

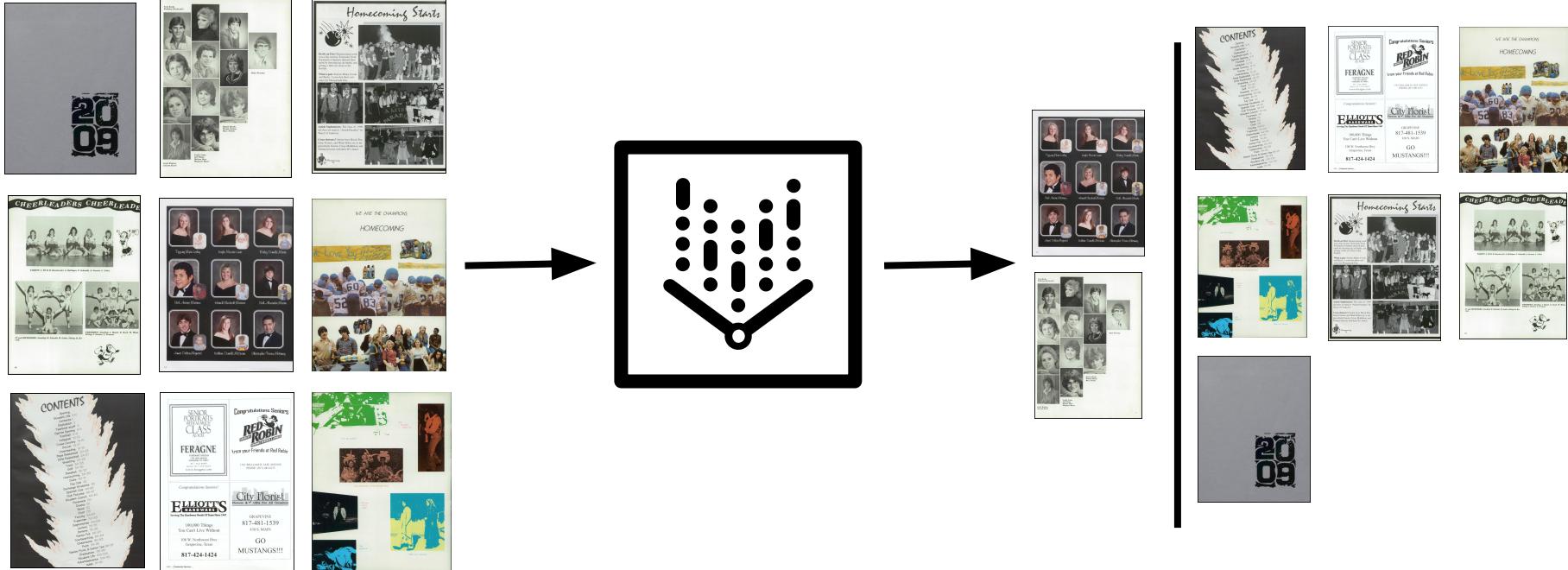


# Google Vertex AI

Introduction to image classification

# IMAGES - senior page classifier

Goal: spot senior pages in high school yearbooks



# Create your AutoML Vertex model

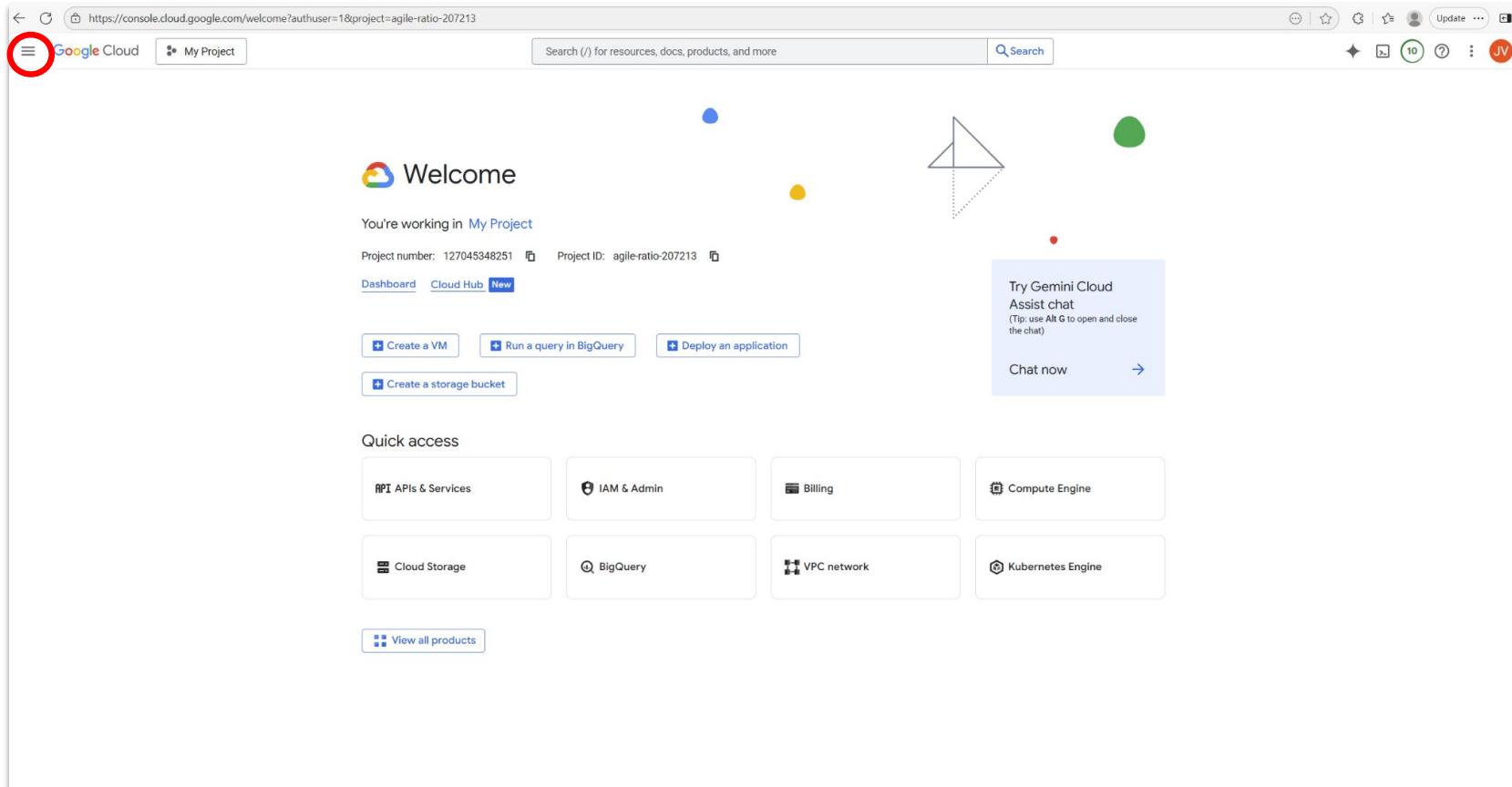
## 1. Training your model

- 1.1. Create a bucket in Cloud
- 1.2. Create a dataset with labels
- 1.3. Train your model

## 2. Deploying your model

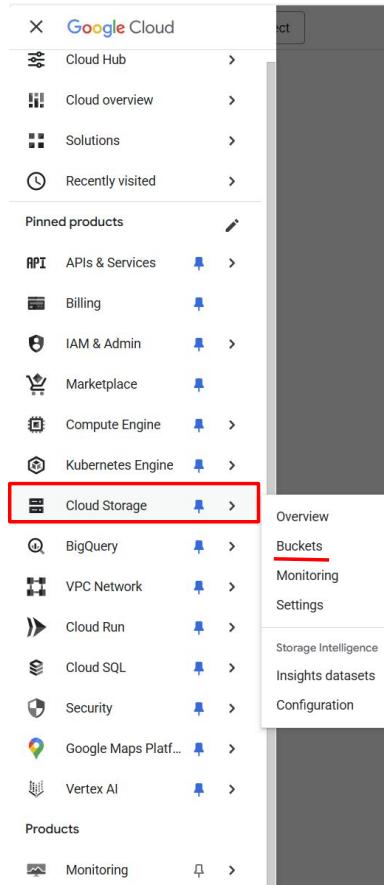
- 2.1. Export your model
- 2.2. Download it from the bucket
- 2.3. Example run

# 1. Train an AutoML Vertex model



The screenshot shows the Google Cloud Welcome page. At the top left, there is a red circle highlighting the "Google Cloud" logo. The page features a search bar at the top right and a navigation bar with "My Project" selected. The main area has a "Welcome" section with a "Cloud" icon, a message about working in "My Project", and project details like "Project number: 127045348251" and "Project ID: agile-ratio-207213". Below this are buttons for "Dashboard", "Cloud Hub", and "New". A sidebar on the right offers options like "Create a VM", "Run a query in BigQuery", "Deploy an application", and "Create a storage bucket". A "Try Gemini Cloud Assist chat" box is also present. The bottom section, titled "Quick access", contains cards for various services: API APIs & Services, IAM & Admin, Billing, Compute Engine, Cloud Storage, BigQuery, VPC network, and Kubernetes Engine. A "View all products" button is located at the bottom left of this section.

# 1.1. Create a bucket in Google Cloud Storage



**Bucket:** top-level container in Google Cloud Storage where your files (objects) are stored.

→ This is where you will store your data, labels, and models.

A screenshot of the Google Cloud Storage Buckets list page. At the top, there's a search bar and a 'Create' button which is circled in red. Below that, there's a section for 'Your pinned buckets' with three items: 'imagesresubmission...', 'seniors\_pages\_classi...', and 'export\_model\_seniors'. Then there's a 'Filter buckets' section and a table listing several buckets. The table columns include Name, Created, Location type, Location, Default storage class, Last modified, Public access, Access control, Protection, and Hierarchical namespace. The buckets listed are: 'm\_portraits\_models' (Created Oct 15, 2025), 'm-portrait-vs-other-models' (Created Oct 9, 2025), 'm-portraits-vs-other' (Created Oct 8, 2025), and 'babies\_model' (Created Oct 6, 2025).

Leave the default settings when creating your bucket.

# 1.2. Create a dataset with labels

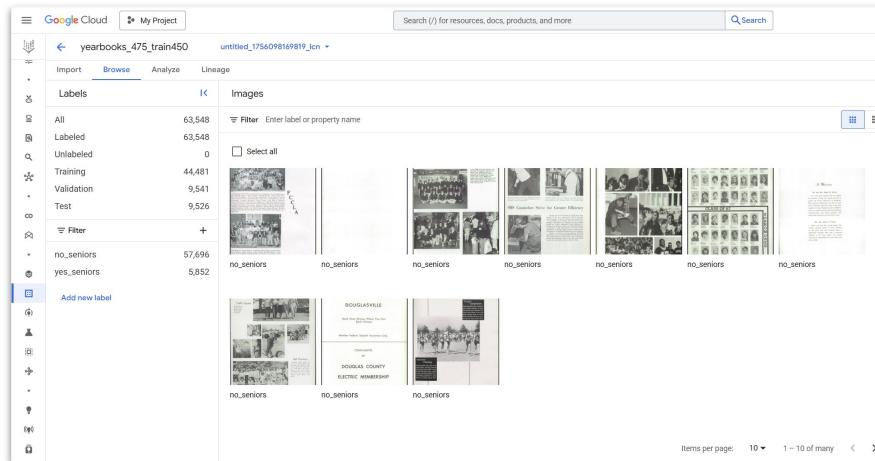
The screenshot shows the Google Cloud Platform navigation menu. The 'Vertex AI' option under the 'Products' section is highlighted with a red border. Other visible options include Cloud Hub, Model Garden, Cloud overview, Solutions, Recently visited, Pinned products, APIs & Services, Billing, IAM & Admin, Marketplace, Compute Engine, Kubernetes Engine, Cloud Storage, BigQuery, VPC Network, Cloud Run, Cloud SQL, Security, Google Maps Platf..., and Monitoring.

**Dataset:** managed data objects used for training and evaluating models.

This screenshot shows the 'Datasets' page in the Google Cloud Platform. It lists several datasets under the 'us-central1 (Iowa)' region. The columns include Name, ID, Status, Region, Type, Items, Last updated, and Labels. The datasets listed are: 'm\_portraits\_vs\_other' (Status: Ready, 860 images), 'babies\_resample' (Status: Ready, 1,069 images), 'babies' (Status: Ready, 32,832 images), 'men\_4\_labels' (Status: Ready, 110,944 images), 'men\_2\_labels' (Status: Ready, 110,944 images), 'women\_4\_labels' (Status: Ready, 112,107 images), 'women\_2\_labels' (Status: Ready, 112,107 images), and 'hairgroundtruth' (Status: Ready, 2,031 images). A 'Create' button is visible at the top right.

Name	ID	Status	Region	Type	Items	Last updated	Labels
m_portraits_vs_other	4856696357804941312	Ready	us-central1	Image	860 images	October 9, 2025	
babies_resample	8118480032011048448	Ready	us-central1	Image	1,069 images	October 8, 2025	
babies	6507484832295486440	Ready	us-central1	Image	32,832 images	October 3, 2025	
men_4_labels	817791642153730648	Ready	us-central1	Image	110,944 images	September 23, 2025	
men_2_labels	1433791201641228	Ready	us-central1	Image	110,944 images	September 23, 2025	
women_4_labels	106194747928092288	Ready	us-central1	Image	112,107 images	September 23, 2025	
women_2_labels	665796767791187648	Ready	us-central1	Image	112,107 images	September 23, 2025	
hairgroundtruth	4572360335792537600	Ready	us-central1	Image	2,031 images	August 29, 2025	

This is what a dataset looks like:



# 1.2. Create a dataset with labels

Google Cloud My Project Search

yearbooks\_475\_train450 untitled\_1756098169819\_icn

Import Browse Analyze Lineage

Add images to your dataset

Before you begin, review the data guide to make sure your data is formatted correctly and optimized for the best results. Supported image file formats: JPEG, PNG, GIF, BMP, TIFF, WebP, ICO

[View data guide](#)

Select an import method

- Upload images: Recommended if you don't have labels yet
- Import files: Recommended if you already have labels. An import file is a list of Cloud Storage URLs to your images and optional data, like labels. [Learn how to create an import file](#)

Upload images from your computer 1

Upload import files from your computer

Select import files from Cloud Storage 2

Upload images from your computer

Add up to 500 images per upload. Images will be preprocessed and stored in Cloud Storage.

[Select Files](#)

Summary

- CUMULUS
- CIRRUS
- STRATUS

Image classification for an image. For example, images of the sky.

Instead of creating a dataset to detect generic objects, we can create a dataset to detect specific objects.

## Option 1 (small dataset):

Upload images from your computer after pointing at the bucket you have created, then label directly from the Dataset “Browse” section. (For single-label classification, we recommend uploading by batches with the similar label.)

## Option 2 (large dataset):

Uploading can take some time, for larger dataset we recommend uploading in the bucket from your terminal (using Google Cloud CLI and gsutil). You should import the data with an import file (that can include the labels).

## 1.2. Create a dataset with labels

The screenshot shows the Google Cloud Storage interface for a project named "My Project". The path "yearbooks\_475\_train450" is selected. A modal window titled "Assign Labels" is open over the list of images. The modal has tabs for "Assign Labels" (which is active) and "Assign ML Use". It shows 10 selected images and includes a "Select all" checkbox. The left sidebar lists categories: All (63,548), Labeled (63,548), and Unlabeled (0). The "Unlabeled" category is circled in red. Below the categories, there are filters for "no\_seniors" (57,696) and "yes\_seniors" (5,852). At the bottom of the sidebar, there is a button "Add new label".

Option 1: Every time you are done uploading a batch of files from a category, go in the unlabeled section, select all and assign a label.

# 1.3. Train your model

Related resources >

Training jobs and models

Use this dataset and annotation set to train a new machine learning model with AutoML or custom code

yearbooks\_475\_train450 (Version 1)  
Model type: Image classification  
Train new model

1. Training method:
  - AutoML
  - Edge (for exporting)
2. Model details
  - Advanced options: default settings
3. Explainability: optional
4. Training options: Higher accuracy
5. Compute and pricing
  - Budget: ~10
  - Enable early stopping

Name	ID	Status	Job type	Model type	Duration	Last updated	Created	Ended	Labels
mporates-vs-other-850	288794475438316656	Finished	Training pipeline	Image classification (Single-label)	1 hr 3 min	Oct 15, 2023, 6:37:49PM	Oct 15, 2023, 6:37:49PM	—	—
mporates-vs-other-750	3935096199763608	Finished	Training pipeline	Image classification (Single-label)	1 hr 3 min	Oct 10, 2023, 4:36:06PM	Oct 10, 2023, 4:36:06PM	—	—
mporates-vs-other	553387053946470400	Finished	Training pipeline	Image classification (Single-label)	48 min 36 sec	Oct 6, 2023, 3:37:18PM	Oct 6, 2023, 2:48:41PM	Oct 6, 2023, 3:37:18PM	—
mporates-vs-other	6765879659474191336	Finished	Training pipeline	Image classification (Single-label)	1 hr 3 min	Oct 6, 2023, 12:15:54AM	Oct 6, 2023, 11:16:46AM	Oct 6, 2023, 12:15:54AM	—
babies_resample	244321218686779292	Finished	Training pipeline	Image classification (Single-label)	1 hr 3 min	Oct 6, 2023, 4:12:30PM	Oct 6, 2023, 3:08:49PM	Oct 6, 2023, 4:12:30PM	—
babies-20231007-090249	47508728897037248	Finished	Training pipeline	Image classification (Single-label)	2 hr 28 min	Oct 7, 2023, 11:31:49AM	Oct 7, 2023, 9:05:50AM	Oct 7, 2023, 11:31:49AM	—
babies	540597525690087038	Finished	Training pipeline	Image classification (Single-label)	2 hr 3 min	Oct 6, 2023, 6:31:22PM	Oct 6, 2023, 4:27:25PM	Oct 6, 2023, 6:31:22PM	—
men_2_labels	4524207299714820090	Finished	Training pipeline	Image classification (Single-label)	4 hr 40 min	Sep 24, 2023, 2:33:30AM	Sep 23, 2023, 2:33:30AM	Sep 24, 2023, 2:33:30AM	—

You can then check how the training is going in the “Training” section:

# 2.1. Export your model

You can find your model in “Model Registry” with statistics on performance.

The screenshot shows the Google Cloud Model Registry interface for a model named "senior\_pages\_classifier\_revised". The "Evaluation" tab is selected, displaying various metrics and plots. A red circle highlights the "Export" button at the top right of the evaluation details section. Another red circle highlights the "Help" icon in the bottom left corner of the page.

Google Cloud My Project

senior\_pages\_classifier\_revised > Version 1 View dataset Export

Evaluate Deploy & test Batch Infer Version details Lineage

untitled\_58--6059542528 Compare Create evaluation

Labels Evaluation details

Confidence threshold 0.5

All labels

Label	Value
All labels	0.997
no_seniors	0.998
yes_seniors	0.982

If you update any labels, train a new model version to get an updated evaluation.

All labels

Metric	Value
Average precision	0.997
Precision	99.1%
Recall	99.3%
Created	Feb 21, 2025, 10:12:23 PM
Total images	14,030
Training images	11,278
Validation images	1,359
Test images	1,393

To evaluate your model, set the confidence threshold to see how precision and recall are affected. The best confidence threshold depends on your use case. Read some [example scenarios](#) to learn how evaluation metrics can be used.

Precision-recall curve

Precision-recall by threshold

Export: TF Lite and Tensorflow.js in your bucket

Export model

Export to Cloud Storage

The TensorFlow Lite (.tflite) format allows you to run your model on mobile and embedded devices.

1. Export your model as a TF Lite package.

Destination folder on Cloud Storage \*  
gs://yearbooks\_475\_train450

2. Model export takes a couple of minutes. After exporting is finished, copy the package to your computer using the following command:

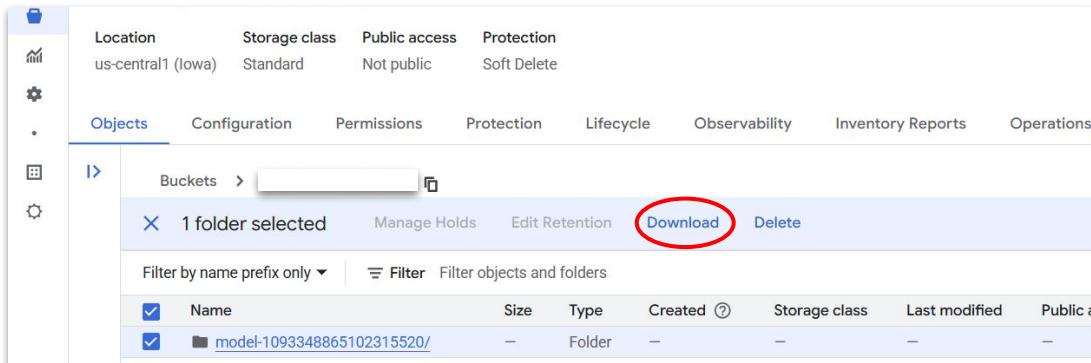
```
$ gsutil cp -r gs://yearbooks_475_train450 ./download_dir
```

TF Lite: your model

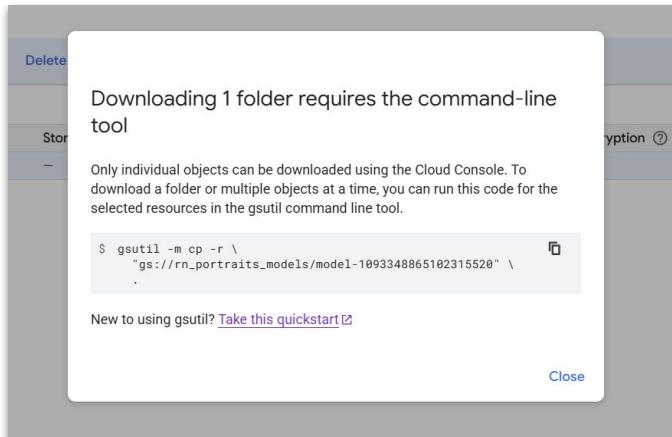
Tensorflow.js: your model dictionary (important to interpret correctly the model output)

## 2.2. Download your model from your bucket

Go back to the bucket and download



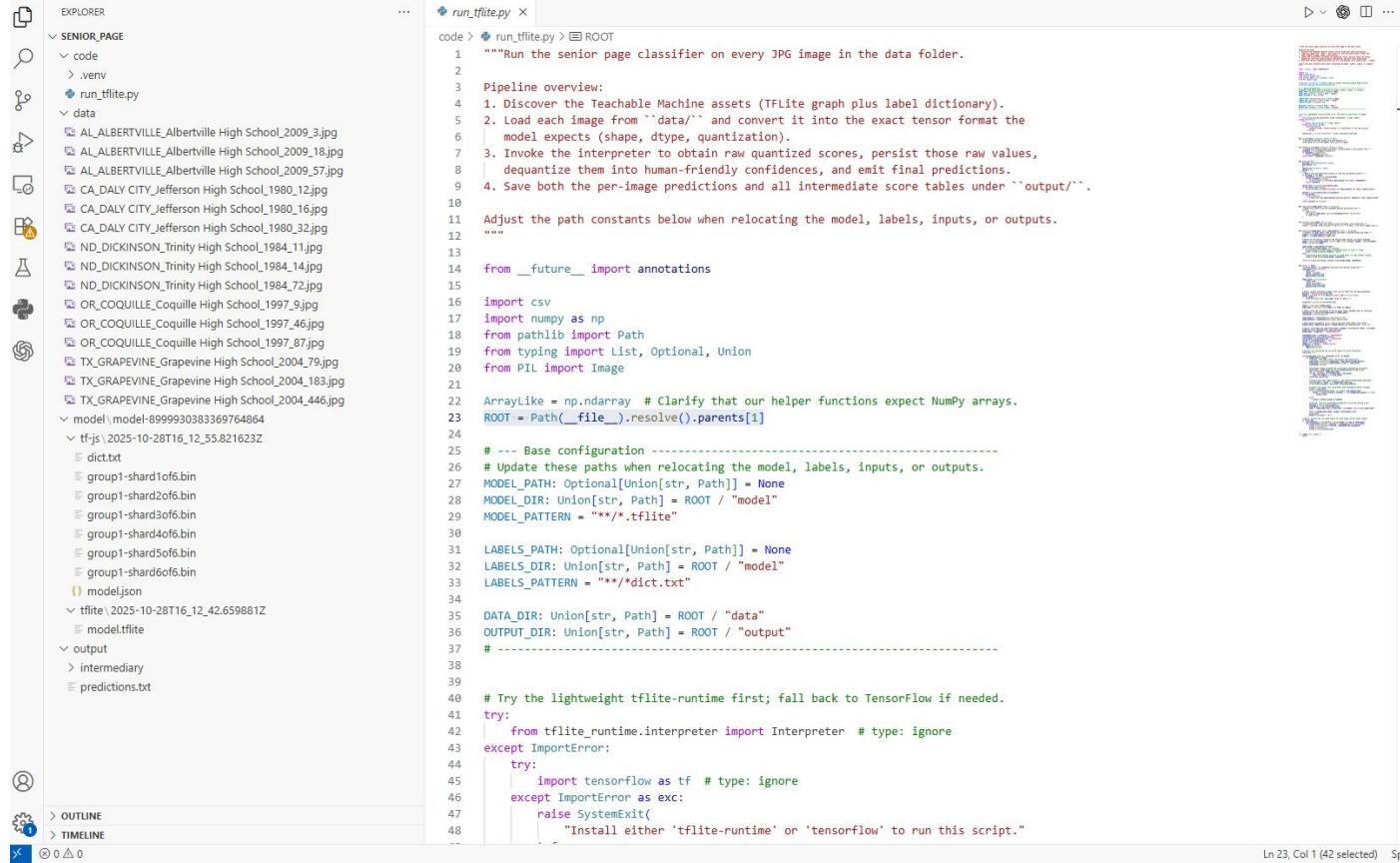
The screenshot shows the Google Cloud Storage console interface. On the left is a sidebar with icons for buckets, storage, and operations. The main area has tabs for Objects, Configuration, Permissions, Protection, Lifecycle, Observability, Inventory Reports, and Operations. Under the Objects tab, there's a breadcrumb navigation showing 'Buckets > [selected bucket]'. Below this, a message says '1 folder selected' with buttons for 'Manage Holds', 'Edit Retention', 'Download' (which is circled in red), and 'Delete'. There are filters for 'Filter by name prefix only' and 'Filter objects and folders'. A table lists the selected folder: 'model-1093348865102315520/' under 'Name', 'Folder' under 'Type', and '-' for size, created, last modified, and storage class. The table also includes columns for 'Size', 'Type', 'Created', 'Storage class', 'Last modified', and 'Public access'.



The modal dialog box contains the following text:  
"Deleting 1 folder requires the command-line tool  
Only individual objects can be downloaded using the Cloud Console. To download a folder or multiple objects at a time, you can run this code for the selected resources in the gsutil command line tool.  
  
\$ gsutil -m cp -r \  
"gs://rn\_portraits\_models/model-1093348865102315520" \  
.  
  
New to using gsutil? [Take this quickstart](#) 

Copypaste this command, run it in the Cloud Console (Cloud Shell Editor), then download the model folder from the explorer.

## 2.3. Example run



The image shows a code editor interface with two windows. The left window is titled 'EXPLORER' and shows a file tree for a project named 'SENIOR\_PAGE'. The right window is titled 'run\_tfLite.py' and contains the Python script code.

```
code > run_tfLite.py > ROOT
1 """Run the senior page classifier on every JPG image in the data folder.
2
3 Pipeline overview:
4 1. Discover the Teachable Machine assets (TFLite graph plus label dictionary).
5 2. Load each image from ``data/`` and convert it into the exact tensor format the
6 | model expects (shape, dtype, quantization).
7 3. Invoke the interpreter to obtain raw quantized scores, persist those raw values,
8 | dequantize them into human-friendly confidences, and emit final predictions.
9 4. Save both the per-image predictions and all intermediate score tables under ``output``.
10
11 Adjust the path constants below when relocating the model, labels, inputs, or outputs.
12 """
13
14 from __future__ import annotations
15
16 import csv
17 import numpy as np
18 from pathlib import Path
19 from typing import List, Optional, Union
20 from PIL import Image
21
22 ArrayLike = np.ndarray # Clarify that our helper functions expect NumPy arrays.
23 ROOT = Path(__file__).resolve().parents[1]
24
25 # --- Base configuration -----
26 # Update these paths when relocating the model, labels, inputs, or outputs.
27 MODEL_PATH: Optional[Union[str, Path]] = None
28 MODEL_DIR: Union[str, Path] = ROOT / "model"
29 MODEL_PATTERN = "**/*.tflite"
30
31 LABELS_PATH: Optional[Union[str, Path]] = None
32 LABELS_DIR: Union[str, Path] = ROOT / "model"
33 LABELS_PATTERN = "**/*dict.txt"
34
35 DATA_DIR: Union[str, Path] = ROOT / "data"
36 OUTPUT_DIR: Union[str, Path] = ROOT / "output"
37 #
38
39 # Try the lightweight tflite-runtime first; fall back to Tensorflow if needed.
40 try:
41     from tflite_runtime.interpreter import Interpreter # type: ignore
42 except ImportError:
43     try:
44         import tensorflow as tf # type: ignore
45     except ImportError as exc:
46         raise SystemExit(
47             f"Install either 'tflite-runtime' or 'tensorflow' to run this script."
48         )
```

The terminal window on the right shows the command being run:

```
Ln 23, Col 1 (42 selected) Spā
```

## 2.3. Example run



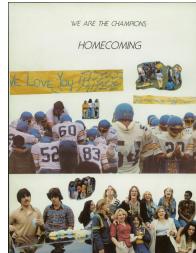
no\_seniors (0.949)



yes\_seniors (0.992)



no\_seniors (0.945)



no\_seniors (0.949)



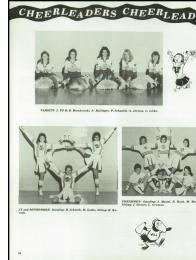
no\_seniors (0.918)



yes\_seniors (0.988)



no\_seniors (0.949)



no\_seniors (0.949)



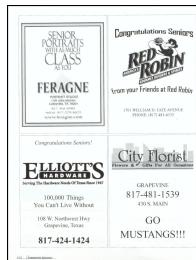
no\_seniors (0.953)



no\_seniors (0.945)



no\_seniors (0.992)



no\_seniors (0.953)

# AutoML Tips

A few rules to respect with your training sample:

- 1) Include enough labeled examples of each category (aim for 1,000, minimum 100)
- 2) Distribute examples equally across categories (even if the true distribution is unbalanced)
- 3) Capture the variation in your problem space (examples should capture the diversity of cases for each category)

## Resource

[AutoML beginner's guide](#) by Google