



Vertex AI

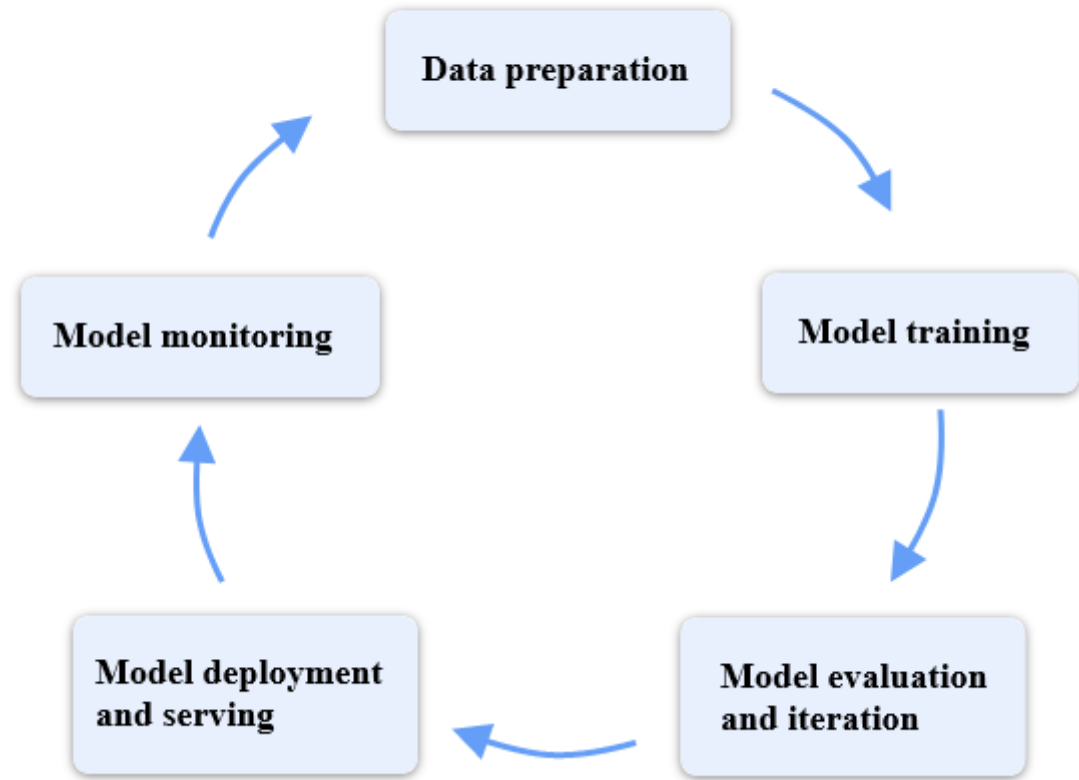
As Data Engineer

# ¿Qué es Vertex AI ?

- Es una plataforma que reúne dos productos de GCP.
  - AutoML
  - AI Platform
- Además nos proporciona herramientas que cubren todo el flujo de trabajo.
  - Notebooks
  - Datasets
    - Datos Tabulados
    - Imágenes/Video
    - Texto
  - Models
    - Mediante AutoML o AI Platform
  - Deploy

¿Qué es Vertex  
AI ?

## Machine learning workflow



# ¿Qué es AutoML?

## ■ AutoML

- Es la capacidad de entrenar un amplio rango de modelos sin preocuparnos por los detalles. Para ello tomará un dataset previamente creado en Vertex AI y realizará múltiples entrenamientos (caja negra) y nos presentará con un modelo ganador.
- Este entrenamiento puede ser limitado a un número de horas (Budget, Maximum node hours) para limitar el costo económico.
- Apenas configurable
  - Seleccionar dataset
  - Seleccionar objetivo
  - Seleccionar peso
  - Seleccionar compute y Early stopping

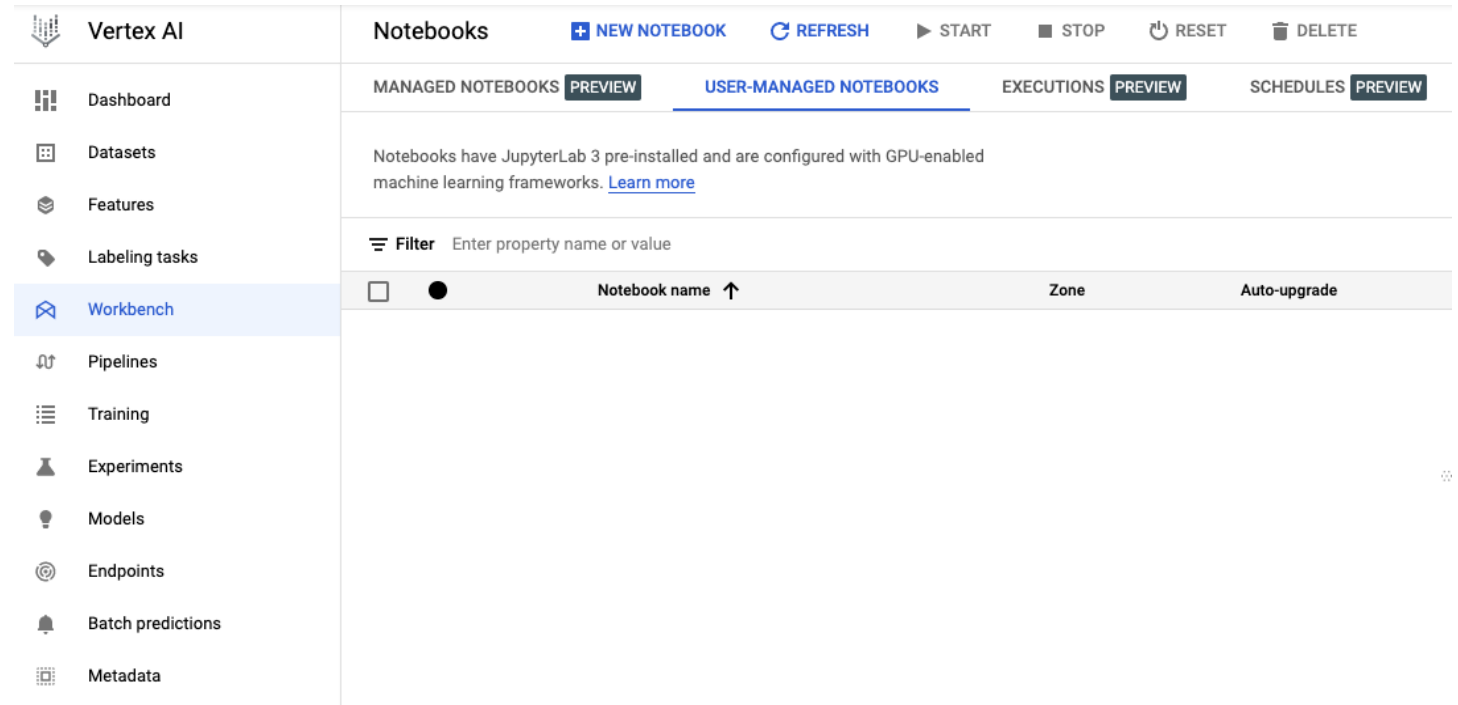
# ¿Qué es Platform AI?

- Platform AI

- Es la capacidad de entrenar un modelo de AI empleando nuestro propio código. Aquí podemos emplear cualquier herramienta que creamos conveniente para resolver la tarea.
- Permite acceder a todo el resto de ventajas que proporciona Vertex AI.
- Es bastante más complicado de emplear.

# Notebooks

- Plataforma de creación de notebooks (Jupyter Notebooks) dentro de Vertex AI. Esto permite ejecutar código Python con todas las librerías relacionadas con Vertex AI ya configuradas sin necesidad de realizar la configuración por nuestra parte.



- Muy similar a Google Colab <https://colab.research.google.com>

# Datasets

- Tenemos cuatro tipos de datasets que podemos crear en Vertex AI
  - Datos Tabulados (csv, excel)
  - Imágenes
  - Video
  - Texto

The screenshot shows the 'Create dataset' page in the Google Cloud Vertex AI console. The page has a left sidebar with navigation icons. The main content area is titled 'Create dataset' and includes a 'Dataset name' field with the value 'untitled\_1619990199973'. Below this is a section 'Select a data type and objective' with tabs for 'IMAGE', 'TABULAR', 'TEXT', and 'VIDEO'. Under the 'IMAGE' tab, there are four options: 'Image classification (Single-label)', 'Image classification (Multi-label)', 'Image object detection', and 'Image segmentation'. The 'Image classification (Single-label)' option is selected. Below these options is a 'Region' dropdown menu set to 'us-central1 (Iowa)'. At the bottom, there is an 'ADVANCED OPTIONS' section with a note about using the dataset for other objectives later. The page ends with 'CREATE' and 'CANCEL' buttons.

← Create dataset

Dataset name \*  
untitled\_1619990199973  
Can use up to 128 characters.

Select a data type and objective  
First select the type of data your dataset will contain. Then select an objective, which is the outcome that you want to achieve with the trained model. [Learn more about model types](#)

IMAGE TABULAR TEXT VIDEO

☒ Image classification (Single-label)  
Predict the one correct label that you want assigned to an image.

☐ Image classification (Multi-label)  
Predict all the correct labels that you want assigned to an image.

☐ Image object detection  
Predict all the locations of objects that you're interested in.

☐ Image segmentation  
Predict per-pixel areas of an image with a label.

Region  
us-central1 (Iowa)

ADVANCED OPTIONS

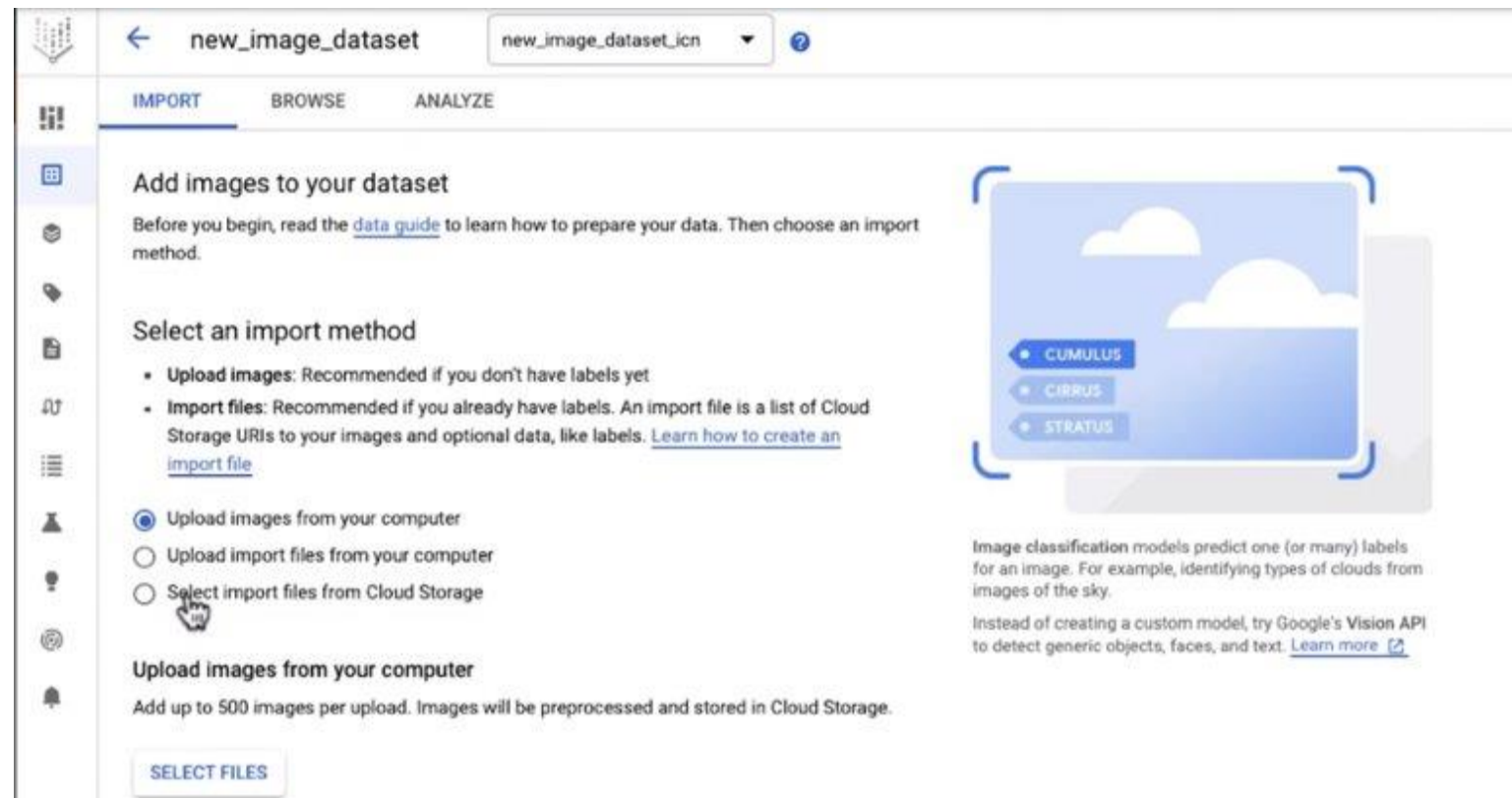
You can use this dataset for other image-based objectives later by creating an annotation set. [Learn more about annotation sets](#)

CREATE CANCEL

Show debug pane

# Datasets

- Tenemos cuatro tipos de datasets que podemos crear en Vertex AI
  - Datos Tabulados (csv, excel)
  - Imágenes
  - Video
  - Texto





## Dataset de datos tabulados

- La primera línea del CSV ha de ser la cabecera, estos son los nombres de las columnas
- Las columnas pueden contener caracteres alfanuméricos y la barra baja "\_" (no puede comenzar por)
- Cada CSV no puede pasar de 10GB, si pesa más de 10GB lo puedes repartir en varios CSV hasta un máximo de 100GB
- El delimitador ha de ser la coma ",".
- Al menos 1000 filas para Tabular Data, 100 imágenes por clase para Vision AI
- El CSV se sube con la variable a predecir incluida

# Dataset de datos tabulados

- No hace falta delimitar el schema del CSV (si las columnas son enteros, flotantes, strings..., etc), Vertex AI lo hace por ti. Se puede repartir los datos entre entrenamiento, validación y test de forma automática o manual
- "John", "Doe", "555-55-5555"
- "Jane", "Doe", "444-44-4444"
- "Roger", "Rogers", "123-45-6789"
- "Sarah", "Smith", "333-33-3333"
- "TRAIN", "John", "Doe", "555-55-5555"
- "TEST", "Jane", "Doe", "444-44-4444"
- "TRAIN", "Roger", "Rogers", "123-45-6789"
- "VALIDATE", "Sarah", "Smith", "333-33-3333"
- 
- "UNASSIGNED", "John", "Doe", "555-55-5555"
- "TEST", "Jane", "Doe", "444-44-4444"
- "UNASSIGNED", "Roger", "Rogers", "123-45-6789"
- "UNASSIGNED", "Sarah", "Smith", "333-33-3333"
-

## Ejercicio datos tabulados

Partiendo del siguiente dataset, genera un CSV compatible con los requerimientos de Vertex AI para subirlo como dataset.

```
from sklearn.datasets import fetch_covtype
import pandas as pd
df = fetch_covtype(return_X_y=False, as_frame=True)['frame']
df = df[df.Cover_Type.isin([5,3]).sample(2000).reset_index(drop=True)]
```

# Podemos subir el CSV resultante en Datasets

## Ejercicio datos tabulados

Dashboard

Datasets

Features

Labelling tasks

Notebooks

Pipelines

Training

Experiments

Models

Endpoints

Batch predictions

Metadata

Data set name \*

forecasting\_demo

Can use up to 128 characters.

Select a data type and objective

First, select the type of data that your dataset will contain. Then, select an objective, which is the outcome

IMAGE

TABULAR

TEXT

VIDEO

Regression/classification

Predict a target column's value. Supports tables with hundreds of columns and millions of rows.

Forecasting **PREVIEW**

Predict the likelihood of certain events or demand.

Region

us-central1 (Iowa)

Vertex AI

ADVANCED OPTIONS

CREATE

CANCEL

Dashboard

Datasets

Features

Labelling tasks

Notebooks

Pipelines

Training

Experiments

Models

Endpoints

Batch predictions

Metadata

← forecasting\_demo

SOURCE

ANALYSE

Add data to your data set

Before you begin, read the [data guide](#) to learn how to prepare your data. Then, choose a data source.

Select a data source

CSV file: Can be uploaded from your computer or on Cloud Storage. [Learn more](#)

BigQuery: Select a table or view from BigQuery. [Learn more](#)

☒ Upload CSV files from your computer

☐ Select CSV files from Cloud Storage

☐ Select a table or view from BigQuery

Upload CSV files from your computer

Add up to 500 CSV files per upload. The files will be stored in a new Cloud Storage bucket ([charges apply](#)). Data from multiple files will be referenced as one data set.

SELECT FILES

# Ejercicio datos tabulados

Una vez importado el CSV, se procesarán los datos durante un tiempo. Finalmente veremos un botón indicando que podemos comenzar un entrenamiento.

The screenshot displays the Vertex AI 'ANALYZE' tab for a dataset named '2Ejercicio'. The interface includes a left-hand navigation menu with options like Dashboard, Datasets, Features, Labeling tasks, Workbench, Pipelines, Training, Experiments, Models, Endpoints, Batch predictions, and Metadata. The main panel shows 'Dataset Info' (Created: Jan 24, 2022 6:54 PM, Format: CSV, Location: gs://cloud-ai-plot\_a0/2ejercicio.csv) and a 'Summary' (Total columns: 55, Total rows: 3,000). Below this is a table of statistics for various features, including Aspect, Cover\_Type, Elevation, Hillshade, and Soil\_Type. A 'Filter' bar is present above the table. On the right, a sidebar shows 'Training jobs and models' with a job '2Ejercicio\_202212417569' in 'Training model...' status and a 'TRAIN NEW MODEL' button.

Column name	Missing % (count)	Distinct values
Aspect	-	358
Cover_Type	-	2
Elevation	-	836
Hillshade_3pm	-	233
Hillshade_9am	-	171
Hillshade_Noon	-	133
Horizontal_Distance_To_Fire_Points	-	850
Horizontal_Distance_To_Hydrology	-	156
Horizontal_Distance_To_Roadways	-	865
Slope	-	49
Soil_Type_0	-	2
Soil_Type_1	-	2
Soil_Type_10	-	2
Soil_Type_11	-	1



Model

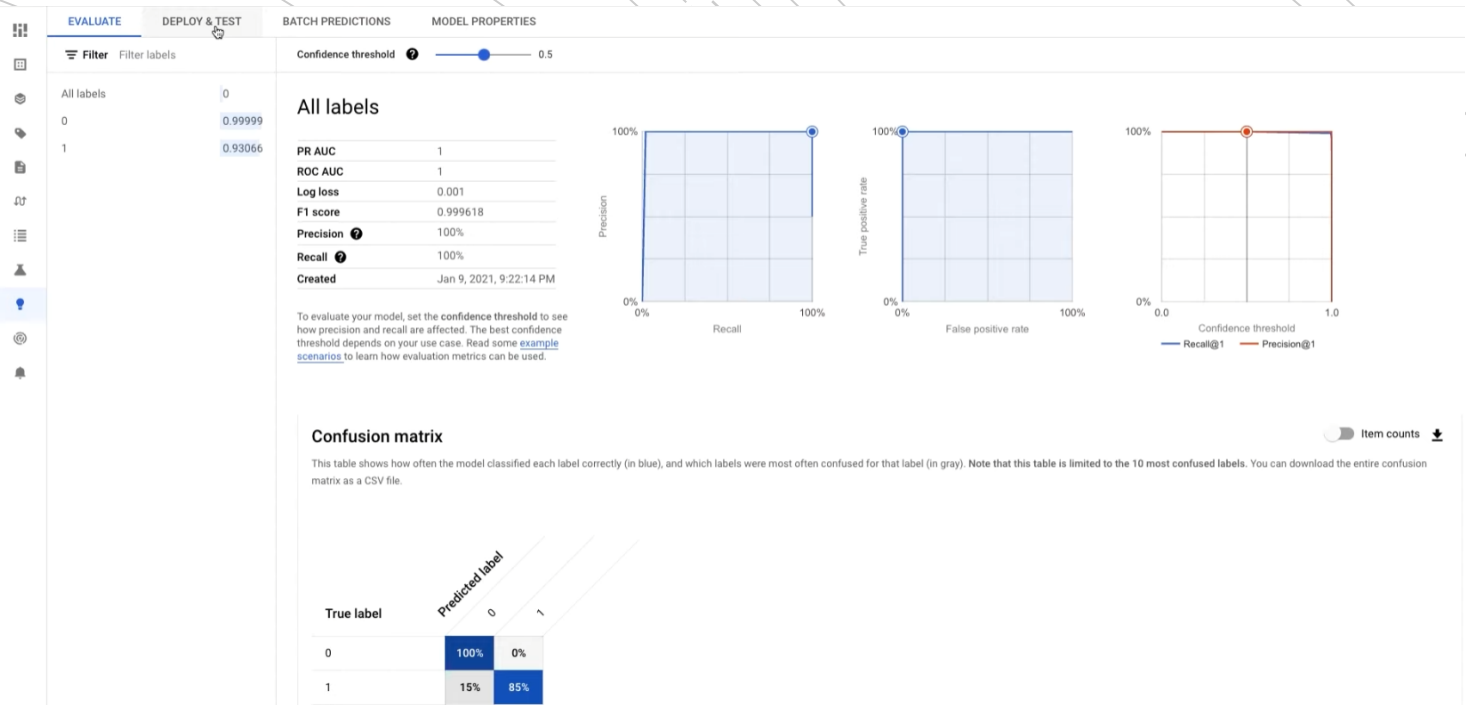
---

Un modelo entrenado (dependiendo del tipo de dataset) reflejará unas métricas.

---

Es muy importante entender las métricas que usa **Vertex AI**, pues no tenemos otra referencia para saber si tenemos un buen modelo o no.

# Model Metrics



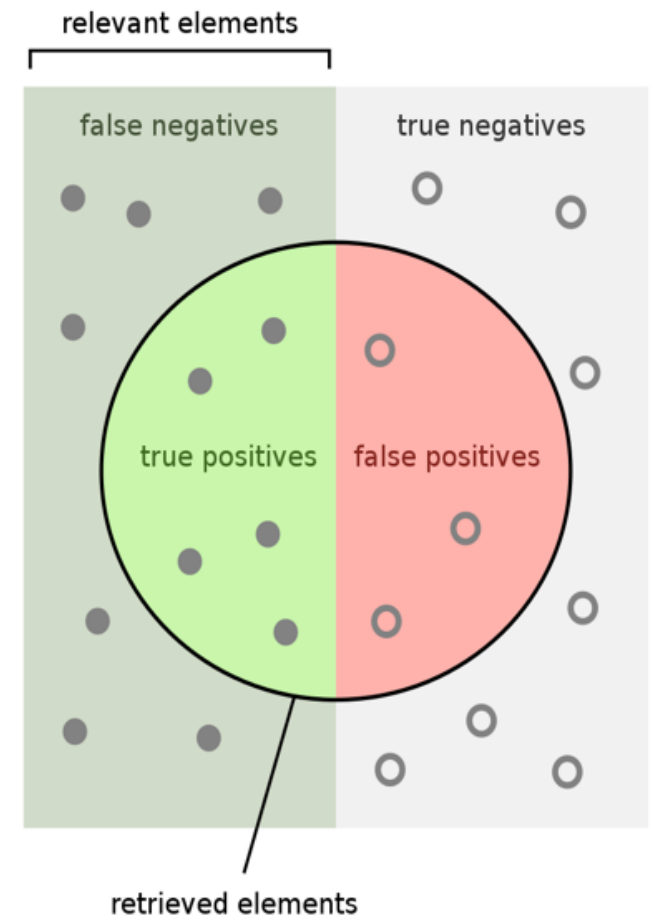
# Model Metrics

- True Positive (TP)
- True Negatives (TN)
- False Positives (FP)
- False Negatives (FN)

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$



How many retrieved items are relevant?

$$\text{Precision} = \frac{\text{Green Circle}}{\text{Green Circle} + \text{Red Circle}}$$

How many relevant items are retrieved?

$$\text{Recall} = \frac{\text{Green Circle}}{\text{Green Circle} + \text{Green Rectangle}}$$



# Model Metrics

- F1 Score

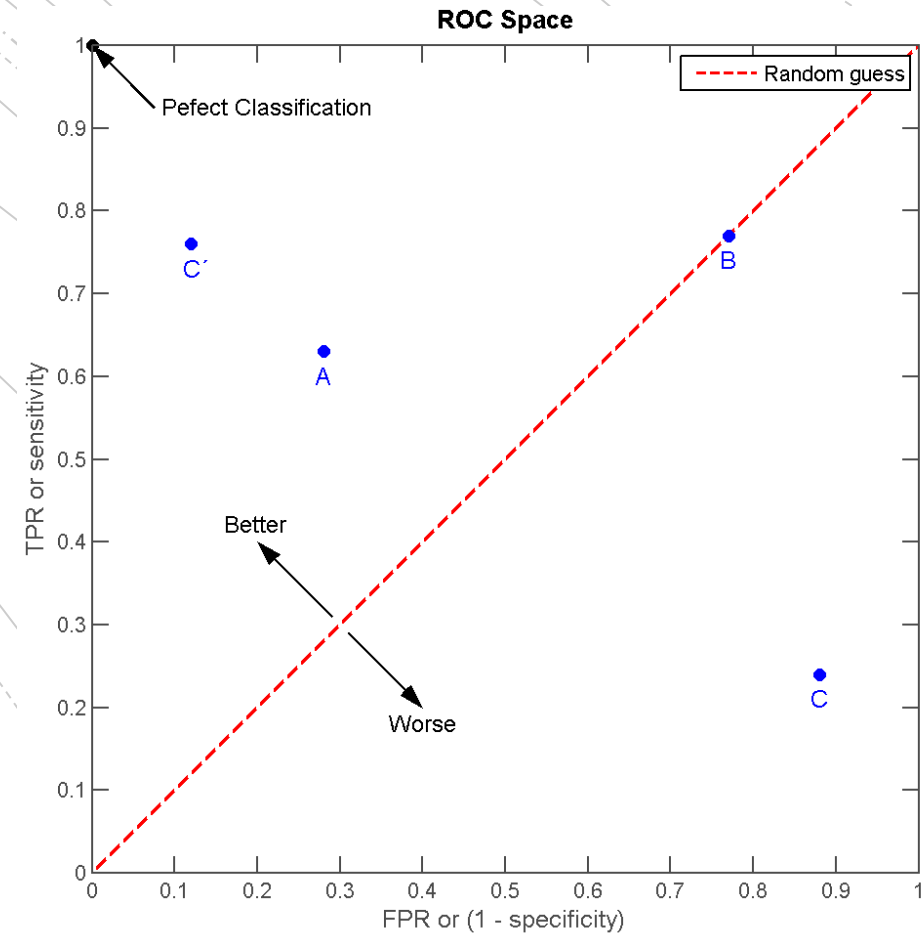
Es la media armónica de la Precision y el Recall

$$F_1 = \frac{2}{\text{recall}^{-1} + \text{precision}^{-1}} = 2 \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2\text{tp}}{2\text{tp} + \text{fp} + \text{fn}}$$

# Model Metrics

## ROC Curve

$$\text{TPR} = \frac{\text{TP}}{\text{P}} = \frac{\text{TP}}{\text{TP} + \text{FN}} = 1 - \text{FNR}$$

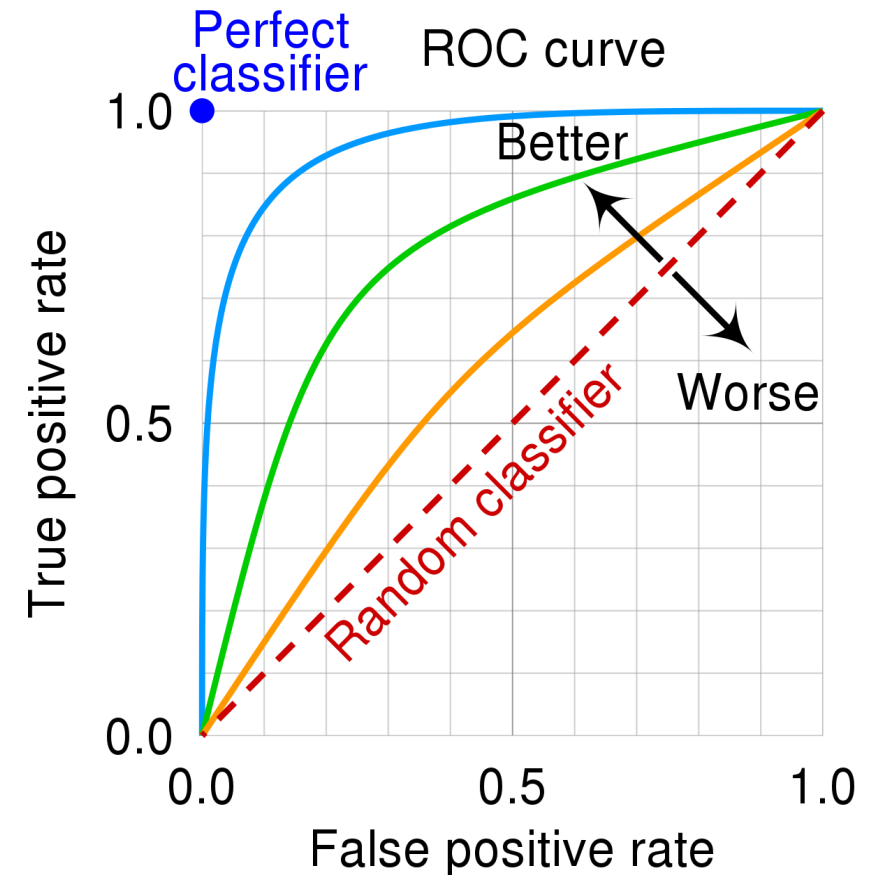


$$\text{FPR} = \frac{\text{FP}}{\text{N}} = \frac{\text{FP}}{\text{FP} + \text{TN}} = 1 - \text{TNR}$$

# Model Metrics

- ROC AUC
- El AUC es **invariable con respecto a la escala**. Mide como de bien se clasifican las predicciones, en lugar de sus valores absolutos.
- El AUC es **invariable con respecto al umbral de clasificación**. Mide la calidad de las predicciones del modelo, independientemente del umbral de clasificación elegido.

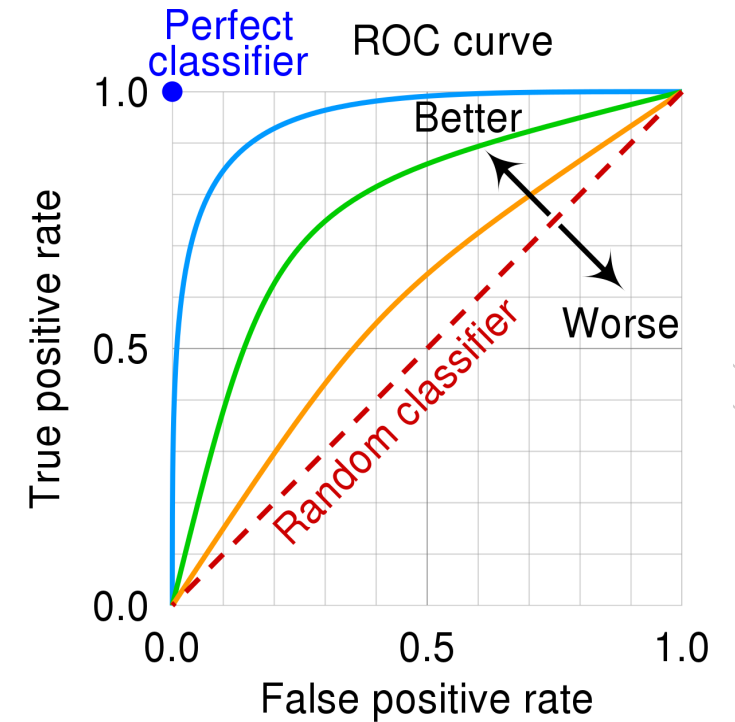
$$\text{TPR} = \frac{\text{TP}}{\text{P}} = \frac{\text{TP}}{\text{TP} + \text{FN}} = 1 - \text{FNR}$$



$$\text{FPR} = \frac{\text{FP}}{\text{N}} = \frac{\text{FP}}{\text{FP} + \text{TN}} = 1 - \text{TNR}$$

# Model Metrics

- ROC AUC
- 1 Clasificador perfecto
- 0.5 Clasificador aleatorio



- ¿Qué significa un clasificador con una ROC AUC  $< 0.5$  ?
- ¿Sería útil un clasificador con una ROC AUC  $= 0$  ?

# Model Metrics

- Matriz de confusión

Se trata de una tabla donde se indica la cantidad de predicciones de una clase frente a la clase verdadera.

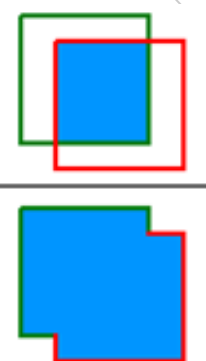
True label	Predicted label	
	0	1
0	100%	0%
1	15%	85%

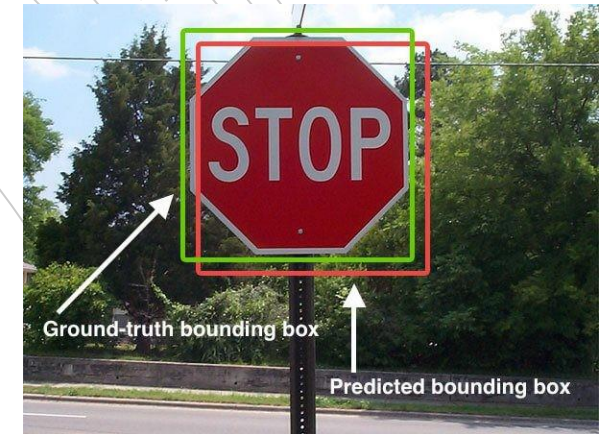
		Valor Predicho		
		Gato	Perro	Conejo
Valor real	Gato	5	3	0
	Perro	2	3	1
	Conejo	0	2	11

# Model Metrics

- IoU Intersection Over Union

Esta es una métrica exclusivamente de visión por computador. Se trata de una métrica de similitud, por lo tanto, cuanto mayor es la cifra mayor similitud. Su valor esta acotado entre 0 y 1.

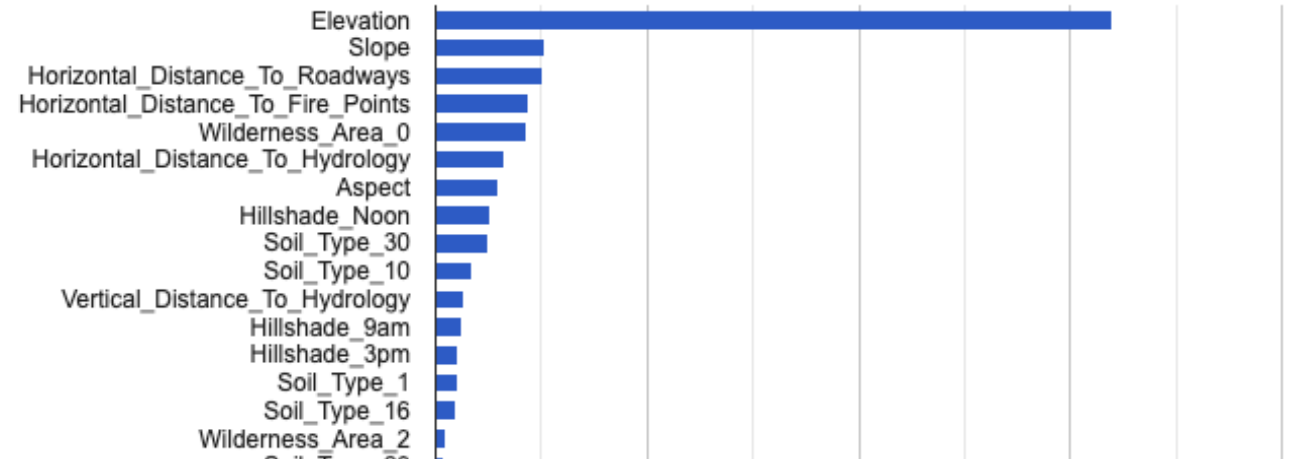
$$IOU = \frac{\text{area of overlap}}{\text{area of union}} =$$




# Model Information

- Feature importance

## Feature Importance



The logo for AutoML is a red speech bubble shape. It consists of a large red rectangle with a smaller red rectangle on top, and a small red triangle pointing downwards from the bottom center. The word "AutoML" is written in white, sans-serif font inside the large red rectangle.

# AutoML

- Como su propio nombre indica, este es posiblemente el producto estrella de Vertex AI, pues nos permite entrenar un modelo simplemente pulsando un botón.
- A continuación, y partiendo de un dataset ya configurado. Vemos la siguiente posibilidad.



# AutoML

Vamos a mostrar como entrenar un modelo con AutoML partiendo de dataset de datos tabulados (CSV)

fraud\_dataset

SOURCE ANALYZE

**Dataset info**

Created: May 02, 2021 3:32 PM  
Dataset format: BigQuery  
Dataset location:  
[bq://bigquery-...ulb\\_fraud\\_detection](#)

**Summary**

Total columns: 31  
Total rows: -

Summary bar chart:

Field Name	Count	Percentage
Amount	30	96.77%
Class	1	3.23%

GENERATE STATISTICS

Filter Enter property name or value

Field Name	BigQuery type	BigQuery mode	Missing % (count)	Distinct values
Amount	Float	Nullable	-	-
Class	Integer	Nullable	-	-
Time	Float	Nullable	-	-
V1	Float	Nullable	-	-
V10	Float	Nullable	-	-
V11	Float	Nullable	-	-

Training jobs and models

Use this dataset and annotation set to learn a model with AutoML or custom

TRAIN NEW MODEL

# AutoML

## Train new model

- 1 Training method
- 2 Model details
- 3 Training options
- 4 Compute and pricing

START TRAINING

CANCEL

Dataset  
fraud\_dataset

Objective \*  
Classification

Please refer to the pricing guide for more details (and available deployment options) for each method.

- ☒ AutoML  
Train high-quality models with minimal effort and machine learning expertise. Just specify how long you want to train. [Learn more](#)
- ☐ Custom training (advanced)  
Run your TensorFlow, scikit-learn, and XGBoost training applications in the cloud. Train with one of Google Cloud's pre-built containers or use your own. [Learn more](#)

CONTINUE

Vemos que la única opción que tenemos que tocar es si en el caso de Objective: Classification o Regression

# AutoML

## Train new model

- ☒ Training method
- 2** ☒ Model details
- 3 Training options
- 4 Compute and pricing

START TRAINING

CANCEL

Model name \*

fraud\_dataset\_cc



Target column \*

Class (INTEGER)



☐ Export test dataset to BigQuery

### Data split

☒ Random assignment

80% of your data is randomly assigned for training, 10% for validation and 10% for testing.

☐ Manual

You assign each data row for training, validation, and testing. [Learn more](#)

☐ Chronological assignment

The earliest 80% of your data is assigned to training, the next 10% for validation and the latest 10% for testing. This option requires a Time column in your dataset. [Learn more](#)

### Encryption

☐ Use a customer-managed encryption key (CMEK)

[^ SHOW LESS](#)

CONTINUE

# AutoML

## Train new model

- ☒ Training method
- ☒ Model details
- ☒ Training options
- ☒ Compute and pricing

START TRAINING

CANCEL

<input type="checkbox"/>	V25	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V26	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V27	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V28	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V3	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V4	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V5	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V6	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V7	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V8	Auto ▾	FLOAT	NULLABLE
<input type="checkbox"/>	V9	Auto ▾	FLOAT	NULLABLE

## Weight column

Select a column ▾

## Optimization objective

- ☐ AUC ROC  
Distinguish between classes
- ☒ Log loss  
Keeps prediction probabilities as accurate as possible
- ☐ AUC PRC  
Maximize precision-recall for the less common class
- ☐ Precision
- ☐ Recall

# AutoML

## Train new model

- ✓ Training method
- ✓ Model details
- ✓ Training options
- 4 Compute and pricing

START TRAINING

CANCEL

Enter the **maximum** number of node hours you want to spend training your model.

You can train for as little as 1 node hour. You may also be eligible to train with free node hours. [Pricing guide](#)

Budget \*

1

Maximum node hours ?

**Estimated completion date:** May 2, 2021 5 PM GMT-7

☒ Enable early stopping

Ends model training when no more improvements can be made and refunds leftover training budget. If early stopping is disabled, training continues until the budget is exhausted.

- Una vez entrenado un modelo, en la sección de models seleccionamos el modelo el cual deseamos crear un endpoint.

# Deploy

**Use your edge-optimized model**

**Container**  
Export your model as a TF Saved Model to run on a Docker container.

**Deploy your model**  
Endpoints are machine learning models made available for online prediction requests. Endpoints are useful for timely predictions from many users (for example, in response to an application request). You can also request batch predictions if you don't need immediate results.

**DEPLOY TO ENDPOINT**

	Name	ID	Models	Region	Last updated	API	Notification	Metadata	Encryption	
⌵	fraud_v2	3534006293532508160	1	us-central1	May 2, 2021, 3:52:17 PM	Sample request			Google-managed key	⋮
✓	fraud_v1	4192182750012243968	1	us-central1	Jan 10, 2021, 7:18:09 PM	Sample request			Google-managed key	⋮

- Una vez entrenado un modelo, en la sección de models seleccionamos el modelo el cual deseamos crear un endpoint.

Deploy

#### Deploy to endpoint

☒ Define your endpoint

**2** Model settings

**3** Model monitoring

DEPLOY

CANCEL

#### Model settings ?

2Ejercicio\_202212417569

Traffic split \*  
100 % ?

#### Compute resources

Choose how compute resources will serve prediction traffic to your model

- **Autoscaling:** If you set a minimum and maximum, compute nodes will scale to meet traffic demand within those boundaries
- **No scaling:** If you only set a minimum, then that number of compute nodes will always run regardless of traffic demand (the maximum will be set to minimum)

Once scaling settings are set, they can't be changed unless you redeploy the model. [Pricing guide](#)

Minimum number of compute nodes \*  
1

Default is 1. If set to 1 or more, then compute resources will continuously run even without traffic demand. This can increase cost but avoid dropped requests due to node initialization.

Maximum number of compute nodes (optional)

Enter a number equal to or greater than the minimum nodes. Can reduce costs but may cause reliability issues for high traffic.

#### ✓ ADVANCED SCALING OPTIONS

Machine type \*  
n1-standard-8, 8 vCPUs, 30 GiB memory ▼ ?

#### Logging

Logging settings are permanent for this endpoint, and Cloud Logging charges will apply. To change your logging preference in the future, create a new endpoint. [Learn more](#)

☒ Enable access logging for this endpoint

☐ Disable container logging for this endpoint



## Deploy

**Traffic Split**, esta variable puede ser entendida en el siguiente escenario.

Cuando tenemos un modelo en un endpoint y queremos realizar una actualización de dicho modelo, si cambiamos el modelo en el endpoint lógicamente va a devolver resultados que pueden variar con el modelo anterior. Para evitar cambios abruptos en los resultados podemos tener dos modelos en un mismo endpoint. Traffic Split nos permite desviar un porcentaje de las peticiones a un modelo e ir incrementado con el paso del tiempo, permitiendo una transición entre modelos mucho más suave, esto puede ser de especial importancia cuando es un cliente el que usa el endpoint.

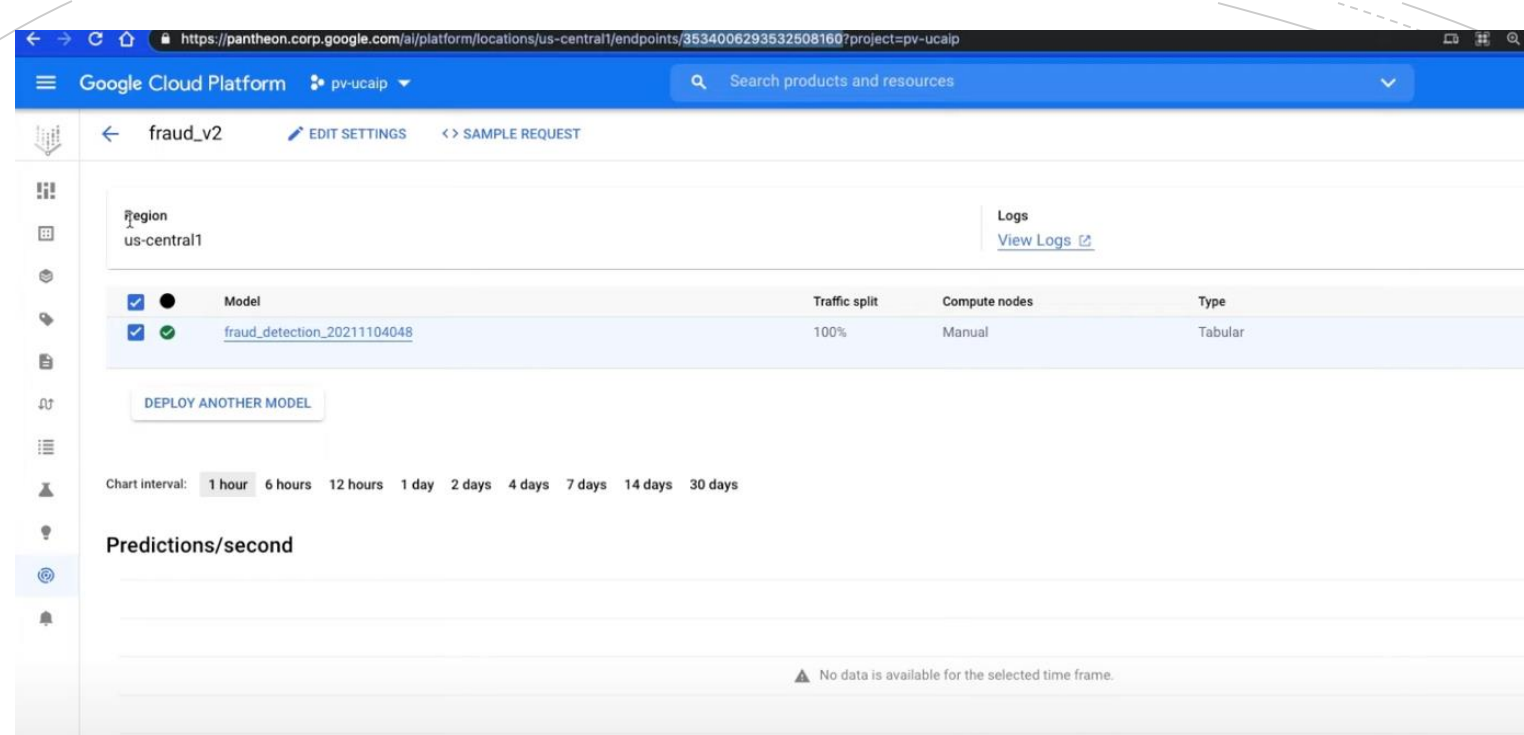
**Minimum de compute nodes** indica como mínimo cuantos modos vamos a tener en funcionamiento en todo momento, con el consecuente gasto económico.

**Maximum compute nodes** indica cuantos nodos como máximo podemos tener, esto es especialmente útil para situaciones donde queremos autoscaling.

Cuando tenemos el endpoint creado, si pulsamos en el veremos la información básica que lo define, si nos fijamos en la URL tendremos un número, como en la siguiente foto.



# Deploy



Una vez desplegado el endpoint, podemos realizar peticiones desde la interfaz o desde código Python y empleando el número que tenemos en la URL.

Recordar que desplegar un endpoint puede tomar desde 15 minutos a horas, el tiempo depende totalmente de GCP.

Para esto tenemos un notebook con el código de ejemplo si deseamos replicarlo.

Avanzado

# Platform AI

Como hemos comentado, Vertex AI se compone de dos piezas a la hora de entrenar modelos.

- AutoML
- Platform AI

AutoML es el que hemos visto hasta ahora, pero también disponemos como herramienta platform AI.

Para estos casos lo mejor es ver cómo construir un ejemplo real y que funcione, pues siempre podemos modificar este caso para adaptarnos a un caso nuestro. Por ello vamos a usar un notebook con el código para realizar un entrenamiento de este tipo.

Recordar como punto clave, que el resultado de este modelo puede ser desplegado como hemos mostrado con anterioridad. Con todas las ventajas de autoscaling que esto conlleva.

Avanzado

# Platform AI

Elementos necesarios para un entrenamiento empleando Platform AI.

- Código que permita el entrenamiento de un modelo.

Disponer de un script de python que al ser ejecutado realice el entrenamiento guardando los resultados.

- Imagen Docker

Para realizar la ejecución es necesario el uso de Docker

Ejemplo al detalle de cómo realizar esto lo podemos ver el notebook **custom model** .

Avanzado

# Platform AI HPO

## Hyper Parameter Optimization (AI Vizier)

- Esta es una de las grandes herramientas que podemos emplear usando Platform AI. Esto nos permite realizar una búsqueda inteligente de los parámetros que pueden componer un entrenamiento.

En el notebook HPTunning podemos ver un ejemplo en detalle.

Avanzado

# Platform AI HPO

Google Cloud Platform

Vertex AI

Dashboard

Datasets

Features

Labelling tasks

Notebooks

Pipelines

Training

Experiments

Models

Endpoints

Batch predictions

Metadata

### Train new model

- 1 Training method
- 2 Model details
- 3 Training container
- 4 Hyperparameters (optional)
- 5 Compute and pricing
- 6 Prediction container (optional)

START TRAINING CANCEL

Dataset \*  
No managed dataset

Annotation set  
-

Objective  
Custom

Please refer to the pricing guide for more details (and available deployment options) for each method.

**1** AutoML options are only available when you train with a managed data set.

☐ AutoML  
Train high quality models with minimal effort and machine learning expertise. Just specify how long you want to train. [Learn more](#)

☐ AutoML Edge  
Train a model that can be exported for on-prem/on-device use. Typically has lower accuracy. [Learn more](#)

☒ Custom training (advanced)  
Run your TensorFlow, scikit-learn and XGBoost training applications in the cloud. Train with one of Google Cloud's pre-built containers or use your own. [Learn more](#)

CONTINUE

Avanzado

# Platform AI HPO

## Train new model

- ✓ Training method
- 2 Model details
- 3 Training container
- 4 Hyperparameters (optional)
- 5 Compute and pricing
- 6 Prediction container (optional)

START TRAINING

CANCEL

Model name \*

✓ ADVANCED OPTIONS

CONTINUE

Avanzado

# Platform AI HPO

### Train new model

- ✓ Training method
- ✓ Model details
- 3 Training container
- 4 Hyperparameters (optional)
- 5 Compute and pricing
- 6 Prediction container (optional)

START TRAINING CANCEL


Select a pre-built container or build a custom container using ML frameworks (as well as non-ML dependencies, libraries and binaries) that are not otherwise supported. [Learn more](#)

☐ Pre-built container  
View the list of [supported runtimes](#) including TensorFlow and scikit-learn versions

☒ Custom container  
Build a custom Docker container. Must be stored in [Container Registry](#)

#### Custom container settings

Container image \* BROWSE

 gs:// Model output directory BROWSE

Your model artifacts and other data needed for training will be stored on Cloud Storage. You should specify a path here if you do not set an output directory in your application code or arguments.

### Select container image

CONTAINER REGISTRY ARTIFACT REGISTRY

Project: pvergadia-demo [CHANGE](#)

- ▶ gcr.io/pvergadia-demo/horse-human
- ▶ gcr.io/pvergadia-demo/mpg
- ▶ gcr.io/pvergadia-demo/tfx-pipeline

SELECT CANCEL

### Train new model

- ✓ Training method
- ✓ Model details
- 3 Training container
- 4 Hyperparameters (optional)
- 5 Compute and pricing
- 6 Prediction container (optional)

START TRAINING CANCEL

Select a pre-built container or build a custom container using ML frameworks (as well as non-ML dependencies, libraries and binaries) that are not otherwise supported. [Learn more](#)


☐ Pre-built container  
View the list of [supported runtimes](#) including TensorFlow and scikit-learn versions

☒ Custom container  
Build a custom Docker container. Must be stored in [Container Registry](#)

#### Custom container settings

Container image \* BROWSE

gcr.io/pvergadia-demo/horse-human@sha256:b8600694462cfda44fb01

 gs:// Model output directory BROWSE

Your model artifacts and other data needed for training will be stored on Cloud Storage. You should specify a path here if you do not set an output directory in your application code or arguments.

#### Arguments

Optional. Add arguments for the command that runs when the container starts. Overrides the container's CMD instruction. Enter one parameter and its argument per line.

--flag\_a=xxxx  
-flag2  
flag3

For parameters you want to tune with HyperTune, enter arguments of the hyperparameters you defined in the training code in the hyperTune setting below. If none, click Next to skip this step.

[CONTINUE](#)

Avanzado

# Platform AI HPO

## Train new model

- ✓ Training method
- ✓ Model details
- ✓ Training container
- 4 Hyperparameters (optional)
- 5 Compute and pricing

START TRAINING

CANCEL

Hyperparameter tuning optimizes your model through multiple trials in one training job, but will increase the cost of this job. After training finishes, the best-performing model will be saved to your Model List. [Learn more](#)

☒ Enable hyperparameter tuning

**i** Ensure that your hyperparameter variables are named and typed correctly.

[VIEW DOCS](#)

### New Hyperparameter

Parameter name \*

learning\_rate

Type \*

Double

Min \*

0.01

Max \*

1

Scaling \*

Log

CANCEL



DONE



Avanzado

# Platform AI HPO

## Train new model

- ✓ Training method
- ✓ Model details
- ✓ Training container
- ✓ Hyperparameters (optional)
- 5 Compute and pricing

START TRAINING CANCEL

Hyperparameter tuning optimizes your model through multiple trials in one training job, but will increase the cost of this job. After training finishes, the best-performing model will be saved to your Model List. [Learn more](#)

✓ Enable hyperparameter tuning

ⓘ Ensure that your hyperparameter variables are named and typed correctly.

[VIEW DOCS](#)

learning\_rate (Double), 0.01 - 1



momentum (Double), 0 - 1



num\_neurons (Discrete), 64,128,512



[ADD NEW PARAMETER](#)

Metric to optimize \*

accuracy

Goal \*

Maximize



Maximum number of trials \*

15

How many training trials should be attempted to optimize the specified hyperparameters. Increasing the number of trials generally yields better results but also increases cost. [Learn more](#)

Maximum number of parallel trials \*

3

The number of training trials to run concurrently. More parallel trials shortens training time but reduces the effectiveness of the tuning.

Algorithm \*

Default



Grid search

Random search

Avanzado

Platform AI  
HPO

### Train new model

- ✓ Training method
- ✓ Model details
- ✓ Training container
- ✓ Hyperparameters (optional)
- 5 Compute and pricing

START TRAINING

CANCEL

Model training pricing is based on the length of time spent training, machine types and any accelerators used. [Learn more](#)

Region

us-central1 (Iowa)



### Compute settings

Select the type of virtual machine to use for your worker pool. You can add up to 4 worker pools. To learn about compute costs and how to map your ML framework's roles to specific worker pools, consult the [documentation](#)

#### Worker pool 0

Machine type \*

n1-standard-4, 4 vCPUs, 15 GiB memory

Accelerator type

NVIDIA\_TESLA\_T4

Accelerators can speed up model training that involves intensive compute tasks. [Learn more](#)

Accelerator count

Worker count

1

Disk type

SSD

Disk size (GB)

100

Avanzado

# Platform AI Edge



**Avanzado**

# Platform AI Edge

Verte AI permite entrenar en la nube un modelo para ser ejecutado en inferencia en un dispositivo edge.

Notebook Edge tenemos más información

Avanzado

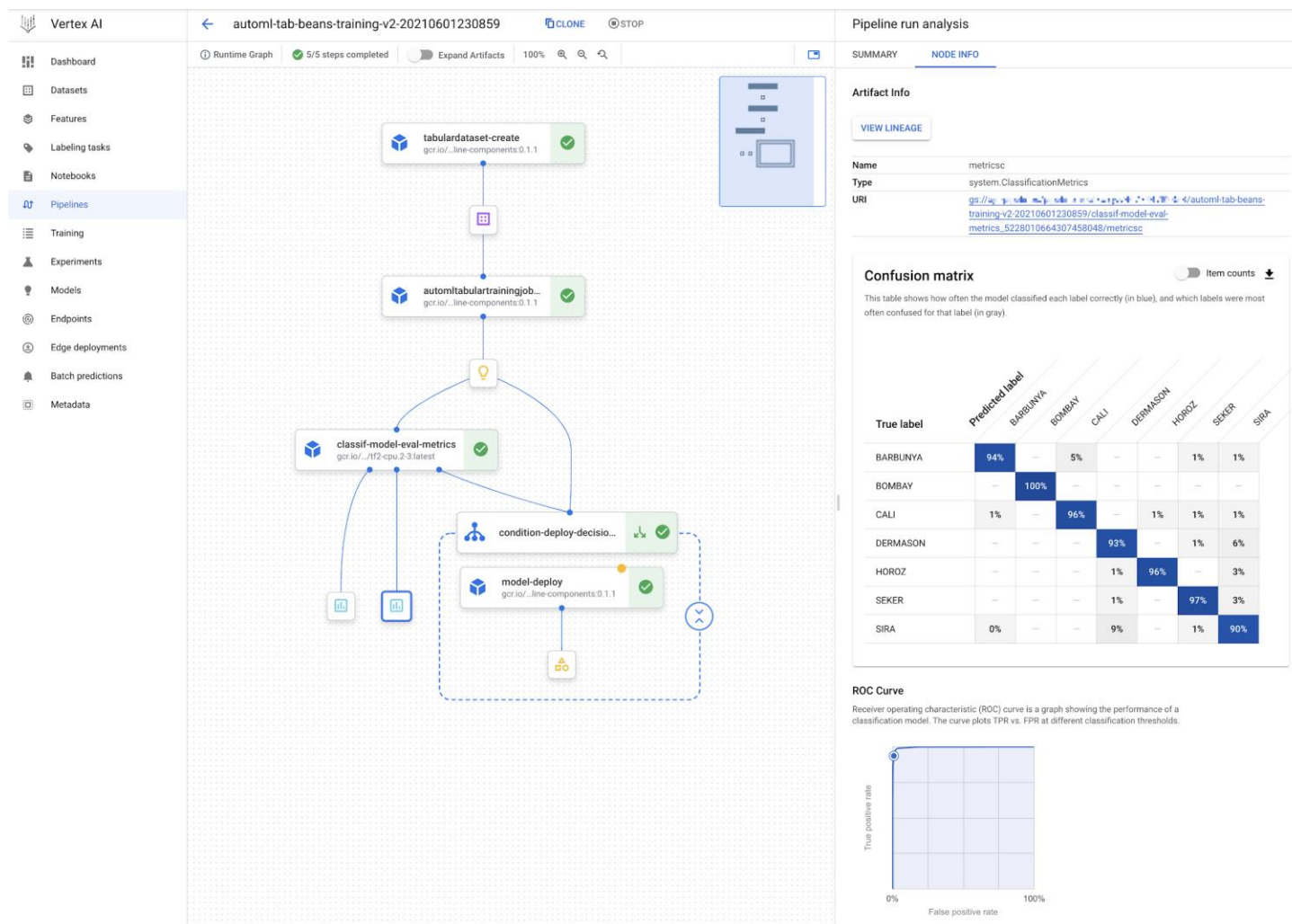
# Platform AI Pipelines

Vertex Pipelines permite automatizar, monitorear y controlar la orquestación de los flujos de trabajo.

- Registra la metainformación de los artifacts, métricas, descendencia (de donde viene un modelo desplegado)
- Realiza una representación gráfica mediante un DAG del flujo de trabajo.

Avanzado

# Platform AI Pipelines



## Otras ventajas

- **Feature Store**
  - Repositorio donde tenemos almacenados entidades o atributos (features) que puedan ser reutilizados para otros modelos.
- **Model Registry**
  - Proporciona una descripción general de tus modelos para que puedas organizar, hacer un seguimiento y entrenar mejor las versiones nuevas. Desde el registro de modelos, puedes evaluar modelos, implementarlos en un extremo, crear predicciones por lotes y ver detalles sobre los modelos y las versiones de modelos específicos.
- **Model Monitoring**
  - Supervisa los modelos para detectar sesgos de entrega y entrenamiento, y te envía alertas cuando los datos de predicción entrantes se inclinan demasiado lejos del modelo de referencia de entrenamiento.

# Vertex AI Resumen

- Con Vertex AI podemos:
  - Entrenar modelos muy sencillamente simplemente preparando un dataset.
  - Podemos desplegarlos para usarlos mediante una API ya sea mediante Python o similares.
  - Podemos realizar entrenamientos más avanzados y complejos con búsqueda de parámetros.
  - Podemos tener un control de flujo para reentrenar modelos.