Task 1. Set up AutoML

AutoML provides an interface for all the steps in training an image classification model and generating predictions on it. Start by enabling the Cloud AutoML API.

1. From the **Navigation menu**, select **APIs & Services** > **Library**.

2. In the search bar type in "Cloud AutoML".

3. Observe the **Cloud AutoML API** is in the **Enable** state.

4. In a new browser, open the AutoML UI.

Create storage bucket

1. Now create a storage bucket by running the following:

```
gsutil mb -p $GOOGLE_CLOUD_PROJECT \
    -c standard   \
    -l us-central1 \
    gs://$GOOGLE_CLOUD_PROJECT-vcm/
```

2. In the Google Cloud console, open the **Navigation menu** and click on **Cloud Storage** to see it.

Click **Check my progress** to verify the objective.

Create a Cloud Storage Bucket

Check my progress

Assessment Completed!

Task 2. Upload training images to Cloud Storage

In order to train a model to classify images of clouds, you need to provide labelled training data so the model can develop an understanding of the image features associated with different types of clouds. In this example your model will learn to classify three different types of clouds: cirrus, cumulus, and cumulonimbus. To use AutoML you need to put your training images in Cloud Storage.

1. Before adding the cloud images, create an environment variable with the name of your bucket.

Run the following command in Cloud Shell:

```
export BUCKET=$GOOGLE_CLOUD_PROJECT-vcm
```

The training images are publicly available in a Cloud Storage bucket.

2. Use the `gsutil` command line utility for Cloud Storage to copy the training images into your bucket:

```
gsutil -m cp -r gs://spls/gsp223/images/* gs://${BUCKET}
```

3. When the images finish copying, click the **Refresh** button at the top of the Storage browser, then click on your bucket name. You should see 3 folders of photos for each of the 3 different cloud types to be classified.

If you click on the individual image files in each folder you can see the photos you'll be using to train your model for each type of cloud.

Task 3. Create a dataset

Now that your training data is in Cloud Storage, you need a way for AutoML to access it. You'll create a CSV file where each row contains a URL to a training image and the associated label for that image. This CSV file has been created for you; you just need to update it with your bucket name.

1. Run the following command to copy the file to your Cloud Shell instance:

```
gsutil cp gs://spls/gsp223/data.csv .
```

2. Then update the CSV with the files in your project:

```
sed -i -e "s/placeholder/${BUCKET}/g" ./data.csv
```

3. Now upload this file to your Cloud Storage bucket:

```
gsutil cp ./data.csv gs://${BUCKET}
```
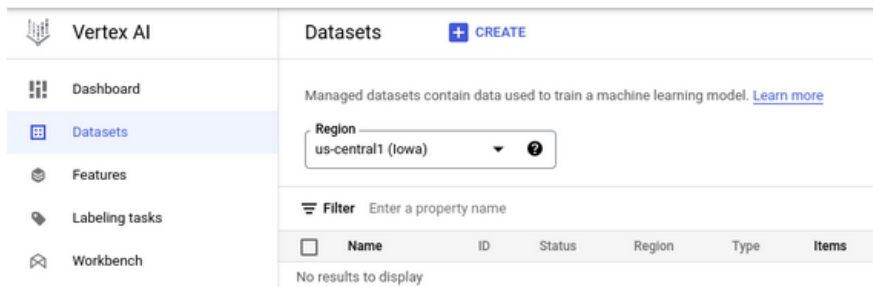
4. Once that command completes, click the **Refresh** button at the top of the Storage browser. Confirm that you see the `data.csv` file in your bucket.

5. Open the Vertex AI Dataset tab. Your page should now resemble the following:

5. Open the Vertex AI Dataset tab. Your page should now resemble the following:

6. At the top of the console, click **+ CREATE**.

7. Type "clouds" for the Dataset name.

8. Select **Single-Label Classification**.

> **Note:** In your own projects, you may want to use multi-class classification.

9. Click **CREATE**.

10. Choose **Select import files on Cloud Storage** and add the file name to the URL for the file you just uploaded - `your-bucket-name/data.csv`

An easy way to get this link is to go back to the Cloud Console, click on the `data.csv` file and then go to the URI field.
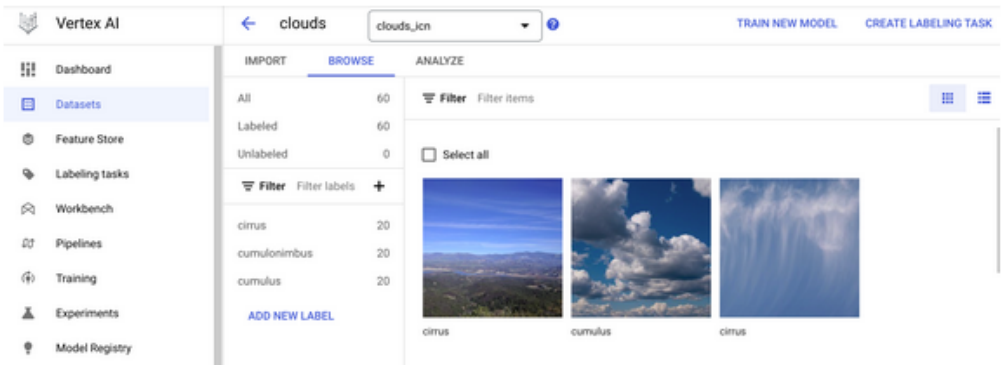
11. Click **CONTINUE**.

It will take 2 - 5 minutes for your images to import. Once the import has completed, you'll be brought to a page with all the images in your dataset.

Click **Check my progress** to verify the objective.

Create a Dataset
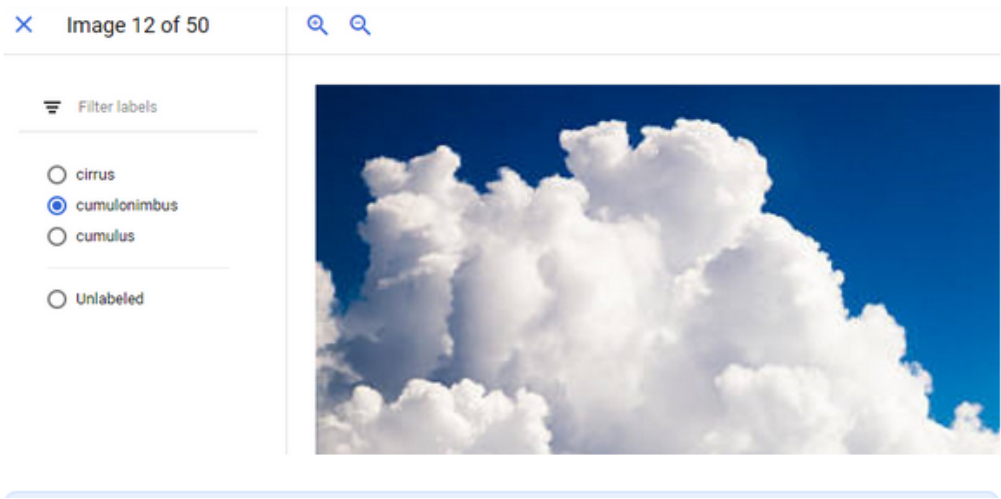
Task 4. Inspect images

After the import completes, you will be redirected to **Browse** tab to see the images you uploaded.

Try filtering by different labels in the left menu (i.e. click cumulus) to review the training images:

> **Note:** If you were building a production model, you'd want *at least* 100 images per label to ensure high accuracy. This is just a demo so only 20 images were used so the model could train quickly.

If any images are labeled incorrectly you can click on the image to switch the label:

Task 5. Train your model

You're ready to start training your model! AutoML handles this for you automatically, without requiring you to write any of the model code.

1. To train your clouds model, click **TRAIN NEW MODEL**.

2. On the **Training method** tab, click **Continue**.

3. On the **Model details** tab, click **Continue**.

4. On the **Explainability** tab, click **Continue**.

5. On the **Compute and pricing** tab, set the node hours to **8**.

6. Click **Start Training**.

Since this is a small dataset, it will only take around **25-30 minutes** to complete. In the meantime, proceed to the next section to use a pre-trained model.

Task 6. Generate predictions

There are a few ways to generate predictions. In this lab, you'll use the UI to upload images. You'll see how your model does classifying these two images (the first is a cirrus cloud, the second is a cumulonimbus).

1. Return to the **Cloudshell** terminal.

2. Download these images to your local machine.

```
gsutil cp gs://spls/gsp223/examples/* .
```

Task 5. Train your model

You're ready to start training your model! AutoML handles this for you automatically, without requiring you to write any of the model code.

1. To train your clouds model, click **TRAIN NEW MODEL**.

2. On the **Training method** tab, click **Continue**.

3. On the **Model details** tab, click **Continue**.

4. On the **Explainability** tab, click **Continue**.

5. On the **Compute and pricing** tab, set the node hours to **8**.

6. Click **Start Training**.

Since this is a small dataset, it will only take around **25-30 minutes** to complete. In the meantime, proceed to the next section to use a pre-trained model.

Task 6. Generate predictions

There are a few ways to generate predictions. In this lab, you'll use the UI to upload images. You'll see how your model does classifying these two images (the first is a cirrus cloud, the second is a cumulonimbus).

1. Return to the **Cloudshell** terminal.

2. Download these images to your local machine.

```
gsutil cp gs://spls/gsp223/examples/* .
```

4. Copy the Endpoint value from the Qwiklabs Panel to an environment variable.

```
ENDPOINT=$(gcloud run services describe automl-service --platform n
```

5. Enter the following command to request a prediction:

```
curl -X POST -H "Content-Type: application/json" $ENDPOINT/v1 -d "
```

The above call will ask AutoML for a prediction. However there is no input data specified, so the request will fail. The 400 HTTP error code indicates the expected data is not present.

Expected Output:

```
{
  "error": {
    "code": 400,
    "message": "Empty instances.",
    "status": "INVALID_ARGUMENT"
  }
}
```

Pop Quiz

Test your understanding of AutoML by completing the short quiz on the topics covered in this lab. Use the knowledge you have gained in the lab to generate predictions.

Pop Quiz

Test your understanding of AutoML by completing the short quiz on the topics covered in this lab. Use the knowledge you have gained in the lab to generate predictions.

What type of cloud is in the above image?

✓ cirrus

✕ ~~cumulus~~

○ cumulonimbus

Submit

Lets check if our model can predict the type of Cloud in the image:

1. Set `CLOUD1-JSON` as the input file.

```
INPUT_DATA_FILE=CLOUD1-JSON
```

2. Enter the following command to request a prediction:

```
curl -X POST -H "Content-Type: application/json" $ENDPOINT/v1 -d "@
```


What type of cloud is in the above image?

✓ cumulonimbus

```
                Submit
```

Lets check if our model can predict the type of Cloud in the image:

3. Set `CLOUD2-JSON` as the input file.

```
INPUT_DATA_FILE=CLOUD2-JSON
```

4. Enter the following command to request a prediction:

```
curl -X POST -H "Content-Type: application/json" $ENDPOINT/v1 -d "@
```

Congratulations!

You've learned how to train your own custom machine learning model and generate predictions on it through the web UI. Now you've got what it takes to train a model on your own image dataset.

What you did:

- Uploaded training images to Cloud Storage and created a CSV for AutoML to find these images.

- Reviewed labels and trained a model in the AutoML UI.

- Generated predictions on new cloud images.

Finish your quest