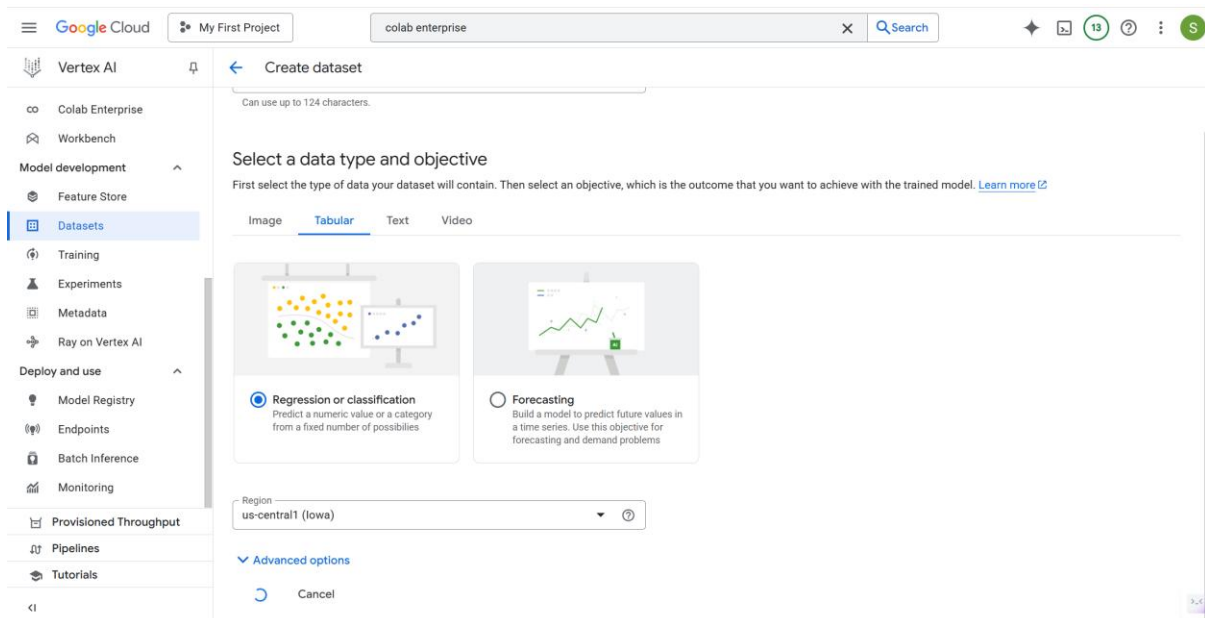


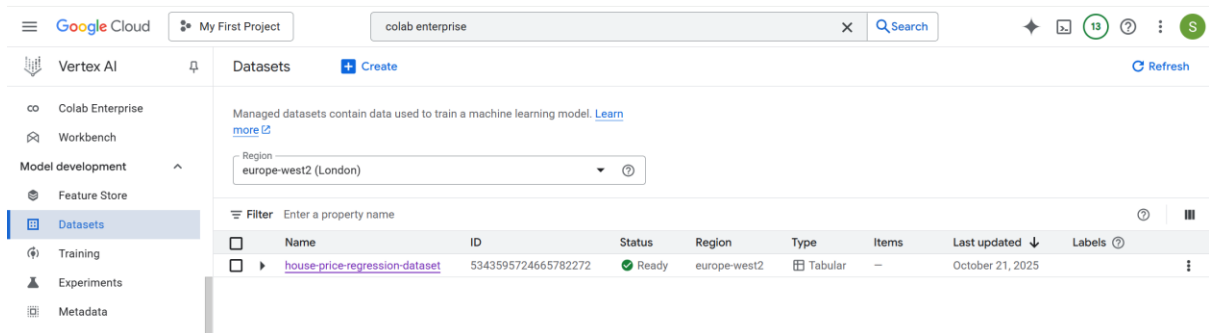
Tools Used: Vertex AI (AutoML Regression), Cloud Storage, Cloud Shell, JSON API calls.

1. **Data Prep:** Uploaded CSV dataset into GCS bucket.
2. **Model Training:** Used Vertex AI AutoML Tabular (regression mode).
 - AutoML tested multiple algorithms (linear, boosted trees, DNNs).
 - Automated hyperparameter tuning & evaluation.
3. **Model Registry:** Best model stored in Vertex Model Registry.
4. **Deployment:** Deployed model as an endpoint in Vertex AI.
5. **Prediction:** Created a sample input.json and fetched predictions using curl command request

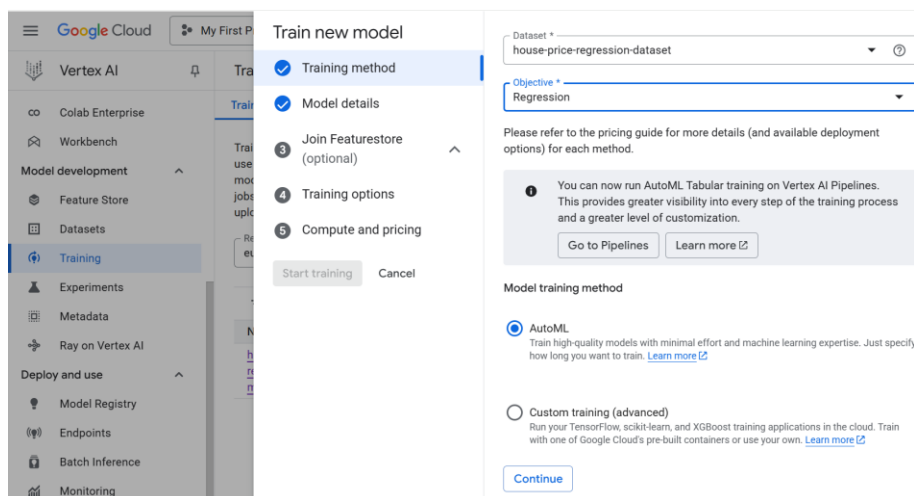
Screenshots:

1. Created a bucket first to store the data
2. Created data set – Tabular format and selected Regression





3. Model training using AutoML for Regression



4. Target set as House price

Google Cloud

My First Project

Vertex AI

Colab Enterprise

Workbench

Model development

Feature Store

Datasets

Training

Experiments

Metadata

Ray on Vertex AI

Deploy and use

Model Registry

Endpoints

Batch Inference

Monitoring

Provisioned Throughput

Pipelines

Tutorials

Train new model

Training method

Model details

Join Featurestore (optional)

Training options

Compute and pricing

Start training

Cancel

Train new model

Creates a new model group and assigns the trained model as version 1

Train new version

Trains model as a version of an existing model

Name *

house-price-regression-dataset

Description

Target column *

House_Price

Export test dataset to BigQuery

Advanced options

Continue

Train new model

Training method

Model details

Join Featurestore (optional)

Training options

Compute and pricing

Start training

Cancel

Data split

Random

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

Training: 80%

Validation: 10%

Test: 10%

Manual

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

Chronological

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](#)

Training 80%

Validation 10%

Testing 10%

Start time

End time

Encryption

Google-managed encryption key

Keys owned by Google

Cloud KMS key

Keys owned by customers

Train new model

Training method

Model details

Join Featurestore (optional)

Training options

Compute and pricing

Start training

Cancel

Before continuing, use the Transformation column to review and specify the data types in your dataset. If unspecified, AutoML will try to apply the most relevant transformation option.

Generate statistics

Filter

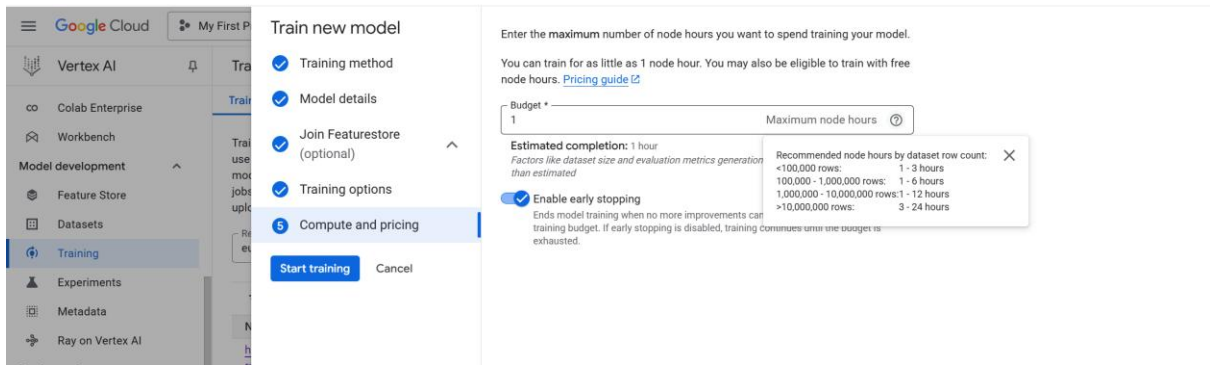
Enter property name or value

Column name	Transformation	Missing % (count)	Distinct values	Column type
Garage_Size	Categorical	-	3	-
House_Price	Target	-	1000	-
Lot_Size	Numeric	-	1000	-
Neighborhood_Quality	Numeric	-	10	-
Num_Bathrooms	Categorical	-	3	-
Num_Bedrooms	Categorical	-	5	-
Square_Footage	Numeric	-	894	-
Year_Built	Numeric	-	73	-

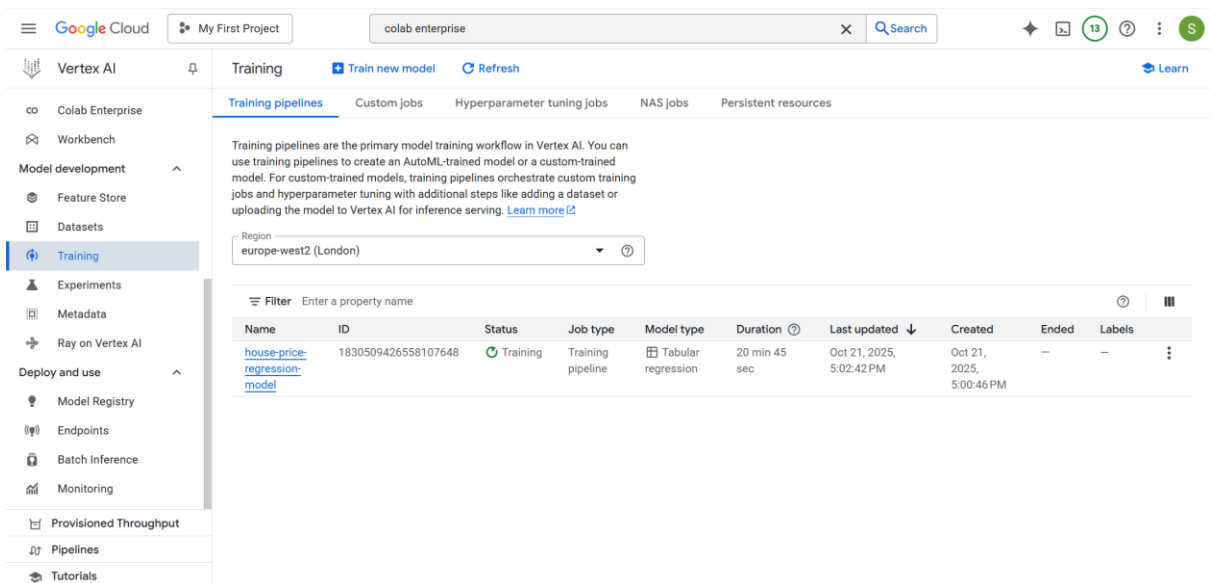
Total 8 feature columns are included in the training

Advanced options

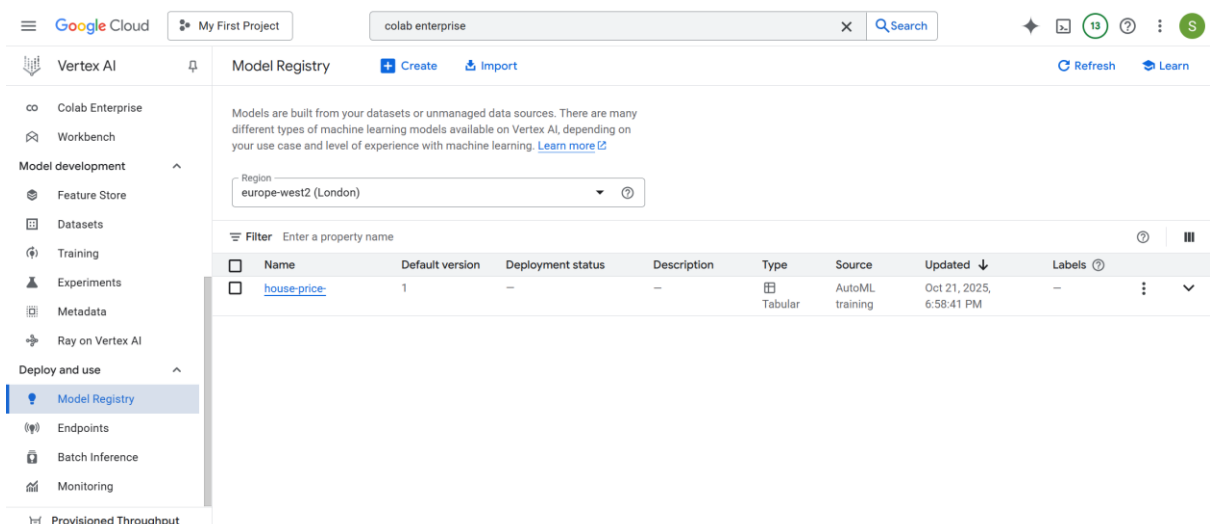
5. Training budget set to 1 hour depending on size of the dataset

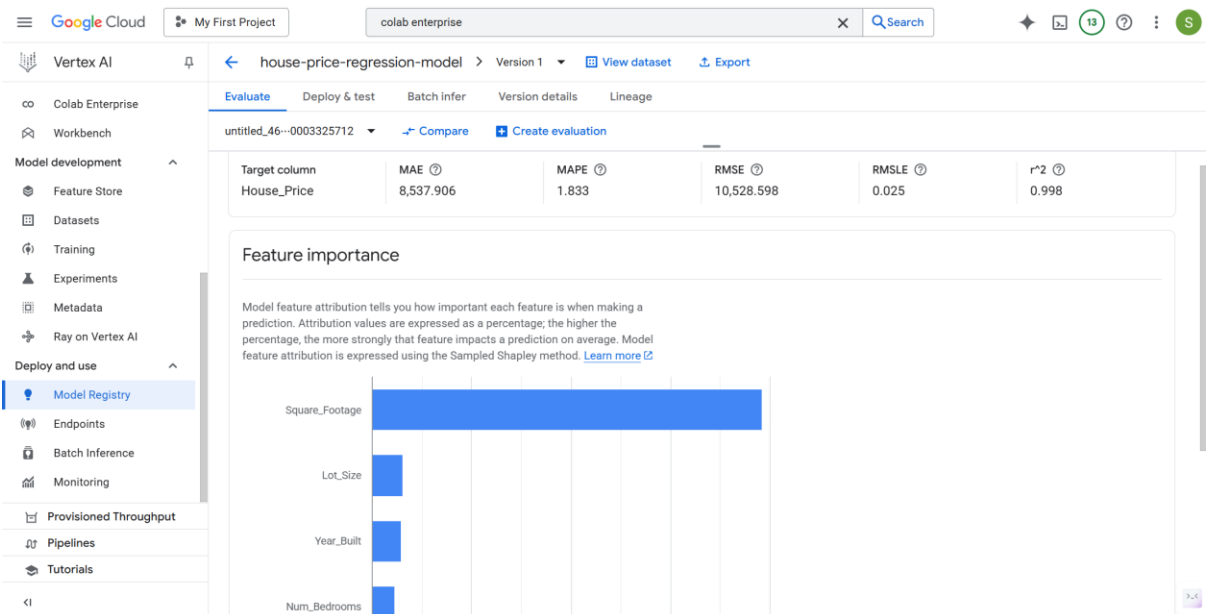


6. Model trained



7. Added to model registry





8. Deploying as end point

The screenshot shows the 'Deploy to endpoint' workflow in the Google Cloud Vertex AI interface. The 'Deploy & test' tab is active, and the 'Model settings' step is selected.

Deploy to endpoint

- Define your endpoint
- Model settings**
- Model monitoring
- Monitoring objectives

Model settings

Deploying a model to an endpoint lets it serve online inferences. You can also deploy multiple models to one endpoint and split traffic. This lets you test out a new model before serving all traffic. [Learn more about model deployment](#)

New model

house-price-regression-model (Version 1)

Traffic split * 100 %

Compute resources

Choose how compute resources will serve inference 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.

☐ Partial success of deployment

Google Cloud

My First Project

colab enterprise

Vertex AI

Colab Enterprise

Workbench

Model development

Feature Store

Datasets

Training

Experiments

Metadata

Ray on Vertex AI

Deploy and use

Model Registry

Endpoints

Batch Inference

Monitoring

Provisioned Throughput

Pipelines

Tutorials

house-price-regression-model

Evaluate

Deploy & test

Batch infer

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 are useful for timely inferences from many users (for exam request). You can also request batch inferences if you don't

Deploy to endpoint

Name	ID	Status	Models
No active endpoints containing this model			

Test your model

Deploy to endpoint

Define your endpoint

Model settings

Model monitoring

Monitoring objectives

Deploy

Cancel

Advanced scaling options

Machine type

n1-standard-8, 8 vCPUs, 30 GiB memory

Reservations

Reservation type

Restrictions may apply when using reservations. [Learn more](#)

Availability policies

VM provisioning model

Standard

Choose "Spot" to get a discounted, preemptible VM. Otherwise, stick to "Standard". [Learn more](#)

Logging

Logging settings are permanent for this endpoint, and 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

Explainability options

☒ No explainability

☐ Feature attribution

☐ Example-based explanation

It may take several minutes for endpoint settings to take effect.

Google Cloud

My First Project

colab enterprise

Vertex AI

Colab Enterprise

Workbench

Model development

Feature Store

Datasets

Training

Experiments

Metadata

Ray on Vertex AI

Deploy and use

Model Registry

Endpoints

Batch Inference

Monitoring

Provisioned Throughput

Pipelines

Tutorials

house-price-regression-model

Evaluate

Deploy & test

Batch infer

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 are useful for timely inferences from many users (for exam request). You can also request batch inferences if you don't

Deploy to endpoint

Name	ID	Status	Models
No active endpoints containing this model			

Test your model

Deploy to endpoint

Define your endpoint

Model settings

Model monitoring

Monitoring objectives

Deploy

Cancel

Advanced scaling options

Machine type

n1-standard-8, 8 vCPUs, 30 GiB memory

Reservations

Reservation type

Restrictions may apply when using reservations. [Learn more](#)

Availability policies

VM provisioning model

Standard

Choose "Spot" to get a discounted, preemptible VM. Otherwise, stick to "Standard". [Learn more](#)

Logging

Logging settings are permanent for this endpoint, and 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

Explainability options

☒ No explainability

☐ Feature attribution

☐ Example-based explanation

It may take several minutes for endpoint settings to take effect.

9. Ran the model using a sample instance as seen below

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Welcome to Cloud Shell! Type "help" to get started, or type "gemini" to try prompting with Gemini CLI.
To set your Cloud Platform project in this session use `gcloud config set project [PROJECT_ID]`.
You can view your projects by running `gcloud projects list`.
msaimukesh8055@cloudshell:~$ ls -ltr
total 8
-rw-r--r-- 1 msaimukesh8055 msaimukesh8055 238 Oct 21 12:31 sample_request.json
-rwxr-xr-x 1 msaimukesh8055 msaimukesh8055 913 Oct 21 15:01 README-cloudshell.txt
msaimukesh8055@cloudshell:~$ cat sample_request.json
{
  "instances": [
    {
      "Square_Footage": "1500",
      "Num_Bedrooms": "3",
      "Num_Bathrooms": "2",
      "Year_Built": "2005",
      "Lot_Size": "0.75",
      "Garage_Size": "1",
      "Neighborhood_Quality": "4"
    }
  ]
}
msaimukesh8055@cloudshell:~$ curl \
-X POST \
-H "Authorization: Bearer $(gcloud auth print-access-token)" \
-H "Content-Type: application/json" \
"https://europe-west2-aiplatform.googleapis.com/v1/projects/ivory-team-475605-m4/locations/europe-west2/endpoints/5553356154966376448:predict" \
-d "@sample_request.json"
```

Predicted the house value as below:

```
{
  "predictions": [
    {
      "value": 347485.4375,
      "upper_bound": 367028.3125,
      "lower_bound": 329450
    }
  ],
  "deployedModelId": "6515321077130854400",
  "model": "projects/579363279387/locations/europe-west2/models/7571241466912047104",
  "modelDisplayName": "house-price-regression-model",
  "modelVersionId": "1"
}
msaimukesh8055@cloudshell:~$ B
0 0 Cloud Code - No Project Ln 1, Col 1
```