



# AAA Club Alliance (“ACA”) ML Workshop

Day 4

03-27-2023



# Agenda



## Overview

Agenda, Introductions to hands on workshop cadence & schedule

## Hands On Use Case

Introductions into the use cases, choosing your use case

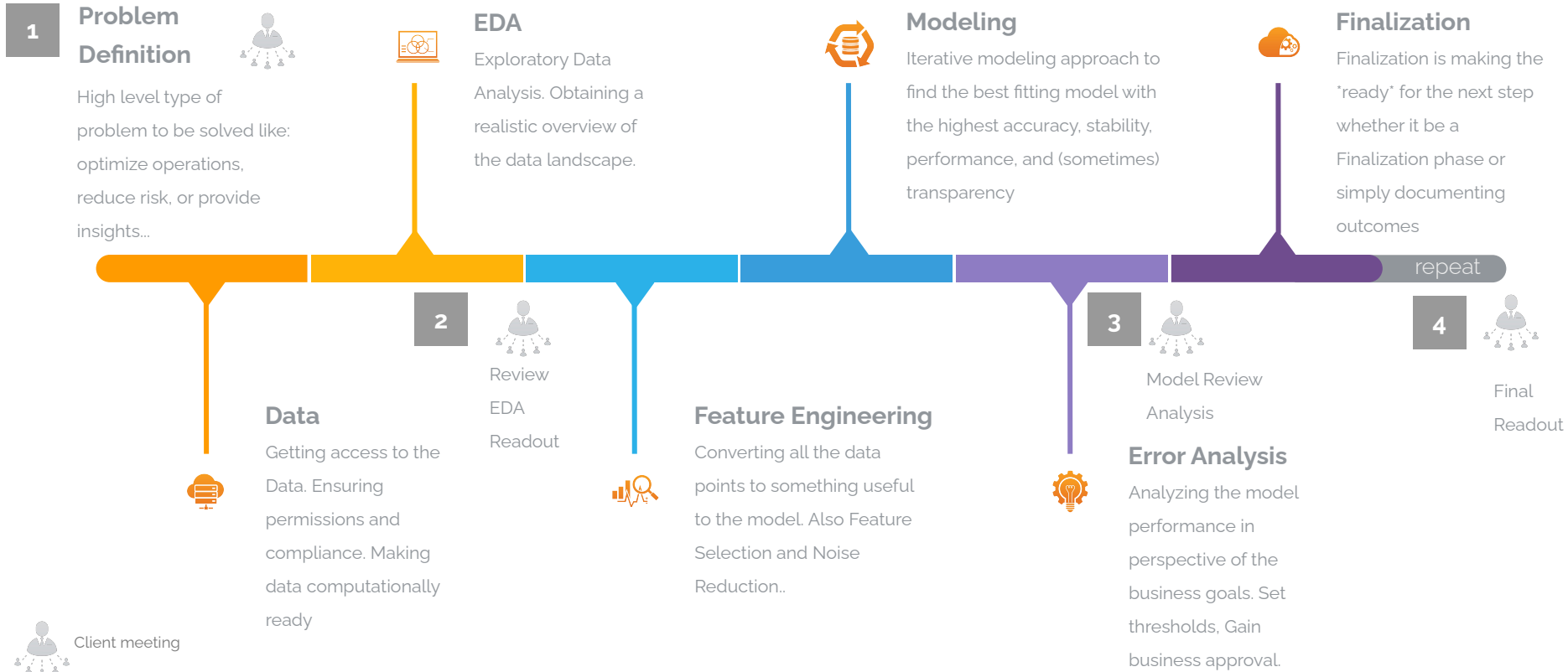
## Hands On Work

Break out into rooms & work on a use case with support

# Today's Schedule

Time (EST)	Activities	Leaders
8:30 AM - 8:50 AM	Introductions	Shelby Cleek, Brian Ray
8:50 AM - 11:30 AM	Work on use case Break @ halfway point	Abadir Yimam, Brian Ray, Anurag Bhatia
11:30 AM - 12:30 PM	Lunch Break	-
12:30 PM - 3:30 PM	Work on second use case Break @ halfway point	Abadir Yimam, Brian Ray, Anurag Bhatia

# Overview Data Science Modeling Process



# Choose two use cases to work through

## Census Income

The census income example is a binary classification problem which has been used here to demonstrate the capabilities of **Vertex AI**, Google Cloud Platform's managed service for managing the entire end-to-end workflow of a machine learning use-case.

The objective of the model itself is to predict whether a said person, given their attributes, earns more than \$50k per annum or not.

## Lending Club

The lending club, an XGBoost model used to demonstrate **Google Cloud Platform's CloudML** hypertune feature which is used to tune the model's hyperparameters. It is a binary classification model which predicts those customers who have a high likelihood of defaulting on loans given a list of their personal and credit report attributes. In this example we explore only three of them, `max_depth`, `num_boost_round`, and `booster`. You can pick any of the parameters; however, and select a range of values to search over to find the optimal combination to maximize a supplied metric.

## Beatles

Build a useful ML Model in hours on GCP to Predict The Beatles' listeners

Solve a real business problem

Host the data in GCP

Use GCP's Cloud **AutoML**

Conduct some EDA "Exploratory Data Analysis"

How the EDA impacts the results

Data Processing on GCP AutoML Tables

Taking advantage of the built in Data Pipeline

Validation of the model using a holdout

Putting the model in production

Improving the model over time

## Business Case Study track

Learn about our business use cases, problems we have solved, worked through and what we have done for other companies.

# Hands On Getting Started Instructions

To get started follow these steps:

**Step 1:** Log onto GCP here <https://console.cloud.google.com/home/dashboard?project=ds-training-380514>

**Step 2:** start a Vertex Workspace with your name on it in the form <name>-<company>

<https://console.cloud.google.com/vertex-ai/workbench/user-managed?project=ds-training-380514>

If one is not present start one with the following settings

Hit "Open JupyterLab"

**Step 3:** Git > Clone a Repository > <https://github.com/data-describe/awesome-data-science-models.git>

To Check out Awesome Data Science Models .

Go to the corresponding directories look for GCP -> EDA as the first step or read the ReadMe

# Assignments

Schedule for working through the use cases

Session	Use Case	Participants	Leader
Morning	Census Income	Mason, Kristin, Dominick	Anurag, Guru, Abadir
	Lending Club		Abadir, Bobby
	Beatles	Daniel, Chintan, Carlton	Jothi, Shubhra, Saurabh, Bobby
	Business Case		Brian, Shelby
Afternoon	Census Income	Daniel, Chintan, Carlton	Anurag, Bobby
	Lending Club	Dominick, Mason	Abadir
	Beatles		Bobby
	Business Case	Kristin	Brian, Shelby

# Topics to be covered and discussed

- Predictions
  - Model Registry
  - Batch Inferences
  - Model Endpoints
  - Online Inferences
- Importance of MLOps
- Model Monitoring
  - Feature skew
  - Drift Detection
  - Email Alerts
- Deployment Pattern
- Intro to Model Retraining
- Best Practices



# MLOps on Vertex AI

- After your models are deployed, they must keep up with changing data from the environment to perform optimally and stay relevant.
- MLOps is a set of practices that improves the stability and reliability of your ML systems.
- Vertex AI MLOps tools help you collaborate across AI teams and improve your models through predictive
  - ***model monitoring***
  - ***alerting***
  - ***Diagnosis***
  - ***actionable explanations.***
- All the tools are modular, so you can integrate them into your existing systems as needed.

## ❖ **Orchestrate workflows:**

Manually training and serving your models can be time-consuming and error-prone, especially if you need to repeat the processes many times.

[Vertex AI Pipelines](#) helps you automate, monitor, and govern your ML workflows.

# MLOps on Vertex AI

## ❖ **Track the metadata used in your ML system:**

In data science, it's important to track the parameters, artifacts, and metrics used in your ML workflow, especially when you repeat the workflow multiple times.

[Vertex ML Metadata](#) lets you record the metadata, parameters, and artifacts that are used in your ML system. You can then query that metadata to help analyze, debug, and audit the performance of your ML system or the artifacts that it produces.

## ❖ **Identify the best model for a use case:**

When you try new training algorithms, you need to know which trained model performs the best.

[Vertex AI Experiments](#) lets you track and analyze different model architectures, hyper-parameters, and training environments to identify the best model for your use case.

[Vertex AI TensorBoard](#) helps you track, visualize, and compare ML experiments to measure how well your models perform.

# MLOps on Vertex AI

## ❖ **Manage model versions:**

Adding models to a central repository helps you keep track of model versions.

[Vertex AI Model Registry](#) provides an overview of your models so you can better organize, track, and train new versions. From Model Registry, you can evaluate models, deploy models to an endpoint, create batch predictions, and view details about specific models and model versions.

## ❖ **Manage features:**

When you re-use ML features across multiple teams, you need a quick and efficient way to share and serve the features.

[Vertex AI Feature Store](#) provides a centralized repository for organizing, storing, and serving ML features. Using a central featurestore enables an organization to re-use ML features at scale and increase the velocity of developing and deploying new ML applications.

# MLOps on Vertex AI

## ❖ Monitor model quality:

A model deployed in production performs best on prediction input data that is similar to the training data. When the input data deviates from the data used to train the model, the model's performance can deteriorate, even if the model itself hasn't changed.

[Vertex AI Model Monitoring](#) monitors models for training-serving skew and prediction drift and sends you alerts when the incoming prediction data skews too far from the training baseline. You can use the alerts and feature distributions to evaluate whether you need to retrain your model.

# Model Monitoring

To help you maintain a model's performance, Model Monitoring monitors the model's prediction input data for feature *skew* and *drift*:

- *Training-serving skew* occurs when the feature data distribution in production deviates from the feature data distribution used to train the model. If the original training data is available, you can enable skew detection to monitor your models for training-serving skew.
- *Prediction drift* occurs when feature data distribution in production changes significantly over time. If the original training data isn't available, you can enable drift detection to monitor the input data for changes over time.

You can enable both **skew and drift detection**.

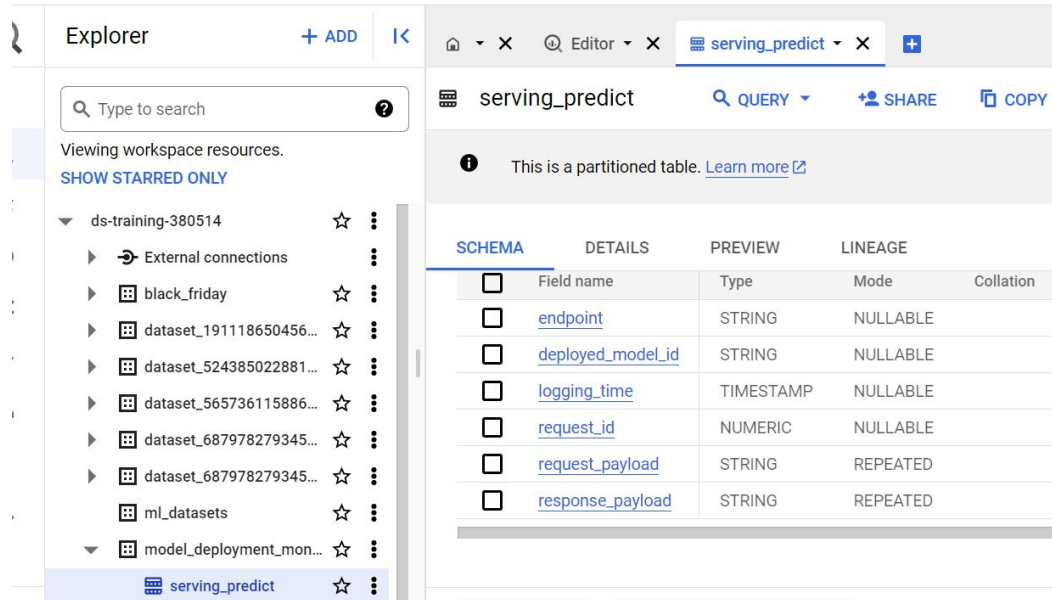
Model Monitoring supports feature skew and drift detection for *categorical* and *numerical* features:

Once the skew or drift for a model's feature exceeds an alerting threshold that you set, Model Monitoring sends you an email alert. You can also view the distributions for each feature over time to evaluate whether you need to retrain your model.

# Model Monitoring

When a model is deployed in production with Model Monitoring enabled, incoming prediction requests are logged in a BigQuery table in your Google Cloud project. The input feature values contained in the logged requests are then analyzed for skew or drift.

You can enable skew detection if you provide the original training dataset for your model; otherwise, you should enable drift detection.



The screenshot displays the Google Cloud Platform console interface. On the left, the 'Explorer' pane shows a list of workspace resources under the project 'ds-training-380514'. The 'serving\_predict' table is highlighted at the bottom. The main pane shows the 'serving\_predict' table details. A message indicates it is a partitioned table. Below this, a tabbed interface shows the 'SCHEMA' view, which lists the table's fields: endpoint, deployed\_model\_id, logging\_time, request\_id, request\_payload, and response\_payload, along with their data types and modes.

Field name	Type	Mode	Collation
<a href="#">endpoint</a>	STRING	NULLABLE	
<a href="#">deployed_model_id</a>	STRING	NULLABLE	
<a href="#">logging_time</a>	TIMESTAMP	NULLABLE	
<a href="#">request_id</a>	NUMERIC	NULLABLE	
<a href="#">request_payload</a>	STRING	REPEATED	
<a href="#">response_payload</a>	STRING	REPEATED	

# Model Monitoring

## Prerequisites

To use Model Monitoring, complete the following:

1. Have an available model in Vertex AI that is either a [tabular AutoML](#) or imported tabular [custom training](#) type.
  - If you are using an existing endpoint, make sure all the models deployed under the endpoint are tabular AutoML or imported custom training types.
2. If you are enabling skew detection, upload your training data to [Cloud Storage](#) or [BigQuery](#) and obtain the URI link to the data. For drift detection, training data is not required.
3. Optional: For custom-trained models, upload the [analysis instance schema](#) for your model to Cloud Storage. Model Monitoring requires the schema to begin the monitoring process and calculate the baseline distribution for skew detection. If you don't provide the schema during job creation, the job remains in a pending state until Model Monitoring can automatically parse the schema from the first 1000 prediction requests the model receives.

# Model Monitoring

## Configure alerts for the Model Monitoring job

For the following events, Model Monitoring sends an email notification to each email address specified when the Model Monitoring job was created:

- Each time skew or drift detection is set up.
- Each time an existing Model Monitoring job configuration is updated.
- Each time a scheduled pipeline fails.



# Sample Email Alert

Vertex AI deployed model monitoring job creation request received.

Inbox x



**Vertex AI** <noreply-vertexai@google.com>

to me ▾

12:27 PM (6 hours ago)



Hello Vertex AI Customer,

You are receiving this mail because you are using the Vertex AI Model Monitoring service.

This mail is to inform you that we received your request to set up drift or skew detection for the Prediction Endpoint listed below. Starting from now, incoming prediction requests will be sampled and logged for analysis.

Raw requests and responses will be collected from prediction service and saved in bq://ds-training-380514.model\_deployment\_monitoring\_1823759936992051200.serving\_predict .

## Basic Information:

Endpoint Name: [projects/354621994428/locations/us-central1/endpoints/1823759936992051200](https://console.cloud.google.com/vertex-ai/locations/us-central1/endpoints/1823759936992051200)

Monitoring Job: [projects/354621994428/locations/us-central1/modelDeploymentMonitoringJobs/3525215010873671680](https://console.cloud.google.com/vertex-ai/locations/us-central1/modelDeploymentMonitoringJobs/3525215010873671680)

Statistics and Anomalies Root Path(Google Cloud Storage): [gs://cloud-ai-platform-d489d350-5fe8-4beb-b383-bf2afb1f888f/model\\_monitoring/job-3525215010873671680](https://cloud-ai-platform-d489d350-5fe8-4beb-b383-bf2afb1f888f/model_monitoring/job-3525215010873671680)

Sincerely,  
The Google Cloud AI Team

# Sample Email Alert

Vertex AI scheduled model monitoring pipeline failed. Inbox x



**Vertex AI** <noreply-vertexai@google.com>

to me ▼

1:55 PM (4 hours ago)



Hello Vertex AI Customer,

You are receiving this mail because you are subscribing to the Vertex AI Model Monitoring service.

This mail is just to inform you that your recent scheduled model monitoring has failed.

## Basic Information:

Endpoint Name: [projects/354621994428/locations/us-central1/endpoints/1823759936992051200](https://console.cloud.google.com/vertex-ai/monitoring/endpoints/projects/354621994428/locations/us-central1/endpoints/1823759936992051200)

Monitoring Job: [projects/354621994428/locations/us-central1/modelDeploymentMonitoringJobs/3525215010873671680](https://console.cloud.google.com/vertex-ai/monitoring/jobs/projects/354621994428/locations/us-central1/modelDeploymentMonitoringJobs/3525215010873671680)

Scheduled time: 2023-04-05 01:00:00

Error message:

Job not found for (354621994428, 3525215010873671680).

Sincerely,

The Google Cloud AI Team

# Model Monitoring

## Configure alerts for feature anomalies

At each monitoring interval, if the threshold of at least one feature exceeds the threshold, Model Monitoring sends an email alert to each email address specified when the Model Monitoring job was created. The email message includes the following:

- The time at which the monitoring job ran.
- The name of the feature that has skew or drift.
- The alerting threshold as well as the recorded statistical distance measure

# Calculate training-serving skew and prediction drift

To detect training-serving skew and prediction drift, Model Monitoring uses [TensorFlow Data Validation \(TFDV\)](#) to calculate the distributions and *distance scores* according to the following process:

1. Calculate the *baseline* statistical distribution:
  - For skew detection, the baseline is the statistical distribution of the feature's values in the training data.
  - For drift detection, the baseline is the statistical distribution of the feature's values seen in production in the recent past.
2. The distributions for categorical and numerical features are calculated as follows:
  - For categorical features, the computed distribution is the number or percentage of instances of each possible value of the feature.
  - For numerical features, Model Monitoring divides the range of possible feature values into equal intervals and computes the number or percentage of feature values that falls in each interval.
3. The baseline is calculated when you [create a Model Monitoring job](#), and is only recalculated if you update the training dataset for the job.

# Calculate training-serving skew and prediction drift

4. Calculate the statistical distribution of the latest feature values seen in production.
5. Compare the distribution of the latest feature values in production against the baseline distribution by calculating a *distance score*:
  - For categorical features, the distance score is calculated using the [L-infinity distance](#).
  - For numerical features, the distance score is calculated using the [Jensen-Shannon divergence](#).
6. When the distance score between two statistical distributions exceeds the threshold you specify, Model Monitoring identifies the anomaly as skew or drift

# Sample Email Alert

Vertex AI <noreply-vertexai@google.com>  
to me ▾

3:06 PM (1 hour ago) ☆ ↶

Hello Vertex AI Customer,

You are receiving this mail because you are subscribing to the Vertex AI Model Monitoring service.  
This mail is just to inform you that there are some anomalies detected in your deployed models and may need your attention.

**Basic Information:**

Endpoint Name: [projects/354621994428/locations/us-central1/endpoints/1823759936992051200](#)  
Monitoring Job: [projects/354621994428/locations/us-central1/modelDeploymentMonitoringJobs/1365176039596097536](#)  
Statistics and Anomalies Root Path(Google Cloud Storage): [gs://cloud-ai-platform-d489d350-5fe8-4beb-b383-bf2afb1f888f/model\\_monitoring/job-1365176039596097536](#)  
BigQuery Command: SELECT \* FROM `bq://ds-training-380514.model\_deployment\_monitoring\_1823759936992051200.serving\_predict`

**Training Prediction Skew Anomalies (Raw Feature):**

Anomalies Report Path(Google Cloud Storage): [gs://cloud-ai-platform-d489d350-5fe8-4beb-b383-bf2afb1f888f/model\\_monitoring/job-1365176039596097536/serving/2023-04-05T04:00:stats\\_and\\_anomalies/7598982145280835584/anomalies/training\\_prediction\\_skew\\_anomalies](#)

For more information about the alert, please visit the [model monitoring alert page](#).

Deployed model id: 7598982145280835584

Feature name	Anomaly short description	Anomaly long description
ABBA	High Linty distance between training and serving	The Linty distance between training and serving is 1 (up to six significant digits), above the threshold 0.05. The feature value with maximum difference is: nan

Sincerely,  
The Google Cloud AI Team

# Update a Model Monitoring job

You can view, update, pause, and delete a Model Monitoring job. You must pause a job before you can delete it

## Considerations when using Model Monitoring

- For cost efficiency, you can set a *prediction request sampling rate* to monitor a subset of the production inputs to a model.
- You can set a frequency at which a deployed model's recently logged inputs are monitored for skew or drift. Monitoring frequency determines the timespan, or monitoring window size, of logged data that is analyzed in each monitoring run.
- You can specify alerting thresholds for each feature you want to monitor. An alert is logged when the statistical distance between the input feature distribution and its corresponding baseline exceeds the specified threshold. By default, every categorical and numerical feature is monitored, with threshold values of 0.3.
- An online prediction endpoint can host multiple models. When you enable skew or drift detection on an endpoint, the following configuration parameters are shared across all models hosted in that endpoint:
  - Type of detection
  - Monitoring frequency
  - Fraction of input requests monitored.

# Deployment Pattern

Deployment or testing pattern	Zero downtime	Real production traffic testing	Releasing to users based on conditions	Rollback duration	Impact on hardware and cloud costs
<b>Recreate</b> Version 1 is terminated, and Version 2 is rolled out.	x	x	x	Fast but disruptive because of downtime	No extra setup required
<b>Rolling update</b> Version 2 is gradually rolled out and replaces Version 1.	✓	x	x	Slow	Can require extra setup for surge upgrades
<b>Blue/green</b> Version 2 is released alongside Version 1; the traffic is switched to Version 2 after it is tested.	✓	x	x	Instant	Need to maintain blue and green environments simultaneously
<b>Canary</b> Version 2 is released to a subset of users, followed by a full rollout.	✓	✓	x	Fast	No extra setup required
<b>A/B</b> Version 2 is released, under specific conditions, to a subset of users.	✓	✓	✓	Fast	No extra setup required
<b>Shadow</b> Version 2 receives real-world traffic without impacting user requests.	✓	✓	x	Does not apply	Need to maintain parallel environments in order to capture and replay user requests



# Useful Links

- [Google Form to chose your use cases](#)
- [Awesome Data Science Models we will be using](#)
- [Census Income](#)
- [Lending Club](#)
- [Beatles](#)
- [Model Monitoring](#)

# Thank you!



Atos, the Atos logo, Atos|Syntel are registered trademarks of the Atos group.  
© 2021 Atos. Confidential information owned by Atos, to be used by the recipient only. This document, or any part of it, may not be reproduced, copied, circulated and/or distributed nor quoted without prior written approval from Atos.