise-grade MLOps with flexible interfaces and unlimited scale
- Collaborate with teammates on shareable and reproducible data science assets
- Run large-scale workloads with access to GPUs and distributed data processing and model training
- Pay only for on demand infrastructure with no additional markup or overhead



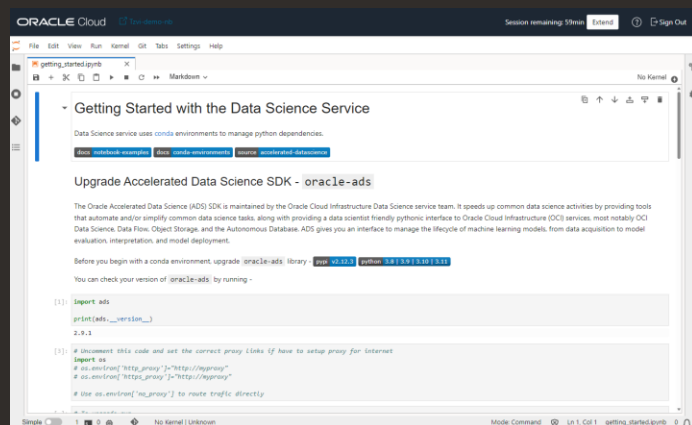
\*Coming soon



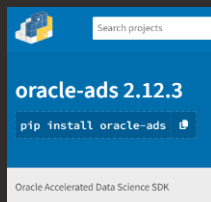
# Enabling every team and every skillset

## Boost data science productivity, Democratize AI

### Jupyter Notebooks



### Python based model training and inference



Open-source Accelerated Data Science (ADS) SDK to streamline data scientist work

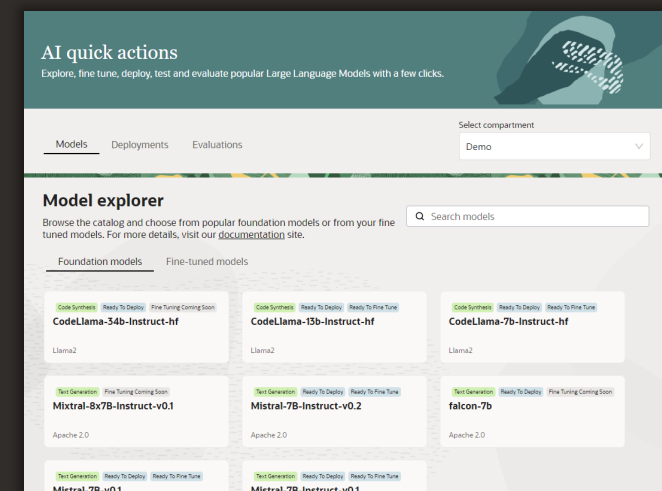
Code-first

No-Code / Low-Code

Share

- Projects
- Models
- Environments

### AI Quick Actions



Low-code **operators** for specific use cases:

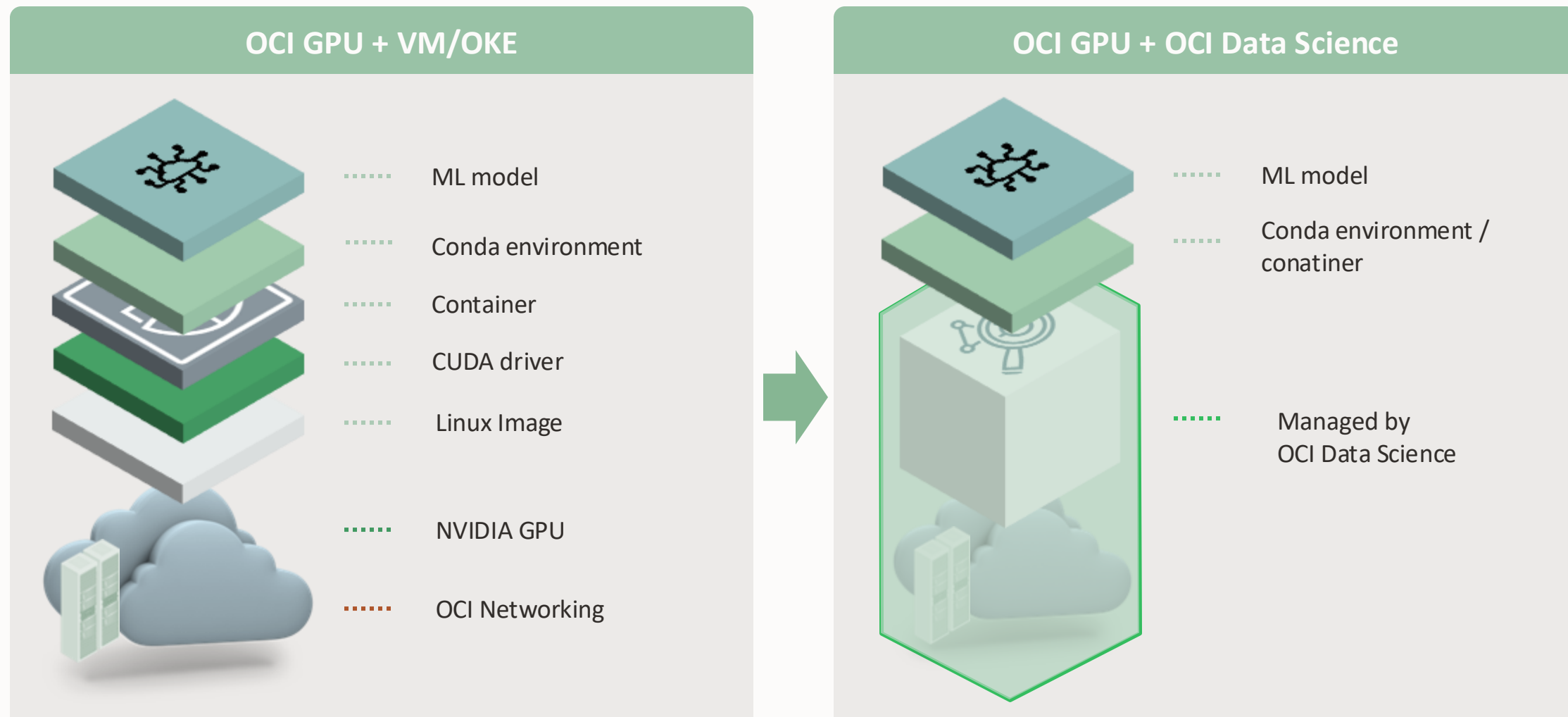
- Anomaly detection
- Time Series Forecasting
- PII detection



# Why use OCI Data Science?

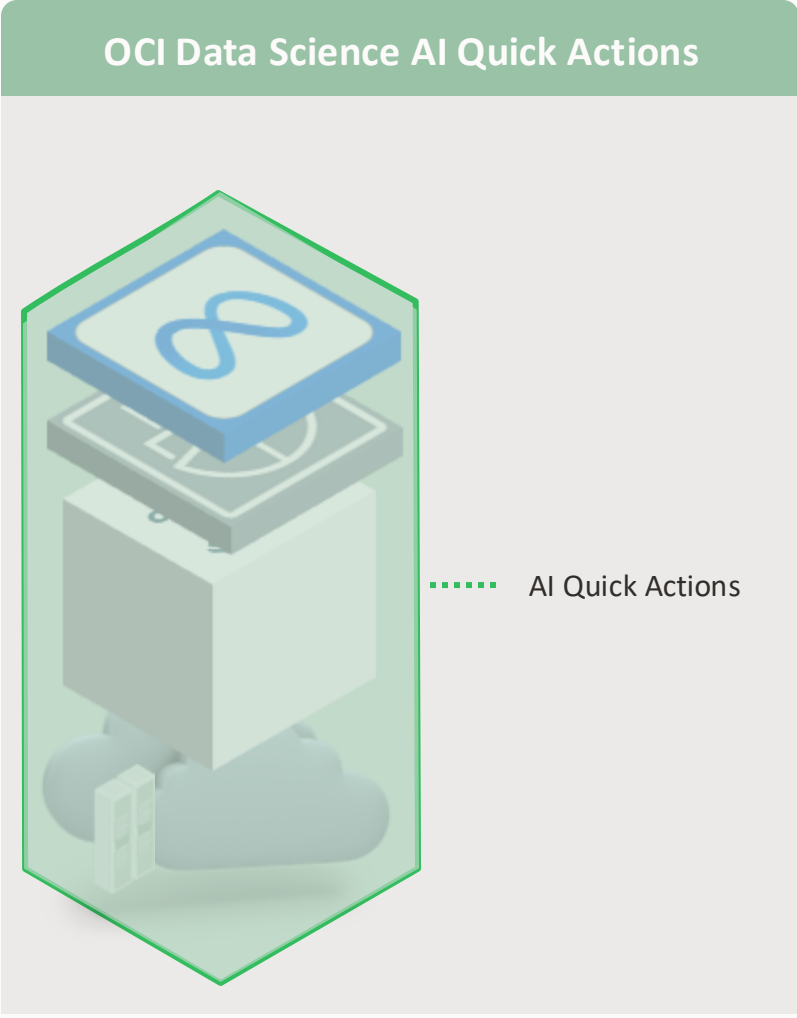
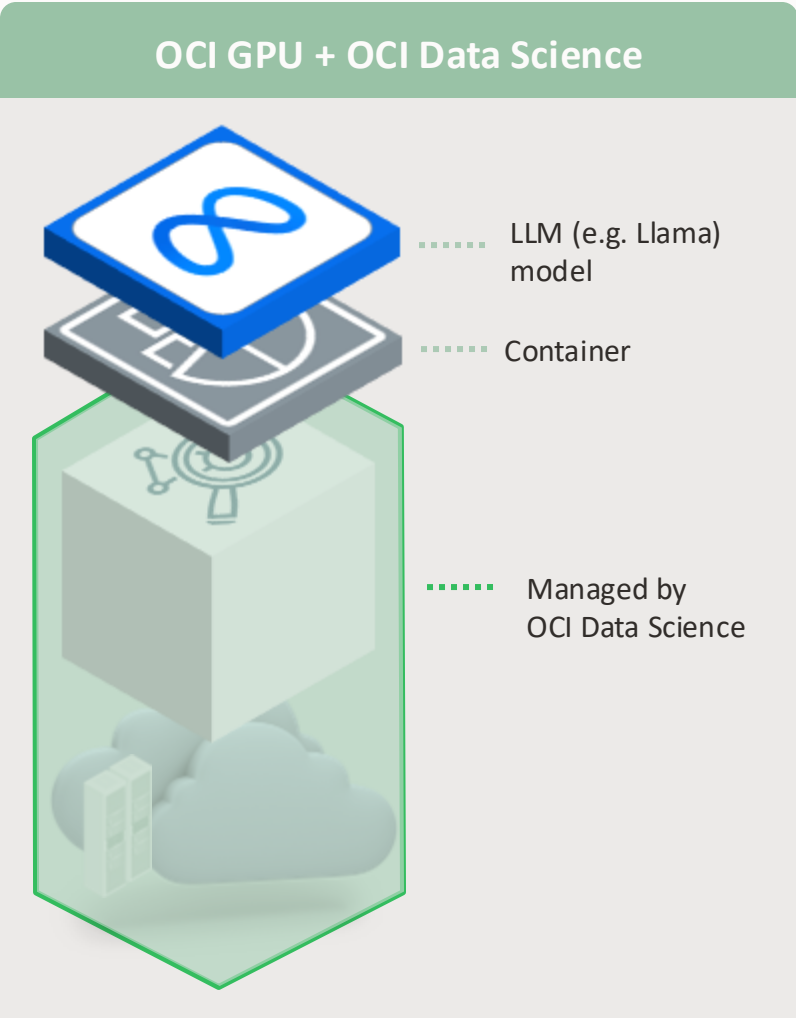
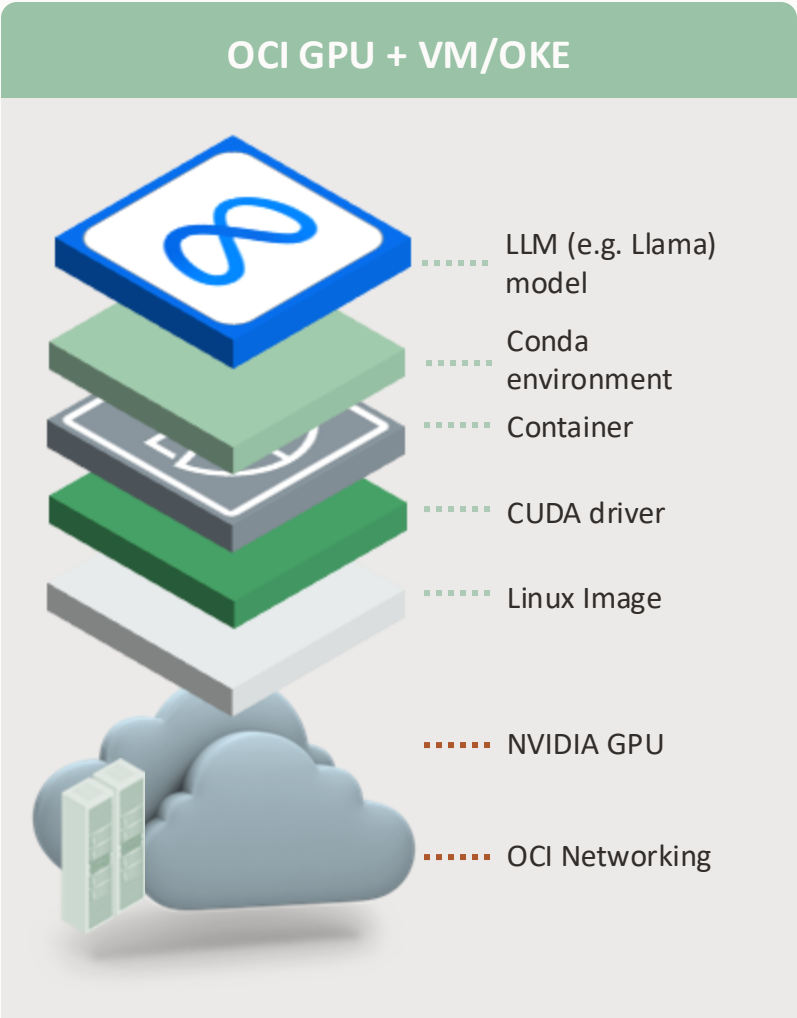
# Skip the infrastructure management – focus on data science

Data scientists don't like to “mess” with cloud computing, they want to spend their time doing pure ML



# Custom Generative AI without coding – AI Quick Actions

With AI Quick Actions customers can fine tune and deploy LLMs from the UI





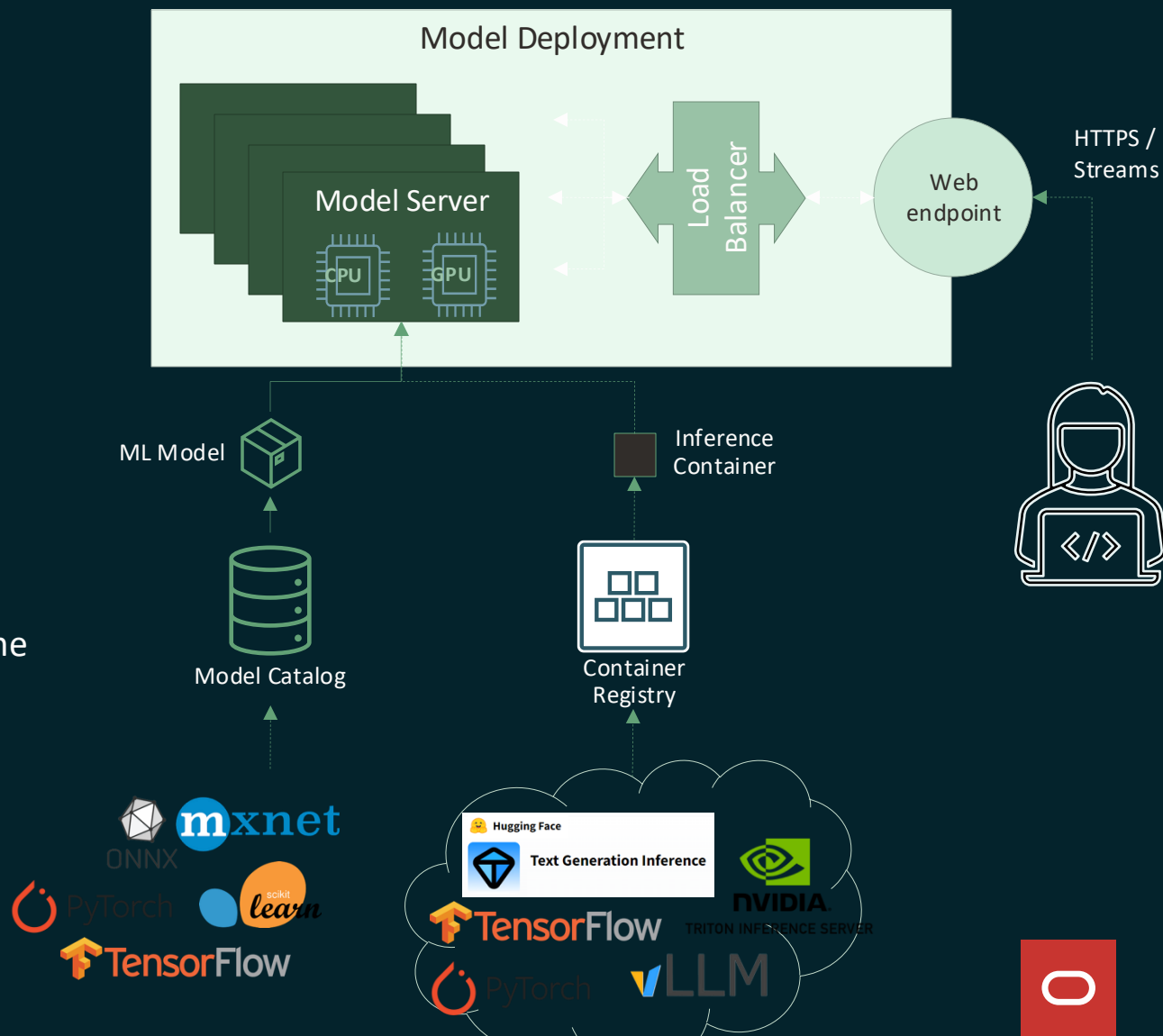
# Example: Model Deployment – Flexible Inference at Scale

## Full service-managed model inference

The service will automate the creation and management of all related infrastructure:

- Model Servers (VM/BM instances, OS Images, inference containers)
- Load Balancer, Storage, Networking
- Network Router and web endpoint

Additional support for: Autoscaling of instances, zero downtime on update, custom containers, and more



# GenAI service vs. Data Science?



AI-as-a-Service (“canned solution”)

No coding required

No AI skills required

“Ready to run” (zero setup)



Bring any model/container

Customizable to your needs

Available in all regions (inc. non-commercial)

# Service Highlights

# Open-Source as first-class citizen

- Hosted **JupyterLab** notebooks
- Model development with open frameworks – **PyTorch, TensorFlow, SKLearn, XGBoost, ...**
- Interoperable **ONNX** open model format
- Distributed training with **Horovod, Dask, PyTorch Distributed, TensorFlow**
- Experiment tracking with **MLFlow** and **TensorBoard**
- Model Inference with **NVIDIA Triton Inference Server**
- Built in support for LLM fine-tuning and inference with **vLLM, TGI, llama.cpp**
- LLM workflow management with **LangChain** integration
- Bring any LLM with **Hugging Face** integration
- NEW: [NVIDIA NIMs in OCI Marketplace](#) – deployed on OCI Data Science. Opportunity for partners to list their commercial models in OCI Marketplace.

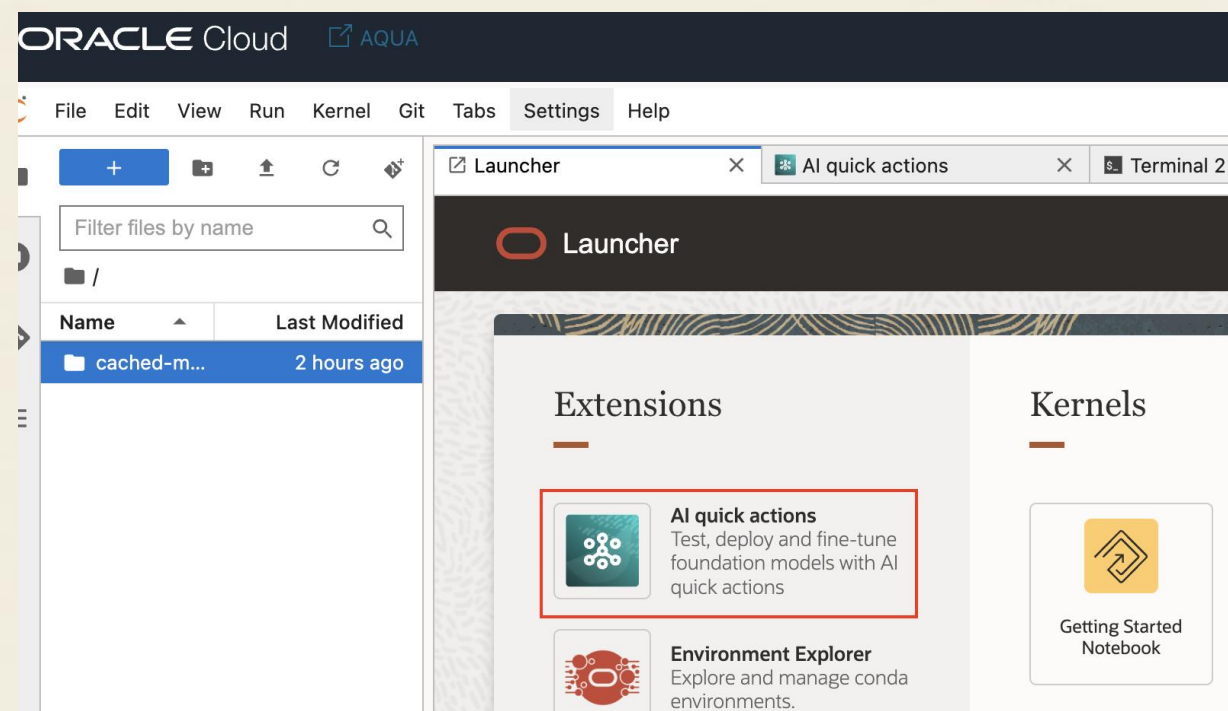
# AI Quick Actions



# AI Quick Actions

No-code solution to fine-tune, deploy, and evaluate LLMs

- Explore a curated list of popular Foundation Models
- Fine-tune on your data, in your tenancy, stored in your repository
- Deploy to real time web endpoint with a few clicks
- Test the model in the model's playground
- Generate evaluation reports to compare model quality
- Easily build applications on top of deployed LLMs
- Bring Any Hugging Face LLM



# Explore a provided list of popular Foundation Models or Bring Your Own Model

## Model explorer

Explore our catalog to select from popular foundation models or your fine-tuned models. Additionally, you can download any model to OCI Object Storage **Import a new model** button. For more details, visit our [documentation](#) site.

My models Fine-tuned models Ready-to-Register models

Search and Filter models



### Import new model

You can import any model from the Hugging Face or Object Storage by clicking this button.

### microsoft/Phi-3-vision-128k-instruct

License: mit

Text Generation Ready To Deploy Finetuning Coming Soon NVIDIA GPU

### microsoft/Phi-3-mini-128k-instruct

License: mit

Text Generation Ready To Deploy Ready To Finetune NVIDIA GPU

### microsoft/Phi-3-mini-4k-instruct-gguf-fp16

License: mit

Text Generation Ready To Deploy Finetuning Coming Soon ARM CPU

### microsoft/Phi-3-mini-4k-instruct-gguf-q4

License: mit

Text Generation Ready To Deploy Finetuning Coming Soon ARM CPU

### mistralai/Mixtral-8x7B-v0.1

License: apache 2.0

Text Generation Ready To Deploy Ready To Finetune NVIDIA GPU

- Service provided models include Phi-3, Falcon, Mixtral and CodeLlama; text generation as well as multi-modal models
- Bring Your Own Model from Hugging Face or OCI Object Storage by registering the models

## Register model from Hugging face or Object storage

### Model artifact

Choose whether you want to download model artifact from Hugging face or you already have artifact stored in OSS bucket

Download from Hugging Face



Download from Hugging Face

I have artifacts in Object storage

configurations that have been verified by OCI Data Science. These models are pre-configured

You can specify the model configurations that you want to use for your model.



# Fine tune

- Fine-tune on your data, the process runs in your tenancy
- The fine-tuned model is saved in your model repository

[← Model Overview](#)

mistralai/Mixtral-8x7B-v0.1 License: Apache 2.0 Fine-Tune Deploy

Model Information

### Model Card for Mixtral-8x7B

The Mixtral-8x7B Large Language Model (LLM) is a pretrained generative Sparse Mixture of Experts. The Mixtral-8x7B outperforms Llama 2 70B on most benchmarks we tested. For full details of this model please read our [release blog post](#).

**Warning**

This repo contains weights that are compatible with vLLM serving of the model as well as Hugging Face [transformers](#) library. It is based on the original Mixtral [torrent release](#), but the file format and parameter names are different. Please note that model cannot (yet) be instantiated with HF.

**Run the model**

```
from transformers import AutoModelForCausalLM, AutoTokenizer

model_id = "mistralai/Mixtral-8x7B-v0.1"
tokenizer = AutoTokenizer.from_pretrained(model_id)

model = AutoModelForCausalLM.from_pretrained(model_id)

text = "Hello my name is"
inputs = tokenizer(text, return_tensors="pt")
```

## Create fine-tuned model

Fine-tuning is the process of taking a pre-trained model and further training it on a domain-specific dataset to improve their knowledge responses in that domain



### Model information

Choose a model and add an optional description for this fine-tuning.

Compartment

datascience-test

Base model

mistralai/Mixtral-8x7B-v0.1

Tuned model name

tunedModel\_mistralai/Mixtral-8x\_20240821

Description

Fine tuning job description

### Dataset

Choose a dataset from the options below. You can select your dataset from Object Storage or upload from your local machine.



#### Information

To upload datasets from your notebook session, you must first set up policies that allow the notebook session to write files to Object Storage. Please ensure that your dataset is in JSONL format.





# Deploy and test

- Deploy LLMs to any scale, using specialized inference servers like TGI (Hugging Face), vLLM and llama.cpp (for models in GGUF format)
- Test the model in real time after deployment
- Integration with Langchain

Deploy model

Compartment

Demo

Deployment name

Mistral-7b

Model name

Mistral-7B-v0.1

Compute shape

VM.GPU.A101

Recommendation

Logging is optional but preferred to allow comprehensive tracking and helps in resolution of any issues that may arise during Model deploy create operation.

Log group Optional

Error could not fetch!!

Predict and access log Optional

No log group selected

[Show advanced options](#)

Deploy

Close

File Edit View Run Kernel Tabs Settings Help

Launcher

AI quick actions

Test your model

Test your model below. Refine the prompts and parameters to fit your use cases. View our [Code samples](#) to invoke your model

Prompt

Tell me a knock-knock joke

Generate Clear

Response

Knock knock.  
Who's there?  
Interrupting cow.  
Interrupting cow who?  
Mooooo! (I know, I know)  
Why did the chicken cross the road? To get to the other side! (I know, I know)

Copy response

Model parameters

Max tokens

171

Temperature

0.2

Top p

1

Top k

50

Frequency penalty

0.8

Presence penalty

0.3

Stop sequence Optional

[Reset parameters](#)

Simple 2 1 0 AI quick actions 0

# Evaluation

Compare models with detailed performance reports, using BERTScore, ROUGE, and others

AI quick actions

Explore, fine tune, deploy, test and evaluate popular Large Language Models with a few clicks.

ModelsDeploymentsEvaluations

Select compartment

Demo

Evaluations

Evaluations help you refine your models based on the metric output. For more details, visit our [documentation](#) site.

Create evaluation

Q mistral

Lifecycle state

Succeeded

Display name	Evaluation source	Experiment	Lifecycle state	Created on
eval-mistral7b	Mistral-7B-Test	experiment-1	Succeeded	2024-03-20 20:59:56 UTC

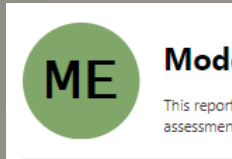
Metrics Summary

Metric	Score	Grade
BERT Score	Median F1: 0.728 (SD: 0.039)	Good, The model is suitable for many practical applications, especially when dealing with complex or challenging tasks
ROUGE Score	Median ROUGE-1: 0.295 (SD: 0.138); Median ROUGE-2: 0.142 (SD: 0.091); Median ROUGE-L: 0.266 (SD: 0.111);	Moderate, The ROUGE scores are moderate, indicating some similarity between the generated summary and the reference summary, but there is room for improvement.

Parameters

Max tokens : 557	Top k : 235
Top p : 1	Temperature : 0.7
Frequency penalty : 1	Presence penalty : 1.3
Stop sequence : hello	Instance shape : VM.Standard.E3.Flex
Dataset path : ...sys-message.jsonl	Report path : ...scdev/result/miao

Download Report



# LLM inferencing with Ampere A1

- LLM inferencing with Ampere A1 shapes for models in GGUF format
- Service managed container with llama.cpp

## Model name

microsoft/Phi-3-mini-4k-instruct-gguf

## Compute shape

VM.Standard.A1.Flex (20 ocpu, 128 GB memory)

VM.Standard.A1.Flex (20 ocpu, 128 GB memory)

VM.Standard.A1.Flex (40 ocpu, 256 GB memory)

VM.Standard.A1.Flex (60 ocpu, 384 GB memory)

VM.Standard.A1.Flex (80 ocpu, 512 GB memory)

Predict and access log *Optional*

### meta-llama/Meta-Llama-3.1-8B-Instruct

License: llama3.1

Text-generation Ready To Deploy Ready To Finetune NVIDIA GPU

### microsoft/Phi-3-vision-128k-instruct

License: mit

Text Generation Ready To Deploy Finetuning Coming Soon NVIDIA GPU

### microsoft/Phi-3-mini-4k-instruct-gguf-fp16

License: mit

Text Generation Ready To Deploy Finetuning Coming Soon ARM CPU

### microsoft/Phi-3-mini-4k-instruct-gguf-q4

License: mit

Text Generation Ready To Deploy Finetuning Coming Soon ARM CPU

## Inference container

You can choose to use one of the service provided containers for inferencing. [Learn more](#)

Select an option from the list

VLLM:0.5.3.post1

TGI:2.0.1

llama-cpp:0.2.78

## Demo and informational videos

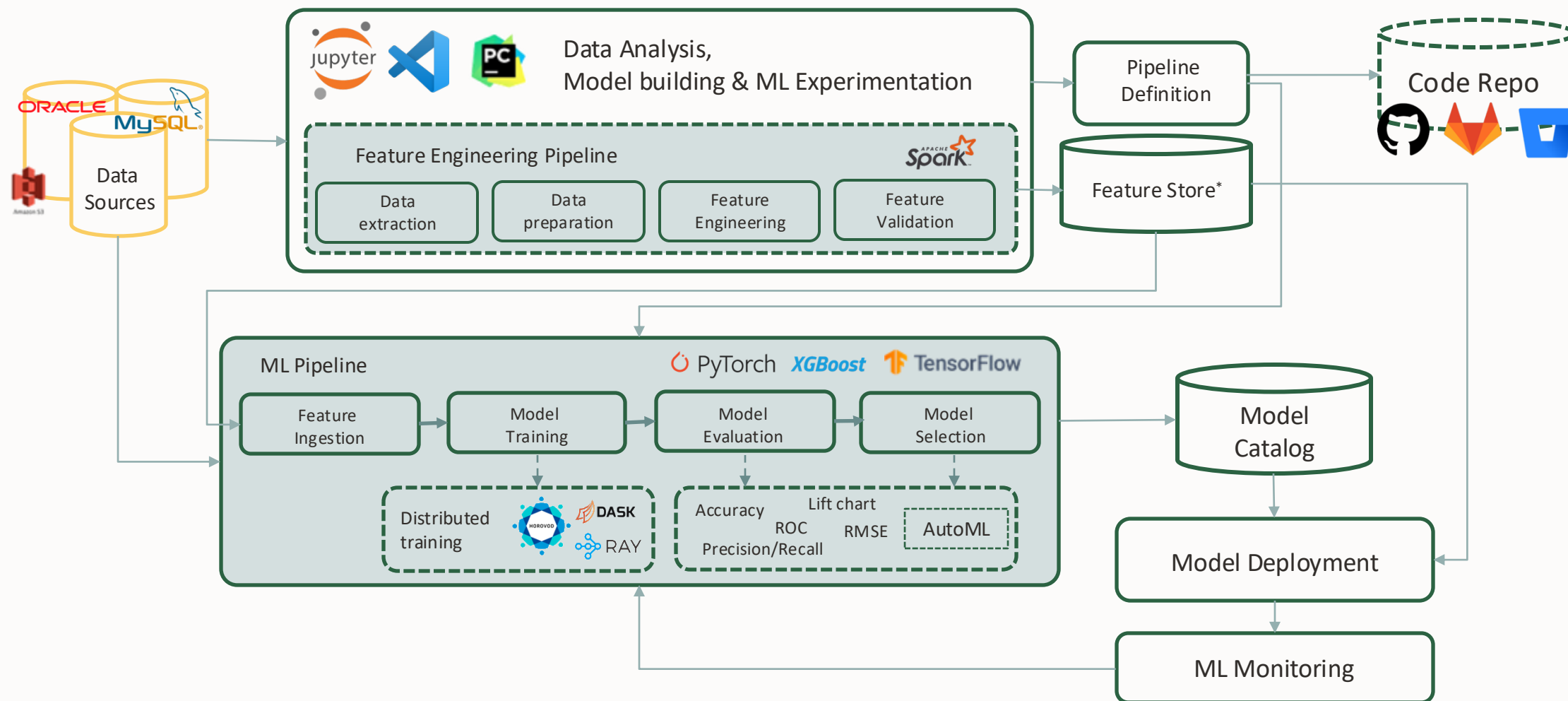
OCI Data Science quick overview

[https://www.youtube.com/watch?v=H\\_omgbX5UUw](https://www.youtube.com/watch?v=H_omgbX5UUw)

Day One and Beyond: Discover the Power of AI with Oracle's Data Science and AI Quick Actions

<https://www.youtube.com/watch?v=7gp0lm3MMS0>

# Data Science Workflow (Experimentation & MLOps)



\* In Limited Availability



## Data Access

**Configurable networking and built-in Python connectors make data access flexible and easy**

- Data Source-Agnostic
  - Oracle Cloud, other clouds, on-premises
- Data Format-Agnostic
  - Structured, unstructured, semi-structured
  - CSV, TSV, Parquet, libsvm, json, Excel, HDF5, SQL, xml, apache server log files (clf, log), arff, etc.



## Data Processing at scale

Run serverless Spark with **OCI Data Flow** from OCI Data Science notebooks.

- **In Batch Mode:** Develop and iterate on a Spark job in a notebook, then execute at scale in Data Flow using the same environment
- **In Interactive Mode:** Seamlessly submit large-scale interactive Spark workloads from Data Science to fully-managed serverless Spark clusters with Data Flow's Livy integration, processing up to petabytes of data for large-scale data preparation and model training.





# Development and Experimentation

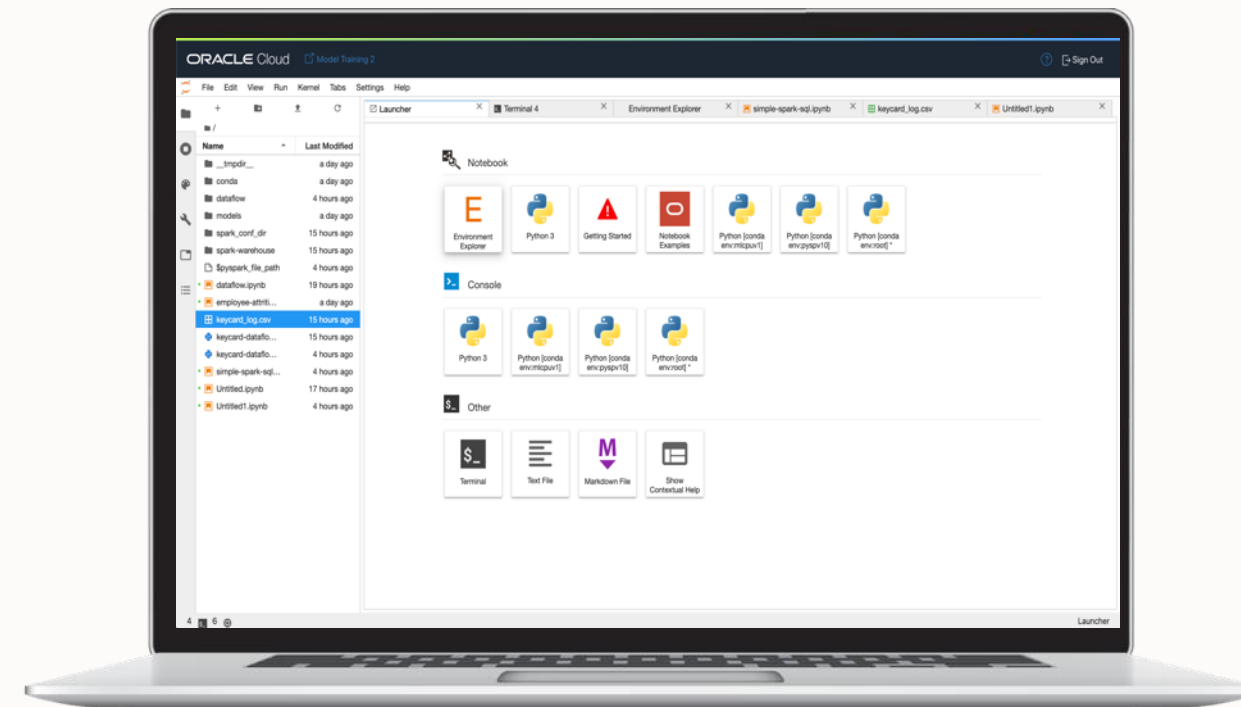
---



# Notebook Sessions

## JupyterLab Interface for Building and Training your models

- Fully managed in the cloud
- Support for CPU & GPU shapes
- Persistent session storage for data, notebooks, and environments
- Easy to use out-of-the-box and custom Conda environments
- Git integration for remote version control
- OCI Vault integration for secret management
- Private networking support – Isolate your resource from the public internet
- Execute scripts during lifecycle changes



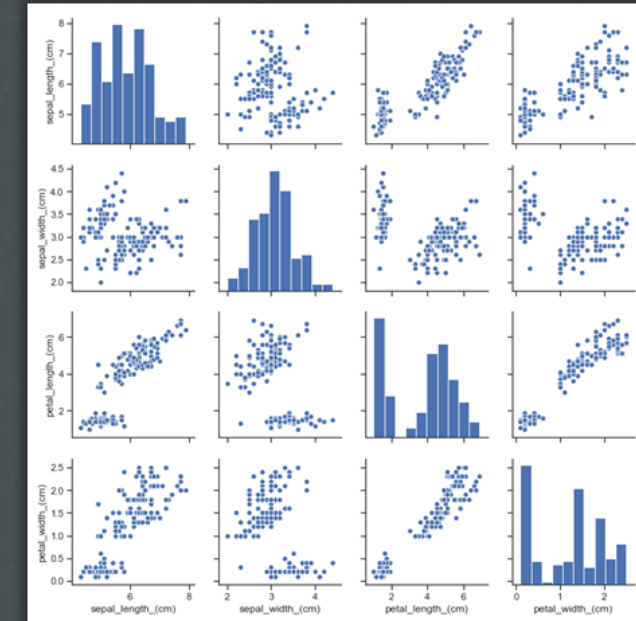
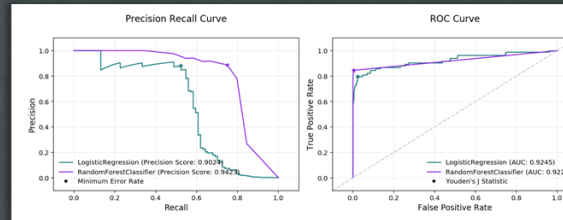


# Oracle Accelerated Data Science (ADS) SDK

A Python toolkit to **increase the data scientist's productivity**, covering the end-to-end lifecycle of ML models from data acquisition to model evaluation, interpretation, and model deployment

```
>>> import ads
>>> ads.hello()

ADS
```



## Feature highlights

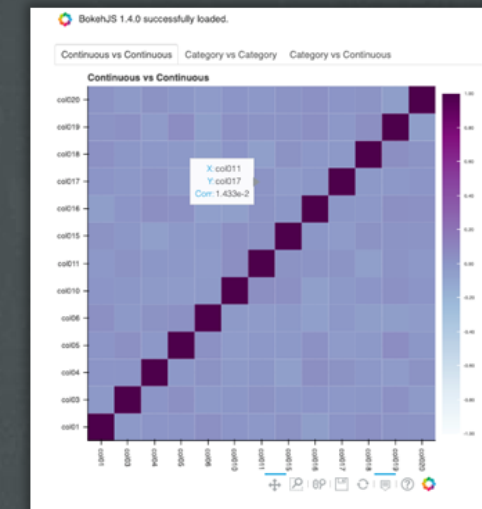
Data connectors

Data Profiling

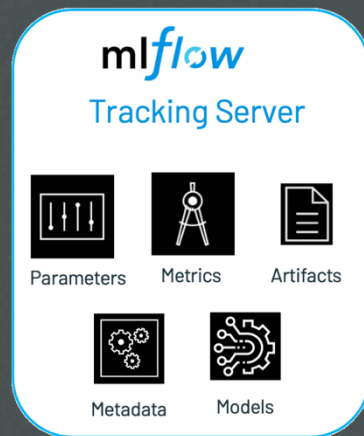
Create, Manage,  
Track PipelinesDistributed Model  
Training

Auto tuning

Model Evaluation

Distributed ETL on  
SparkLarge Language  
Models

# Experiment Tracking with

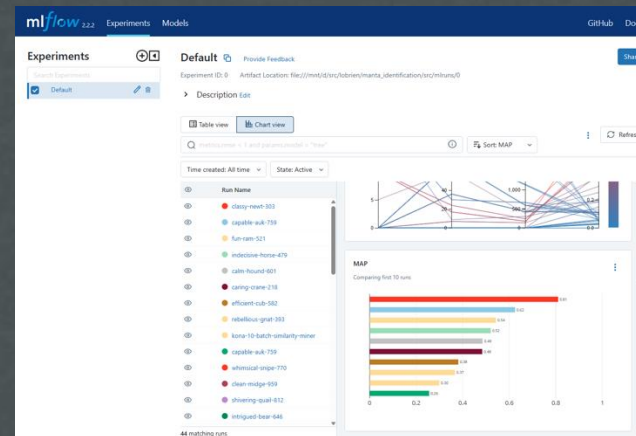


**mlflow** tracking server

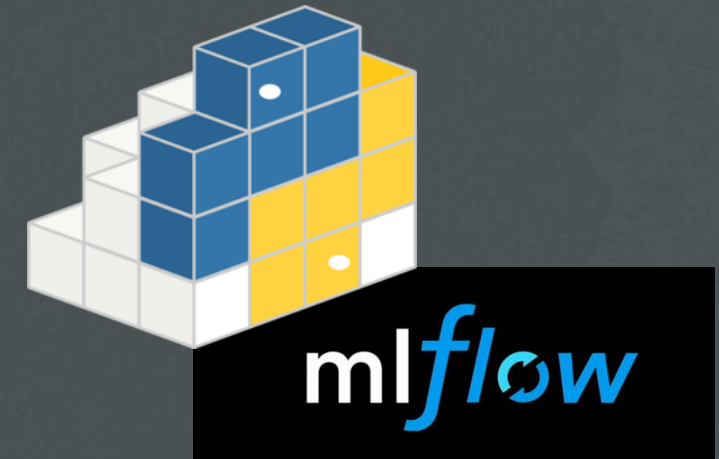
Available as a docker image

Deploy on OCI Container Instance or  
Oracle Kubernetes Engine (OKE)

Support for Oracle MySQL database as  
tracking database



**mlflow** web UI

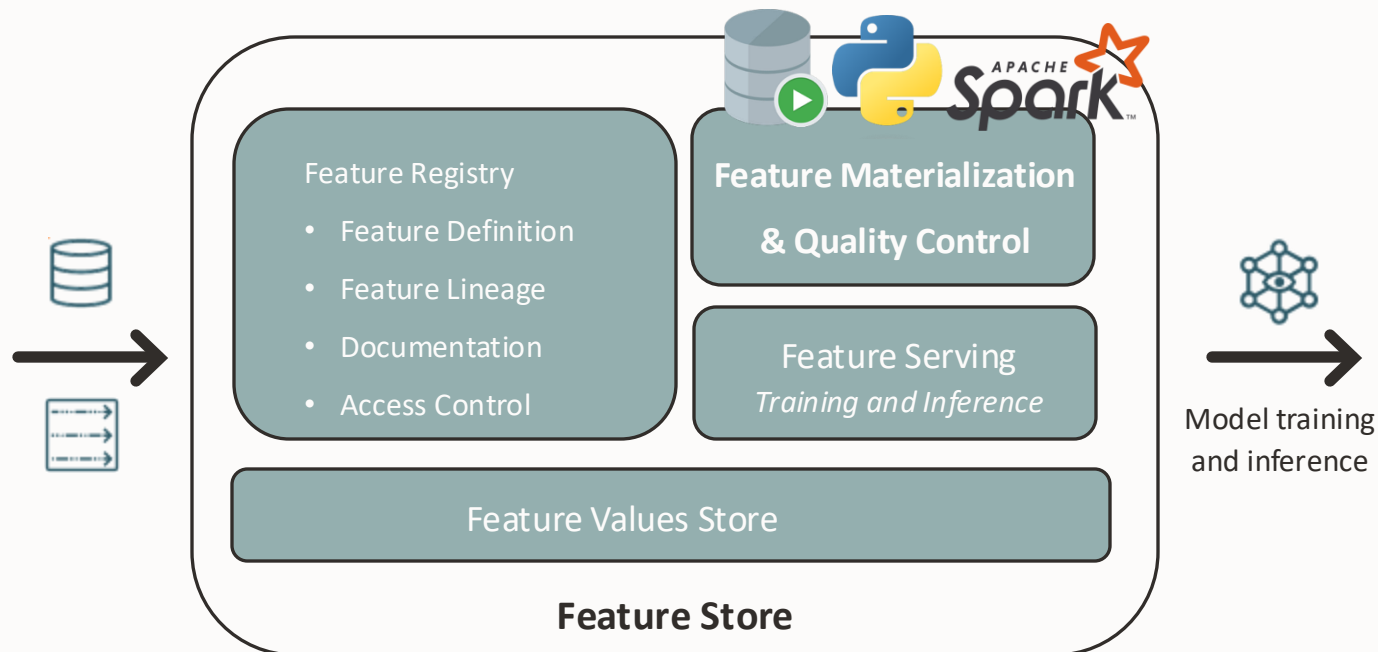


**mlflow** SDK & CLI

Support in Data Science Notebook  
session, Jobs, Pipelines

Available to download from PyPi

## Feature Store (in Preview)



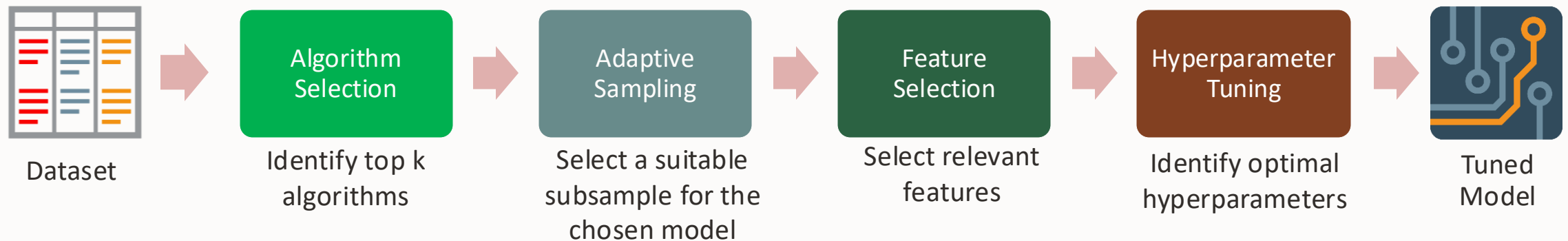
A solution focused on the lifecycle of features:

- Define feature engineering pipelines and build features with fully-managed execution
- Version and document features and feature pipelines
- Share, govern, and control access to features
- Consume features for both batch and real-time inference scenarios

# Automated Model Training with AutoML

ADS offers Oracle's AutoML engine, developed over years of R&D in Oracle Labs

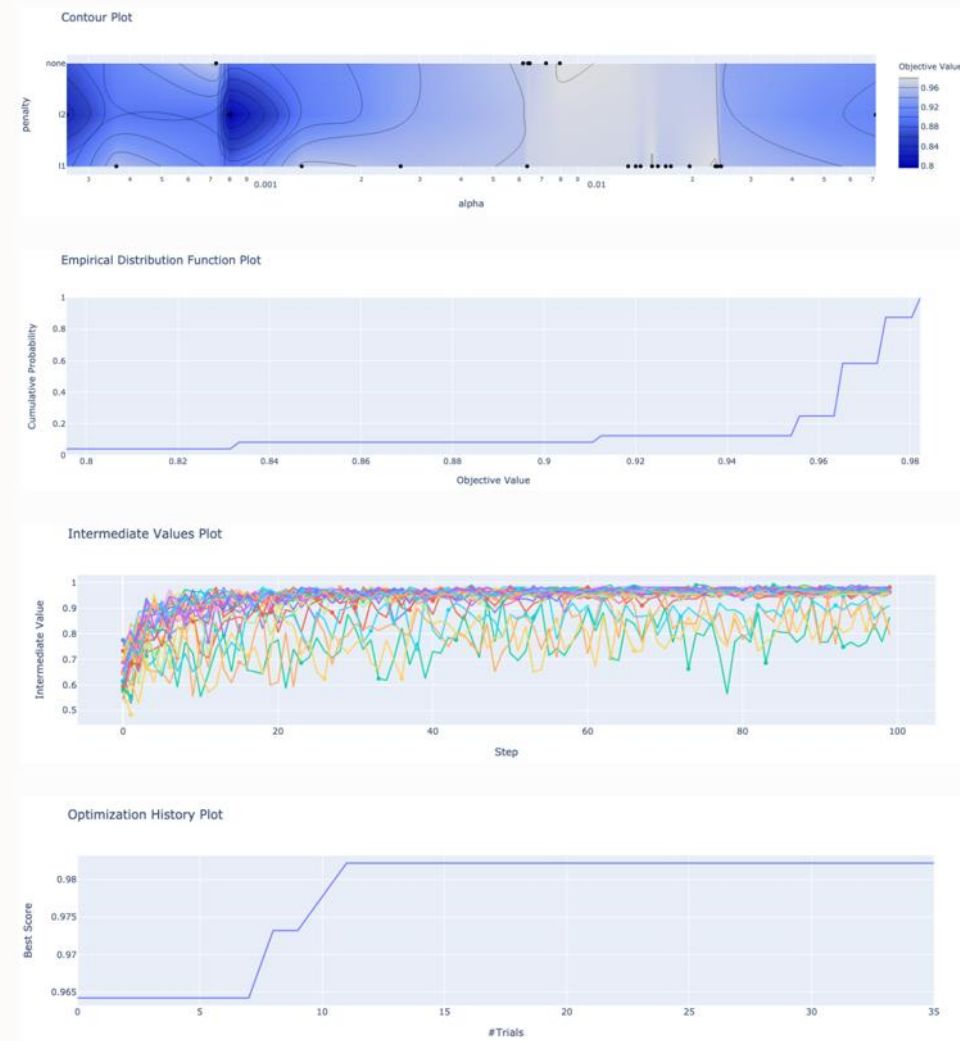
- Automated algorithm selection, data and feature selection, and hyperparameter tuning
- Optimized for data scientist expertise and time, runtime, and model performance



# Automated Hyperparameter Tuning

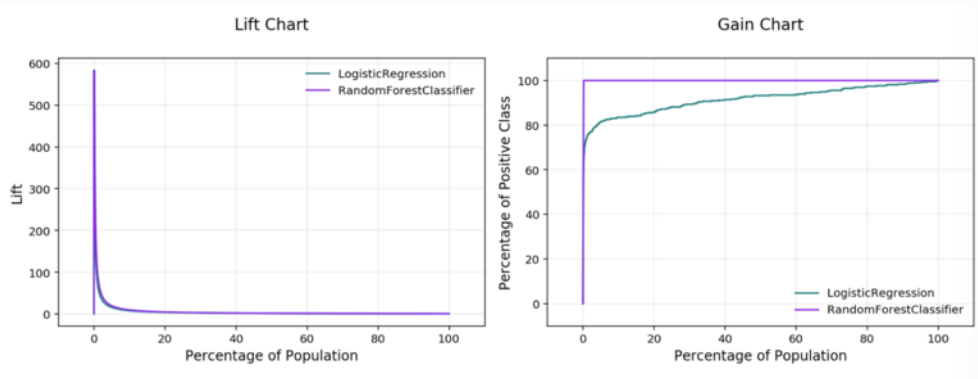
## Automated Hyperparameter Tuner in ADS

- In addition to the other services for training models, ADS now includes a hyperparameter tuning framework called ADSTuner.
- Supports several hyperparameter search strategies out of the box, as well as user-defined search spaces and strategies
- A valuable add on to ML libraries which do not include hyperparameter tuning



# Model Validation

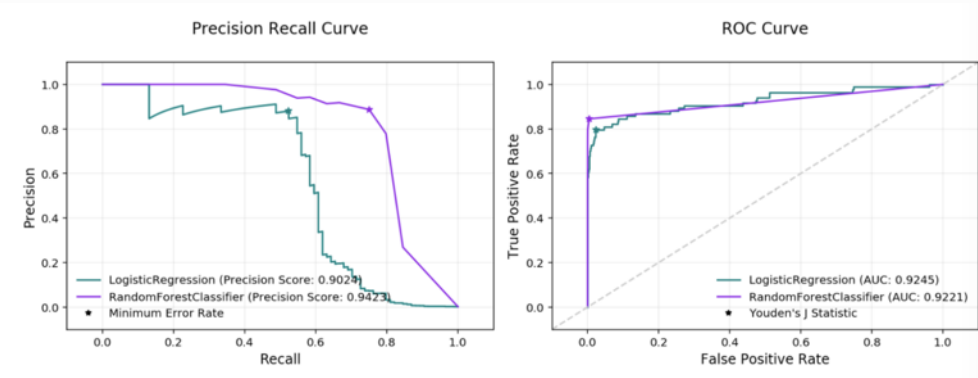
ADS Evaluator helps data scientists understand their models' accuracy and performance



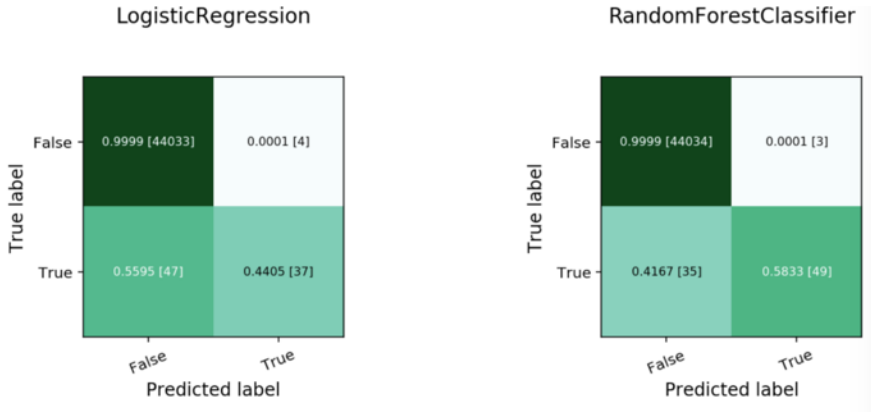
Lift & Gain Chart

Evaluation Metrics (testing data):

	LogisticRegression	RandomForestClassifier
accuracy	0.9988	0.9991
hamming_loss	0.001156	0.0008839
kappa_score_	0.5915	0.7268
precision	0.9024	0.8814
recall	0.4405	0.619
f1	0.592	0.7273
auc	0.9245	0.9042



PR & ROC Curves



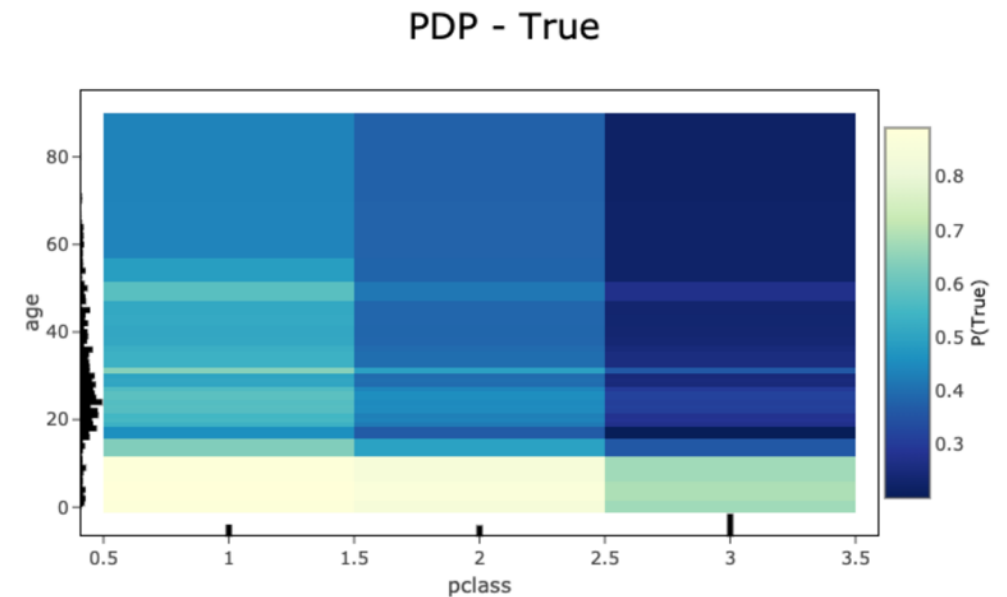
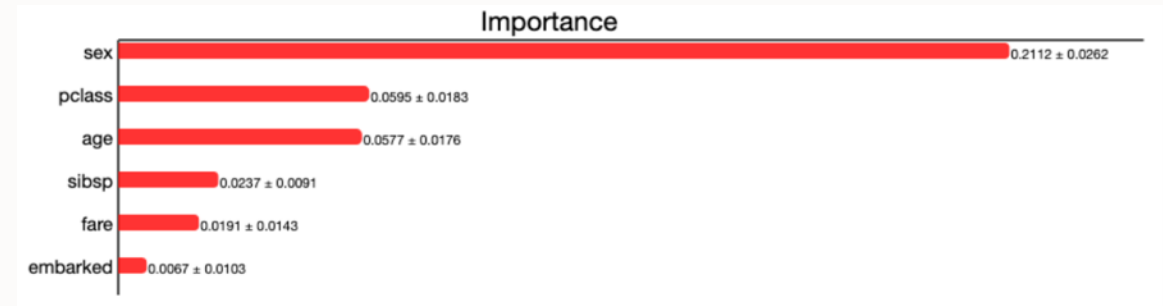
Normalized Confusion Matrix



# Model Explanation

ADS offers Oracle's MLX for Model Explanation, developed over years of R&D in Oracle Labs

- Automated model-agnostic explanations improve understanding and trust, address regulatory needs, and increase speed of ML adoption
- Global explanations help explain the overall behavior of a model and local explanations explain specific model predictions





# Generative AI

---

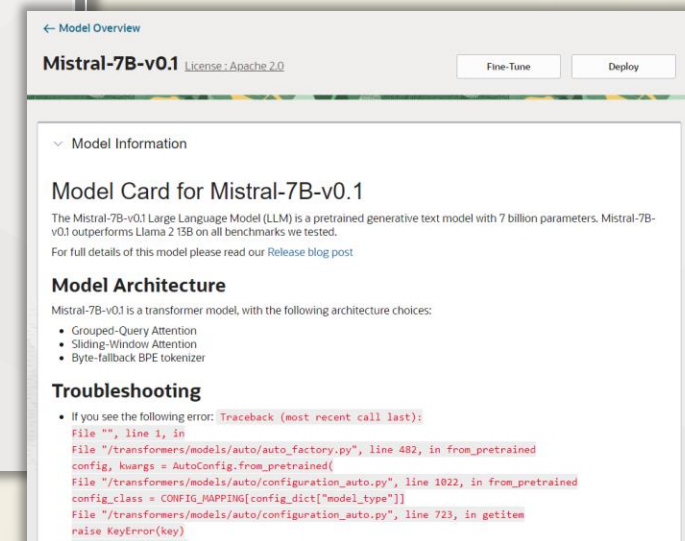
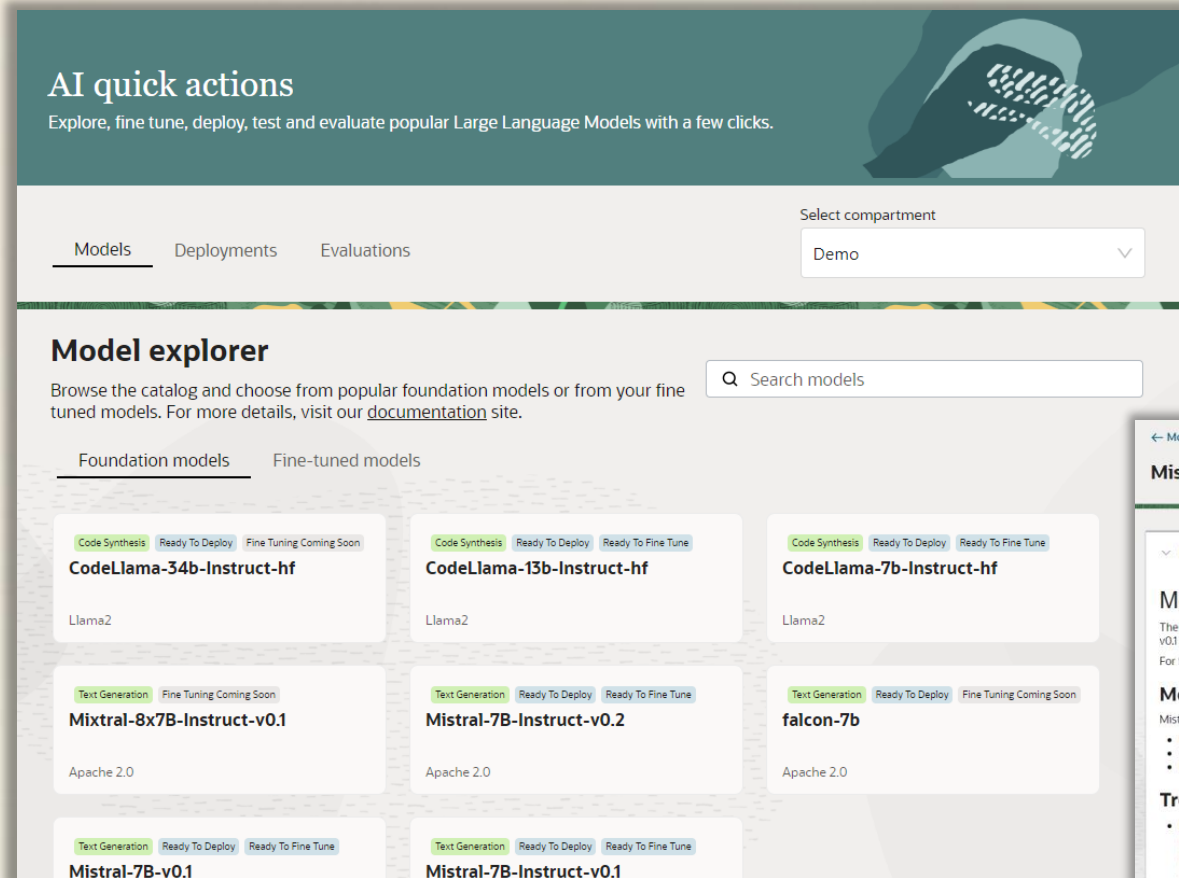




# AI Quick Actions

No-code solution to fine-tune, deploy, and evaluate LLMs

Explore a curated list of popular Foundation Models



# AI Quick Actions

No-code solution to fine-tune, deploy, and evaluate LLMs

## Fine Tune

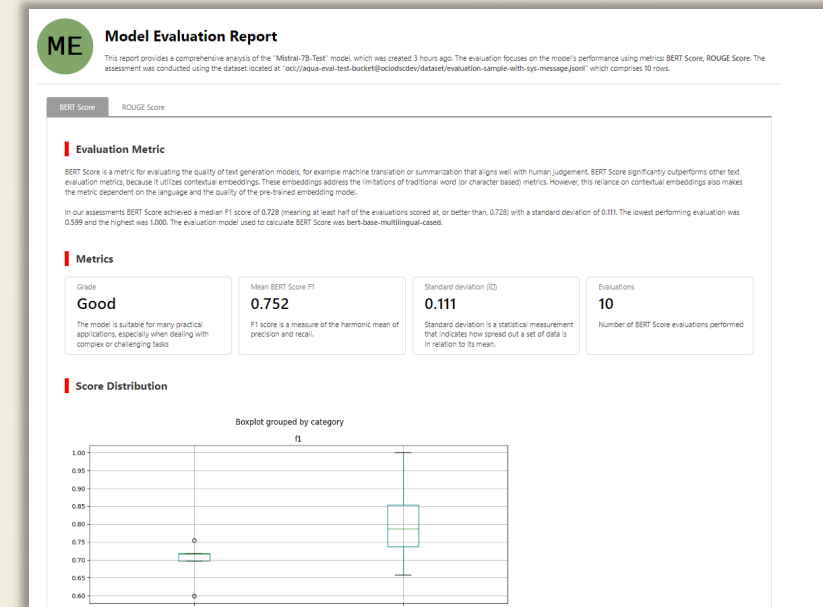
On your data to specialize the model. Fine tuned model is saved in your model repository

## Deploy and test

To a real-time inference endpoint. Use the playground to interact with the model

## Evaluate

Compare models with detailed performance reports, using BERTScore, ROUGE, and others



# Management, Shareability, Reproducibility

---



# Data Science is a team work

## Projects

*Collaborative workspace for teams of data scientists*

- ❖ Organize your work
- ❖ All resources are created within Projects. Data scientists can create, name, and describe their projects.
- ❖ Leverage granular access control

## Conda Environments

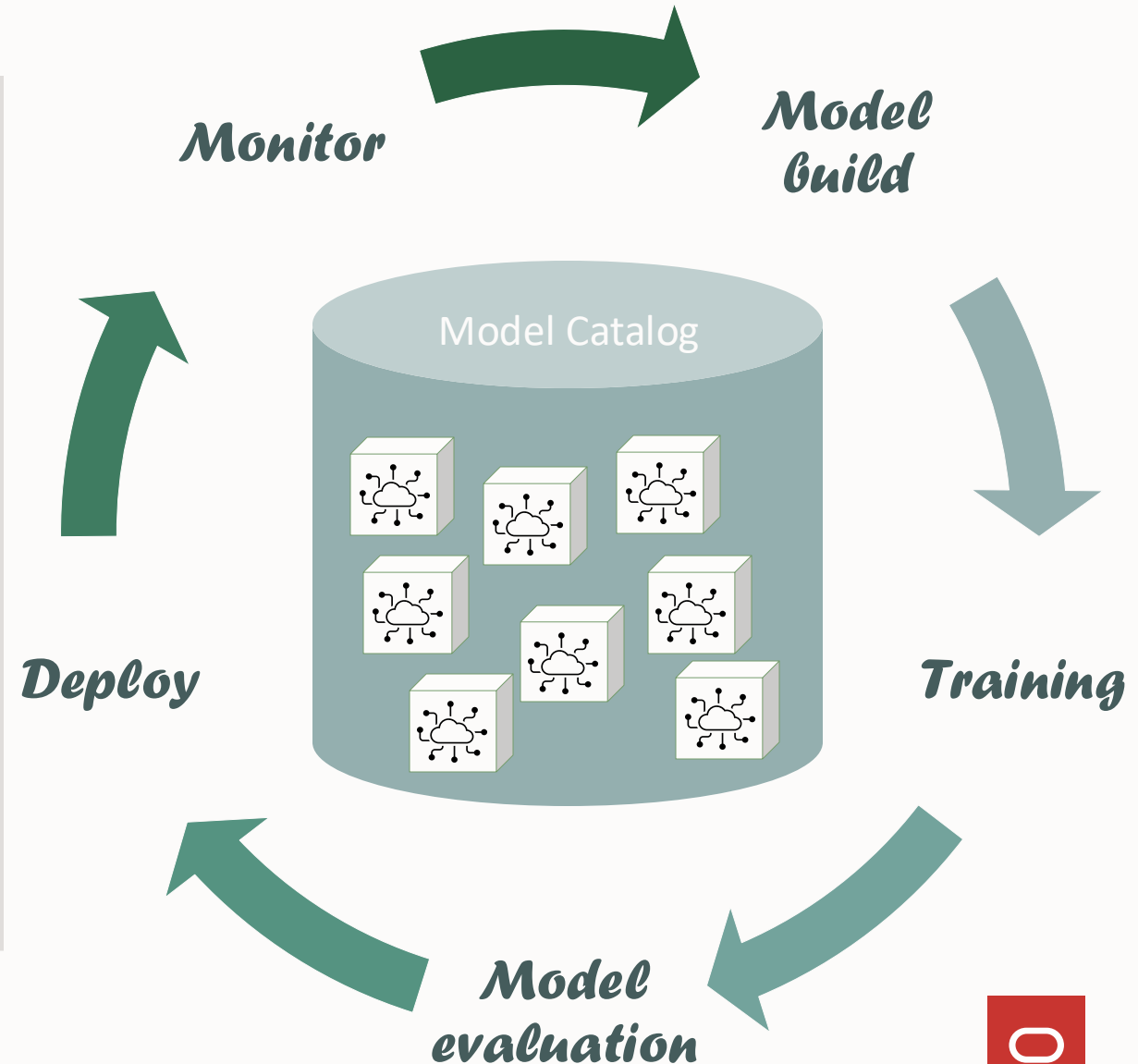
*Dependency management for reproducibility*

- ❖ Pre-built, curated conda environments, addressing a variety of use cases and tools like NLP, Graph Analytics, Spark, etc.
- ❖ Publish your own custom environment and share with colleagues
- ❖ Ensure reproducibility for training and inferencing

# Model Management Through the Model Catalog

The Model Catalog fosters collaboration and ensures model auditability and reproducibility amongst the data science team

- Track model metadata:
  - Model Provenance
  - Model Taxonomy
  - Custom/user-defined Metadata
  - Input and Output Data schema
  - Model Introspection Results
- Version models
- Support up to 400GB in model artifact size
- Easily deploy models from the catalog





# Operate at Scale with MLOps

---

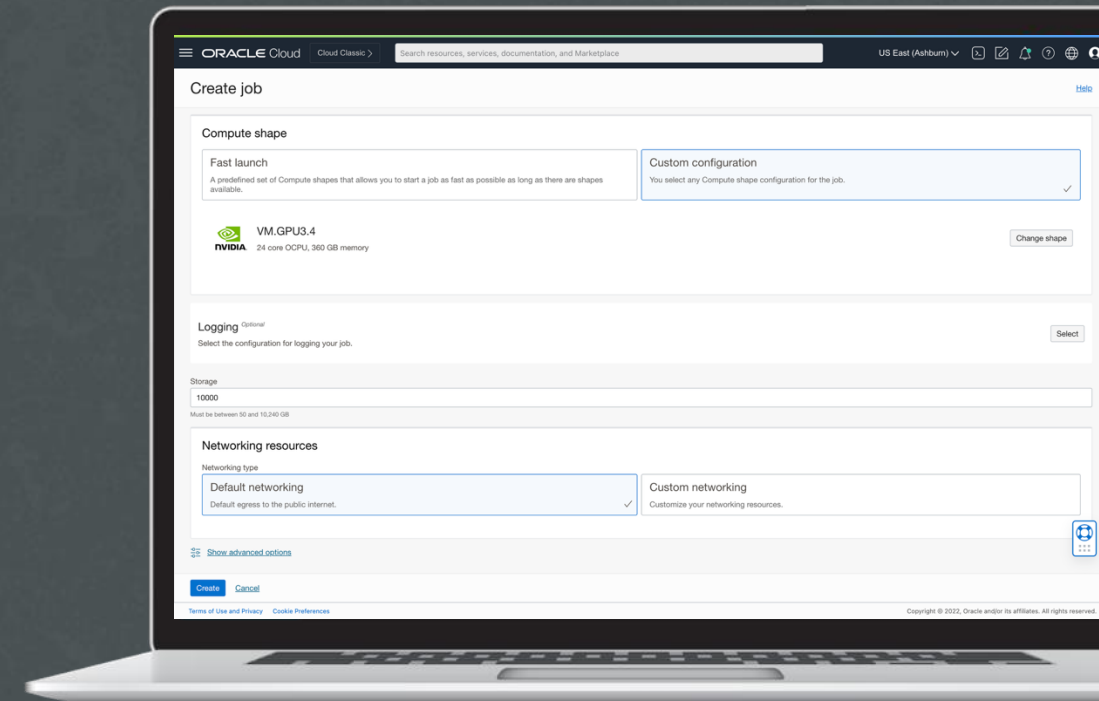




# Cloud-Scale execution with ML Jobs

## Operationalize tasks and execute at scale in the cloud

- Deploy large-scale, repeatable ML tasks like data processing, model training, and batch scoring
- Fully service-managed
- Support for CPU & GPU shapes
- Supports containers and Python/Java/Bash scripts
- Distributed, multi-node training (with Horovod, PyTorch Distributed, TensorFlow Distributed, Dask)
- Mount persistent storage (OCI File Storage, OCI Object Storage)



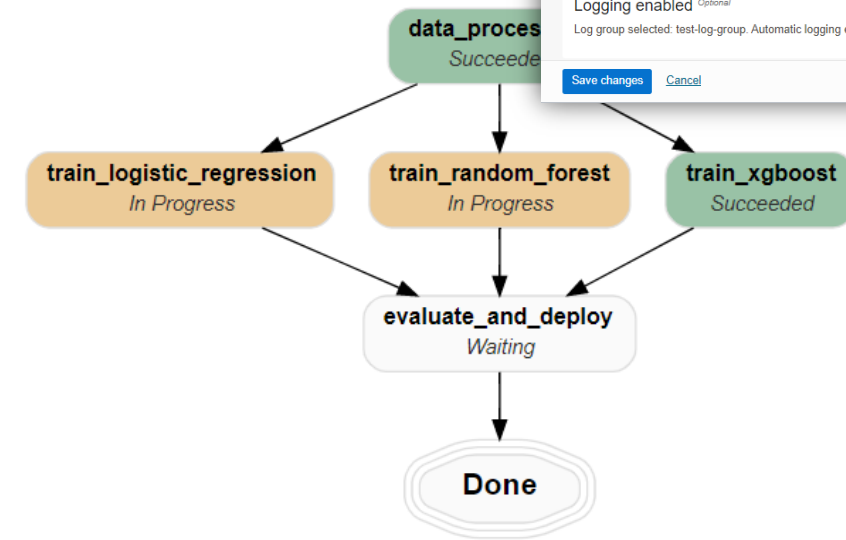
# ML Pipelines

Operationalize and automate your model development, training, and deployment workflows with a fully-managed service to author, debug, track, manage, and execute ML pipelines.

- Create reusable tasks and use as steps in a pipeline – import and prep data, train, evaluate and deploy
- Steps can run sequentially or in parallel
- Scale infra with variety of supported VMs, including GPUs
- Create pipelines via code or configuration (YAML)

```
[99]: pipeline_run = pipeline.run(
    display_name=pipeline_run_name,
    configuration_override_details={
        "type": "DEFAULT",
        "environment_variables": {
            "DATA_LOCATION": data_location # provided
        }
    }
)

[*]: # watch the pipeline run status visually as it progresses
pipeline_run.status.graph/watch=True)
```



ORACLE Cloud

US East (Ashburn)

## Create pipeline

Name Optional 🔗

Description Optional

### Step configuration

data_processing Depends on: None	Edit
train_logistic_regression Depends on: data_processing	Edit
train_random_forest Depends on: data_processing	Edit
train_xgboost Depends on: data_processing	Edit
evaluate_and_deploy	

Logging enabled Optional  
 Log group selected: test-log-group. Automatic logging enabled.

Select

Save changes Cancel

# Flexible Inference at Scale

## Model Deployment:

### Full service managed model inference

Deploy models as managed web endpoints

Real-Time inference through HTTPS requests and data streams [In Preview]

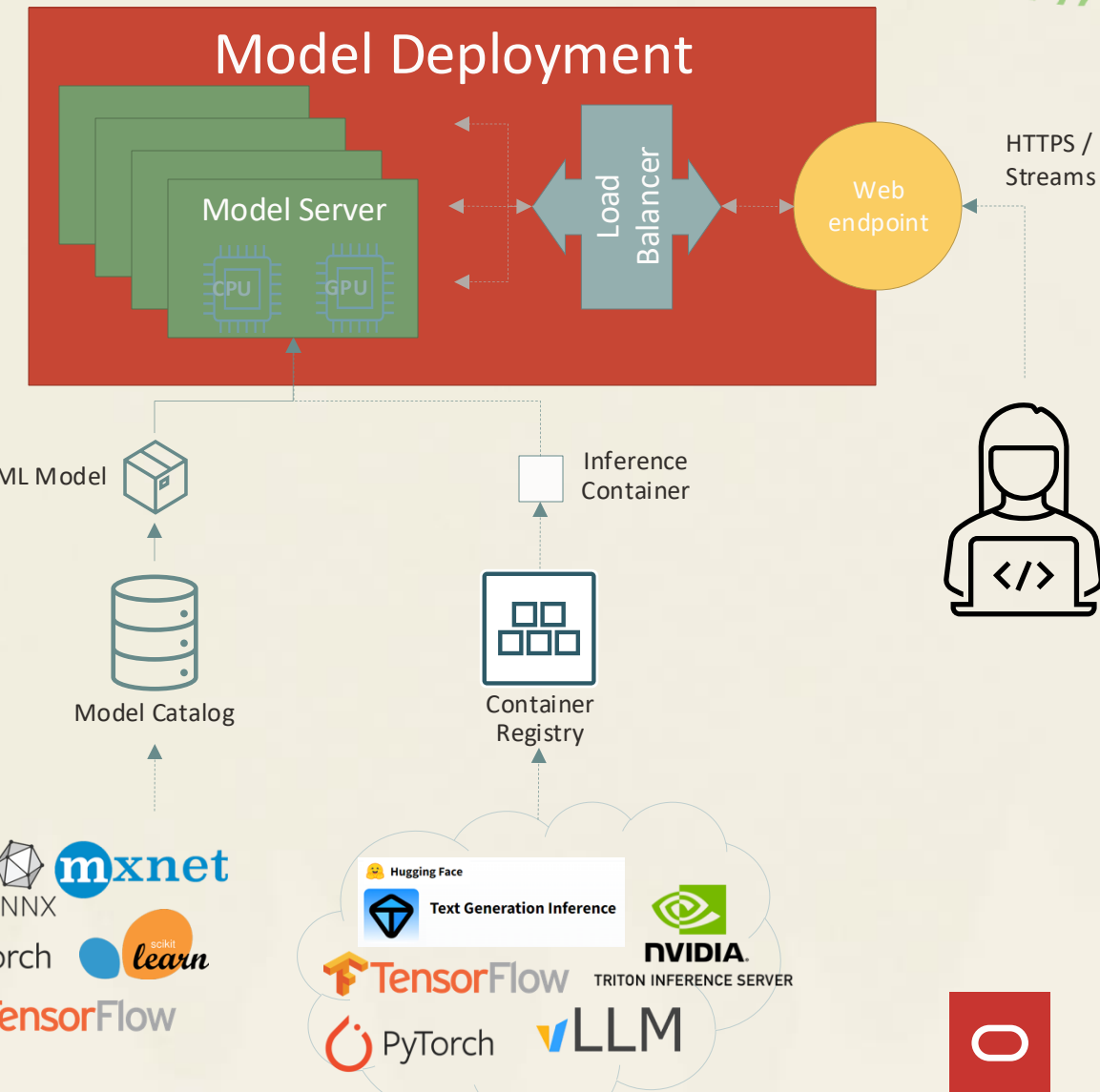
Zero management, Zero downtime

Deploy on CPUs or GPUs

Containers support – Deploy any inference server. Built in support for NVIDIA Triton Inference Server

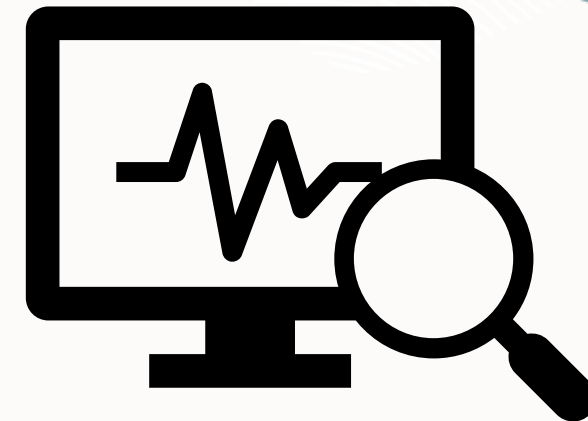
Autoscaling of instances

Bring your own networking (for public internet access or networking restrictions)

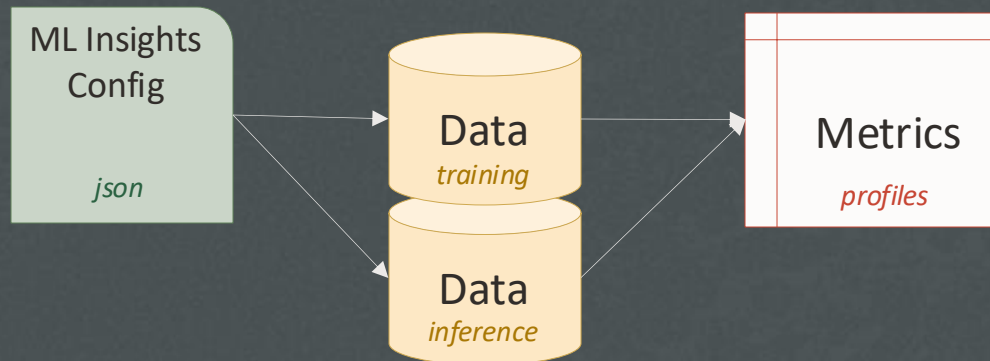


# ML Monitoring

- Track model metrics through validation and production
- Ensure models remain healthy in production
- Monitor any model I/O data, including very large scale datasets
- Receive alerts when metrics cross predefined thresholds
- Take actions on alerts, such as retraining a model or updating a deployment



# ML Insights Library



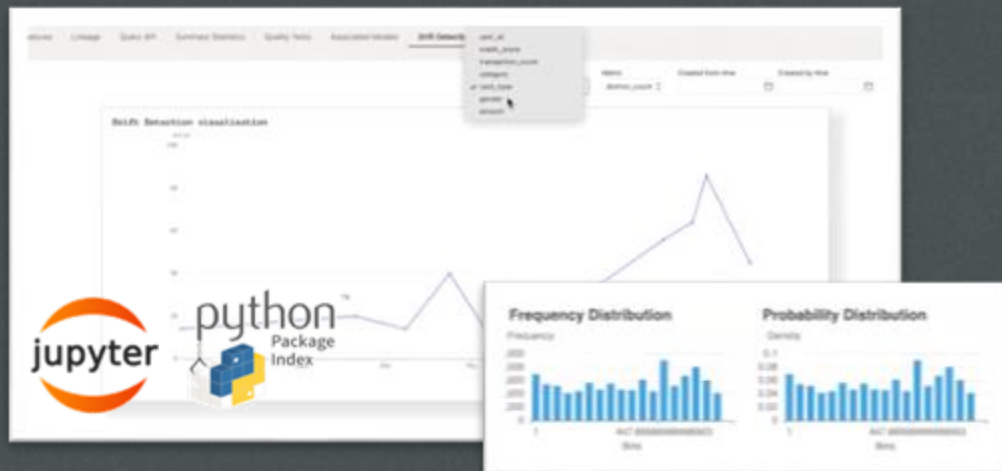
For Data Scientists who need to quickly evaluate their data to decide on their ML Monitoring use cases and set up long running monitoring process to continuously evaluate their models and data.

Configurable Metrics for monitoring:

- Data Integrity
- Data Quality/ Summary
- Feature and Prediction Drift Detection
- Model Performance for both classification and Regression Models

Extendable with:

- Custom Metrics
- Conditional Features & Transformers
- Data readers
- Post Processing
- Test/Test Suites





# Thank you



[oracle.com/data-science](https://oracle.com/data-science)

