

console.cloud.google.com/compute/instances?project=cmpe260&authuser=3

Google Cloud Platform cmpe260 compute engine

Compute Engine VM instances

INSTANCES INSTANCE SCHEDULE

VM instances are highly configurable virtual machines for running workloads on Google infrastructure. [Learn more](#)

Filter Enter property name or value

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>	tensorflow-2-3-20211125-131035	us-west1-b			10.138.0.2 (nic0)	None	SSH

Related actions

- View billing report
- Monitor VMs
- Explore VM logs
- Set up firewall rules
- Patch management

tensorflow-2-3-20211125-131035

PERMISSIONS LABELS MONITORING

Edit or delete permissions below or "Add Principal" to grant new

Show inherited permissions

Filter Enter property name or value

Role / Principal

- AI Platform Notebooks Service Agent (1)
- Editor (3)
- Owner (1)

console.cloud.google.com/apis/library/containerregistry.googleapis.com?utm_source=codelabs&utm_medium=et&utm_campaign=et&utm_content=et&utm_term=et

Google Cloud Platform cmpe260

Google Container Registry API

Google Enterprise API

Google Container Registry provides secure, private Docker image storage on Google Cloud Platform. ...

MANAGE API Enabled

OVERVIEW DOCUMENTATION

Overview

Google Container Registry provides secure, private Docker image storage on Google Cloud Platform. Our API follows the Docker Registry API specification, so we are fully compatible with the Docker CLI client, as well as standard tooling using the Docker Registry API.

Additional details

Type: SaaS & APIs

Last updated: 7/22/21

Category: Developer tools, Google Enterprise APIs

Service name: containerregistry.googleapis.com

Tutorials and documentation

Google Cloud Platform cmpe260 vertex AI

Vertex AI Notebooks

MANAGED NOTEBOOKS PREVIEW USER-MANAGED NOTEBOOKS EXECUTIONS PREVIEW SCHEDULES PREVIEW

Notebooks service has been moved under the Vertex AI Workbench service. Please find your Notebooks instances in Workbench under the User-Managed Notebooks tab.

As of the M80 DLVM release, all environments will include JupyterLab 3.x by default. To continue using an existing environment's JupyterLab 1.x version, disable auto-upgrade (if enabled) and do not manually upgrade the environment to a new environment version. To create new Notebooks with JupyterLab 1.x installed, [see creating specific versions of Notebooks](#).

Notebooks have JupyterLab pre-installed and are configured with GPU-enabled machine learning frameworks. [Learn more](#)

Filter Enter property name or value

Notebook name	Zone	Auto-upgrade	Environment	Machine type	GPUs	Permission
tensorflow-2-3-20211125-131035	us-west1-b	—	TensorFlow.2.3	4 vCPUs, 15 GB RAM	None	Service account

Info panel

DOCUMENTATION LABELS

Documentation Home

Registering legacy DLVMs

File Edit View Run Kernel Git Tabs Settings Help

Filter files by name

Name	Last Modified
/	
mpg	4 minutes ago
src	4 days ago
tutorials	4 days ago
Untitled.ipynb	4 days ago

Terminal 1

```
(base) jupyter@tensorflow-2-3-20211125-131035:~$ mkdir mpg
(base) jupyter@tensorflow-2-3-20211125-131035:~$ cd mpg
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ touch Dockerfile
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ FROM gcr.io/deeplearning-platform-release/tf2-cpu.2-3
bash: FROM: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ WORKDIR /root
bash: WORKDIR: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ WORKDIR /
bash: WORKDIR: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ # Copies the trainer code to the docker image.
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ COPY trainer /trainer
bash: COPY: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ # Sets up the entry point to invoke the trainer.
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ ENTRYPOINT ["python", "-m", "trainer.train"]
bash: ENTRYPOINT: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ ls
Dockerfile
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ vi Dockerfile
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gcloud config list --format 'value(core.project)'
cpe260
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ PROJECT_ID=cpe260
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ BUCKET_NAME="gs://cpe260-bucket"
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gsutil mb -l us-central1 $BUCKET_NAME
InvalidUrlError: Invalid bucket name in URL "-bucket".
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gsutil mb -l us-central1 $BUCKET_NAME
InvalidUrlError: Invalid bucket name in URL "-bucket".
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ BUCKET_NAME="gs://$(PROJECT_ID)-bucket"
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gsutil mb -l us-central1 $BUCKET_NAME
Creating gs://cpe260-bucket/...
```

Simple 1 1 1 1

Terminal 1

File Edit View Run Kernel Git Tabs Settings Help

Filter files by name

Name	Last Modified
/	
mpg	4 minutes ago
src	4 days ago
tutorials	4 days ago
Untitled.ipynb	4 days ago

Terminal 1

```
(base) jupyter@tensorflow-2-3-20211125-131035:~$ mkdir mpg
(base) jupyter@tensorflow-2-3-20211125-131035:~$ cd mpg
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ touch Dockerfile
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ FROM gcr.io/deeplearning-platform-release/tf2-cpu.2-3
bash: FROM: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ WORKDIR /root
bash: WORKDIR: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ WORKDIR /
bash: WORKDIR: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ # Copies the trainer code to the docker image.
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ COPY trainer /trainer
bash: COPY: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ # Sets up the entry point to invoke the trainer.
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ ENTRYPOINT ["python", "-m", "trainer.train"]
bash: ENTRYPOINT: command not found
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ ls
Dockerfile
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ vi Dockerfile
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gcloud config list --format 'value(core.project)'
cpe260
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ PROJECT_ID=cpe260
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ BUCKET_NAME="gs://cpe260-bucket"
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gsutil mb -l us-central1 $BUCKET_NAME
InvalidUrlError: Invalid bucket name in URL "-bucket".
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gsutil mb -l us-central1 $BUCKET_NAME
InvalidUrlError: Invalid bucket name in URL "-bucket".
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ BUCKET_NAME="gs://$(PROJECT_ID)-bucket"
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ gsutil mb -l us-central1 $BUCKET_NAME
Creating gs://cpe260-bucket/...
```

Simple 1 1 1 1

Terminal 1

```
File Edit View Run Kernel Git Tabs Settings Help

Terminal 1

(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ mkdir trainer
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ touch trainer/train.py
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ ls
Dockerfile  trainer
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ cd trainer/
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg/trainer$ vi train.py
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg/trainer$ docker build -t gcr.io/$PROJECT_ID/mpg:v1 .
unable to prepare context: unable to evaluate symlinks in Dockerfile path: latest /home/jupyter/mpg/trainer/Dockerfile: no such file or directory
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg/trainer$ cd ..
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$ docker build -t gcr.io/$PROJECT_ID/mpg:v1 .
Sending build context to Docker daemon 8.704kB
Step 1/5 : FROM gcr.io/deeplearning-platform-release/tf2-cpu.2-3
latest: Pulling from deeplearning-platform-release/tf2-cpu.2-3
7b1afab2e4d4: Pull complete
6d576096c0bf: Pull complete
5a39c8988a9a: Pull complete
3bf0fd278fc1: Pull complete
4f4fb706ef54: Pull complete
a7b7cd42e273: Pull complete
5ed778ec318f: Pull complete
f99f0c7e1e5e: Pull complete
f330924abe9c: Pull complete
c0c380d7a70d: Pull complete
ea41e0ef378e: Pull complete
43854f3d04da: Pull complete
536331575f08: Pull complete
2e85fd51a83d: Pull complete
4d20ed165f23: Pull complete
1e0b4f132f82: Pull complete
f15ad2a24b1a: Pull complete
3774007c1526: Pull complete
94bbf24e27a0: Pull complete
f5423bf6d752: Pull complete
Digest: sha256:d4ed0cd71aa0f635f9df9138f39df19fdd4d4dcdf1322b33a6582ac13f4aa9e6
Status: Downloaded newer image for gcr.io/deeplearning-platform-release/tf2-cpu.2-3:latest
--> 452a88fe95cc
Step 2/5 : WORKDIR /root
--> Running in 0d73a02c9f26
Removing intermediate container 0d73a02c9f26
--> f56c8bf8e624
Step 3/5 : WORKDIR /
--> Running in bce9df33a7d9
Removing intermediate container bce9df33a7d9
--> 54c8dab9aa8f
Step 4/5 : COPY train.py /train.py
--> 22f09d4c623c
Step 5/5 : ENTRYPOINT ["python", "-m", "trainer.train"]
--> Running in 14567a253da5
Removing intermediate container 14567a253da5
--> 9679df48c4b7
Successfully built 9679df48c4b7
Successfully tagged gcr.io/cmp260/mpg:v1
(base) jupyter@tensorflow-2-3-20211125-131035:~/mpg$
```

Google Cloud Platform

Vertex AI

- Dashboard
- Datasets
- Features
- Labeling tasks
- Workbench
- 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.

AutoML options are only available when you train with a managed dataset.

- ☐ 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

Google Cloud Platform

Vertex AI

Dashboard

Datasets

Features

Labeling tasks

Workbench

Pipelines

Training

Experiments

Models

Endpoints

Batch predictions

Metadata

Model drift cases

Reusability

Name

fraud

Train new model

☒ Training method
 ☒ Model details
 ☒ Training container
 ☒ Hyperparameters (optional)
 ☒ Compute and pricing
 ☒ Prediction container (optional)

START TRAINING

CANCEL

You can associate your custom-trained model with a container in order to serve prediction requests using Vertex AI. [Learn more about getting predictions](#)

☐ No prediction container
 You can always import your model artifact later to serve prediction requests

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

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

Pre-built container settings

Vertex AI provides Docker container images for serving predictions. To use a pre-built container, your trained model code must be in Python 3.7. [Learn more about pre-built containers](#)

In order to run in a pre-built container, your code needs to be in Python 3.7

Model framework *

TensorFlow

Model framework version *

2.1

Accelerator type *

None

Model directory *

gs://cmpe260-bucket/mpg

BROWSE

Cloud Storage location containing the model artifact and any supporting files

Predict schemata

Optional. [Learn more about the predict schemata](#)

Instances

BROWSE

Cloud Storage location to a YAML file that defines the format of a single instance used in prediction and explanation requests.

Parameters

BROWSE

Cloud Storage location to a YAML file that defines the prediction and explanation parameters.

Predictions

BROWSE

Cloud Storage location to a YAML file that defines the format of a single prediction or explanation.

Google Cloud Platform

cmpe260

vertex AI

Vertex AI

Dashboard

Datasets

Features

Labeling tasks

Workbench

Pipelines

Training

Experiments

Models

Endpoints

Batch predictions

Metadata

Training

CREATE

REFRESH

TRAINING PIPELINES

CUSTOM JOBS

HYPERPARAMETER TUNING JOBS

Training pipelines are the primary model training workflow in Vertex AI. You can use training pipelines to create an AutoML-trained model or a custom-trained model. For custom-trained models, training pipelines orchestrate custom training jobs and hyperparameter tuning with additional steps like adding a dataset or uploading the model to Vertex AI for prediction serving. [Learn More](#)

Region

us-central1 (Iowa)

Filter

Enter a property name

Name	ID	Status	Job type	Model type	Created	Elapsed time
fraud_detection_2021112512351	850544811852169216	Finished	Training pipeline	Tabular classification	Nov 25, 2021, 1:26:32 PM	1 hr 56 min
automl-beans1637862305	1811500382342348800	Finished	Training pipeline	Tabular classification	Nov 25, 2021, 9:49:58 AM	2 hr 22 min

File Edit View Run Kernel Git Tabs Settings Help

+

Filter files by name

/

mpg

src

tutorials

Untitled.ipynb

Untitled1.ipynb

29 minutes ago

4 days ago

4 days ago

4 days ago

seconds ago

Untitled1.ipynb

Python 3

```

e-c-cloud-us-quey>=0.0dev,>=1.15.0->google-cloud-bigquery<3.0.0dev,>=1.12.0->google-cloud-storage<2.0.0dev,>=1.32.0->google-cloud-aiplatform (1.16.0)
Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from google-cloud-storage<2.0.0dev,>=1.32.0->google-cloud-aiplatform) (1.16.0)
Requirement already satisfied: pyparsing==2.0.2 in /opt/conda/lib/python3.7/site-packages (from packaging==14.3->google-cloud-aiplatform) (3.0.6)
Requirement already satisfied: rsa<5,>=3.1.4 in /opt/conda/lib/python3.7/site-packages (from google-auth<3.0dev,>=1.25.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (4.7.2)
Requirement already satisfied: pyasn1-modules==0.2.1 in /opt/conda/lib/python3.7/site-packages (from google-auth<3.0dev,>=1.25.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (0.2.7)
Requirement already satisfied: cachetools<5.0,>=2.0.0 in /opt/conda/lib/python3.7/site-packages (from google-auth<3.0dev,>=1.25.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (4.2.4)
Requirement already satisfied: google-crc32c<2.0dev,>=1.0 in /opt/conda/lib/python3.7/site-packages (from google-resumable-media<3.0dev,>=0.6.0->google-cloud-bigquery<3.0.0dev,>=1.15.0->google-cloud-aiplatform) (1.1.2)
Requirement already satisfied: chardet<5,>=3.0.2 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0dev,>=2.18.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (4.0.0)
Requirement already satisfied: idna<3,>=2.5 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0dev,>=2.18.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (2.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0dev,>=2.18.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (1.26.7)
Requirement already satisfied: certifi==2017.4.17 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0dev,>=2.18.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (2021.10.8)
Requirement already satisfied: cffi==1.0.0 in /opt/conda/lib/python3.7/site-packages (from google-crc32c<2.0dev,>=1.0->google-resumable-media<3.0dev,>=0.6.0->google-cloud-bigquery<3.0.0dev,>=1.15.0->google-cloud-aiplatform) (1.15.0)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in /opt/conda/lib/python3.7/site-packages (from pyasn1-modules==0.2.1->google-auth<3.0dev,>=1.25.0->google-api-core[grpc]<3.0.0dev,>=1.26.0->google-cloud-aiplatform) (0.4.8)
Requirement already satisfied: pycparser in /opt/conda/lib/python3.7/site-packages (from cffi==1.0.0->google-crc32c<2.0dev,>=1.0->google-resumable-media<3.0dev,>=0.6.0->google-cloud-bigquery<3.0.0dev,>=1.15.0->google-cloud-aiplatform) (2.21)

[ ]: from google.cloud import aiplatform

endpoint = aiplatform.Endpoint(
    endpoint_name="projects/YOUR-PROJECT-NUMBER/locations/us-central1/endpoints/YOUR-ENDPOINT-ID"
)

[ ]: test_mpg = [1.4838871833555929,
1.8659883497083019,
2.234620276849616,
1.0187816540094983,
-2.530898710602246,
-1.6046416850441676,
-0.4651483719733302,
-0.4952254087173721,
0.7746763768735953]

response = endpoint.predict([test_mpg])

print('API response: ', response)

print('Predicted MPG: ', response.predictions[0][0])

```

Simple

1 2

Python 3 | Idle

Mode: Edit

Ln 15, Col 53

Untitled1.ipynb