

Shreya Goyal  
Assignment : VERTEX AI

Part a) <https://codelabs.developers.google.com/vertex-pipelines-intro#0>

The screenshot shows the Google Cloud Platform Vertex AI Workbench interface. On the left, there's a sidebar with options like Dashboard, Datasets, Features, Labeling tasks, Workbench (which is selected), Pipelines, Training, Experiments, Models, Endpoints, Batch predictions, and Metadata. Below that is a Marketplace section. The main area has a search bar at the top right with the text "billing projects". A modal window titled "New notebook" is open in the center. It contains fields for "Notebook name" (set to "tensorflow-2-3-20211204-091213"), "Region" (set to "us-west1 (Oregon)"), and "Zone" (set to "us-west1-a"). Under "Notebook properties", there are sections for Environment (TensorFlow Enterprise 2.3), Machine type (4 vCPUs, 15 GB RAM), Boot disk (100 GB Standard persistent disk), Data disk (100 GB Standard persistent disk), Subnetwork (default(10.138.0.0/20)), External IP (Ephemeral(Automatic)), Permission (Compute Engine default service account), and Estimated cost (\$102.70 monthly, \$0.141 hourly). At the bottom of the modal are "ADVANCED OPTIONS", "CANCEL", and "CREATE" buttons.

← → ⌂ console.cloud.google.com/vertex-ai/workbench/list/instances?project=haysack1

Apps Save All Images Subjects Master's Project www.ee.oulu.fi Question Bank - fi...

Google Cloud Platform • Haysack1 ▾

billing projects

Vertex AI Notebooks NEW NOTEBOOK REFRESH START STOP RESET DELETE SHOW INFO PANEL

Dashboard Datasets Features Labeling tasks Workbench Pipelines Training Experiments Models Endpoints Batch predictions Metadata Marketplace

MANAGED NOTEBOOKS PREVIEW USER-MANAGED NOTEBOOKS EXECUTIONS PREVIEW SCHEDULES PREVIEW

**Notebooks**

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	Last modified
<input type="checkbox"/>	example-instance OPEN JUPYTERLAB	us-central1-a	—	PyTorch:1.9	4 vCPUs, 15 GB RAM	None	Service account	Dec 4, 2023 9:07:03
<input type="checkbox"/>	tensorflow-2-3-20211204-091213 OPEN JUPYTERLAB	us-west1-a	—	TensorFlow:2.3	4 vCPUs, 15 GB RAM	None	Service account	Dec 4, 2023 9:16:17

← → ⌂ 1766f01a2187d253-dot-us-west1.notebooks.googleusercontent.com/lab

Apps Save All Images Subjects Master's Project www.ee.oulu.fi Question Bank - fi...

File Edit View Run Kernel Git Tabs Settings Help

Launcher

Filter files by name

Name Last Modified

- src 6 minutes ago
- tutorials 6 minutes ago

**Notebook**

- Python 3
- Python [conda env:root] \*

**Console**

- Python 3
- Python [conda env:root] \*

**Other**

- Terminal
- Text File
- Markdown File
- Python File
- Show Contextual Help

Simple 0 0 0 0 0

Launcher

1766f01a2187d253-dot-us-west1.notebooks.googleusercontent.com/lab/tree/Untitled.ipynb

File Edit View Run Kernel Git Tabs Settings Help

Launcher Untitled.ipynb Python 3

```

KFP SDK version: 1.8.9
google_cloud_pipeline_components version: 0.2.0

[2]: import os
PROJECT_ID = "haytsack1"

# Get your Google Cloud project ID from gcloud
if not os.getenv("IS_TESTING"):
    shell_output=gcloud config list --format 'value(core.project)' 2>/dev/null
    PROJECT_ID = shell_output[0]
    print("Project ID: ", PROJECT_ID)

Project ID: haytsack1

[3]: BUCKET_NAME="gs://" + PROJECT_ID + "-bucket"

[4]: import kfp

from kfp.v2 import compiler, dsl
from kfp.v2.dsl import component, pipeline, Artifact, ClassificationMetrics, Input, Output, Model, Metrics

from google.cloud import aiplatform
from google.cloud.pipeline_components import aiplatform as gcc_aip
from typing import NamedTuple

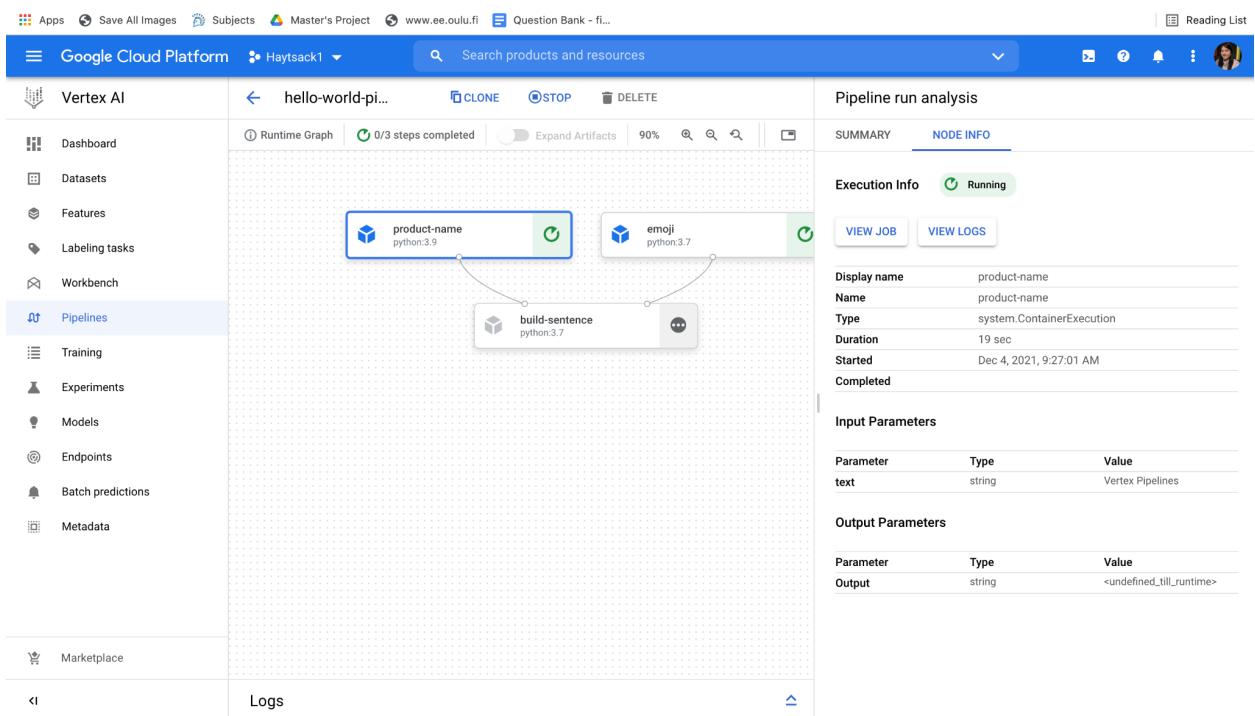
[5]: PATH=$env PATH
env PATH=$PATH:/home/jupyter/.local/bin
REGION="us-central1"

PIPELINE_ROOT = f"{BUCKET_NAME}/pipeline_root/"
PIPELINE_ROOT

env: PATH=/usr/local/cuda/bin:/opt/conda/bin:/opt/conda/condabin:/usr/local/bin:/bin:/usr/local/games:/usr/games:/home/jupyter/.local/bin
PIPELINE_ROOT
'gs://haytsack1-bucket/pipeline_root/'

[ ]:
```

Simple 0 1 Python 3 | Idle Mode: Edit Ln 4, Col 1 Untitled.ipynb



Part b) <https://codelabs.developers.google.com/vertex-automl-tabular#0>

console.cloud.google.com/vertex-ai/datasets/create?project=haytsack1

Apps Save All Images Subjects Master's Project www.ee.oulu.fi Question Bank - fi...

Update Reading List

### Create dataset

**Dataset name \*** fraud\_detection  
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](#)

**TABULAR** (selected)

**IMAGE** **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)

**ADVANCED OPTIONS**

**CREATE** **CANCEL**

console.cloud.google.com/vertex-ai/locations/us-central1/datasets/976366325465088/analyze?project=haytsack1

Apps Save All Images Subjects Master's Project www.ee.oulu.fi Question Bank - fi...

Reading List

### fraud\_detection

**SOURCE** **ANALYZE** (selected)

**Dataset Info**  
Created: Dec 04, 2021 9:38 AM  
Dataset format: BigQuery  
Dataset location(s): [bq://bigquery-public-data\\_fraud\\_detection](bq://bigquery-public-data_fraud_detection)

**Summary**  
Total columns: 31  
Total rows: -

FLOAT 30 (96.77%)  
INTEGER 1 (3.23%)

**GENERATE STATISTICS**

**Filter** Enter property name or value

Column 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	-	-
V12	FLOAT	NULLABLE	-	-

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

**TRAIN NEW MODEL**

Google Cloud Platform

**Train new model**

1 Training method (checked)

2 Model details: Model name \* fraud\_detection\_2021124174133, Target column \* Class (INTEGER)

3 Training options

4 Compute and pricing

Export test dataset to BigQuery (unchecked)

ADVANCED OPTIONS

CONTINUE

START TRAINING CANCEL

Google Cloud Platform

Haytsack1

Training

CREATE

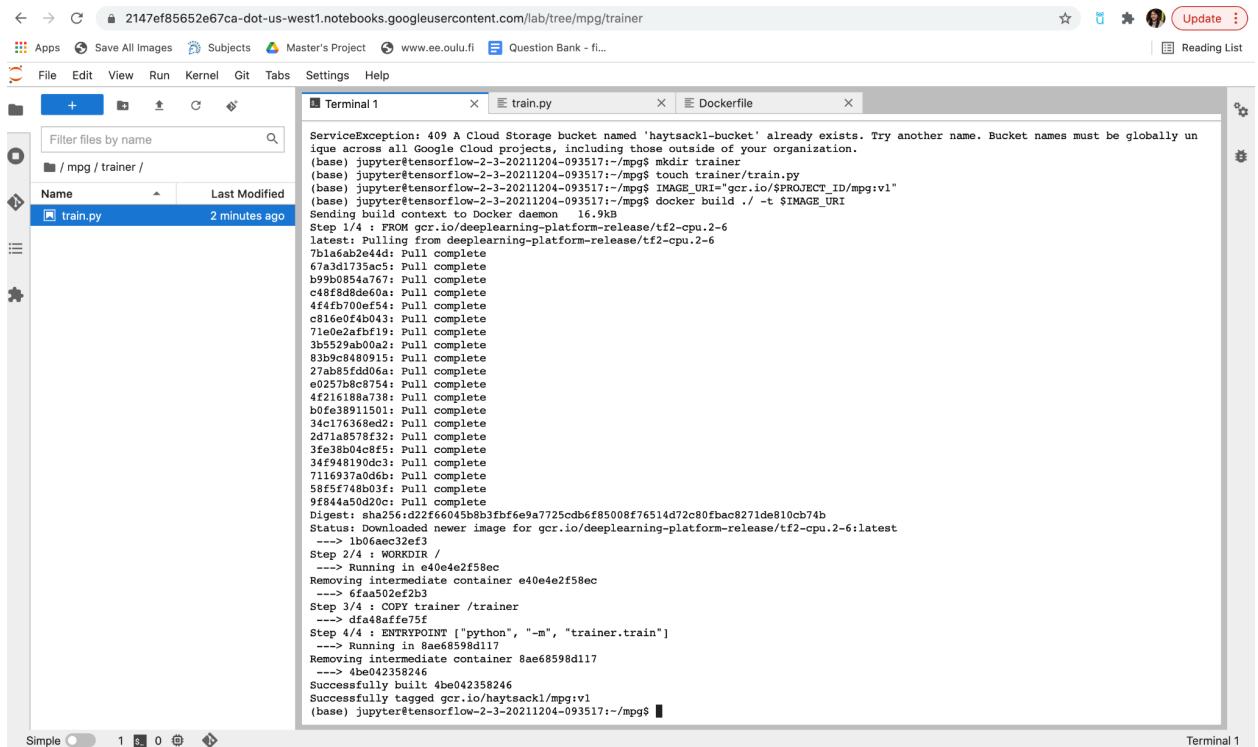
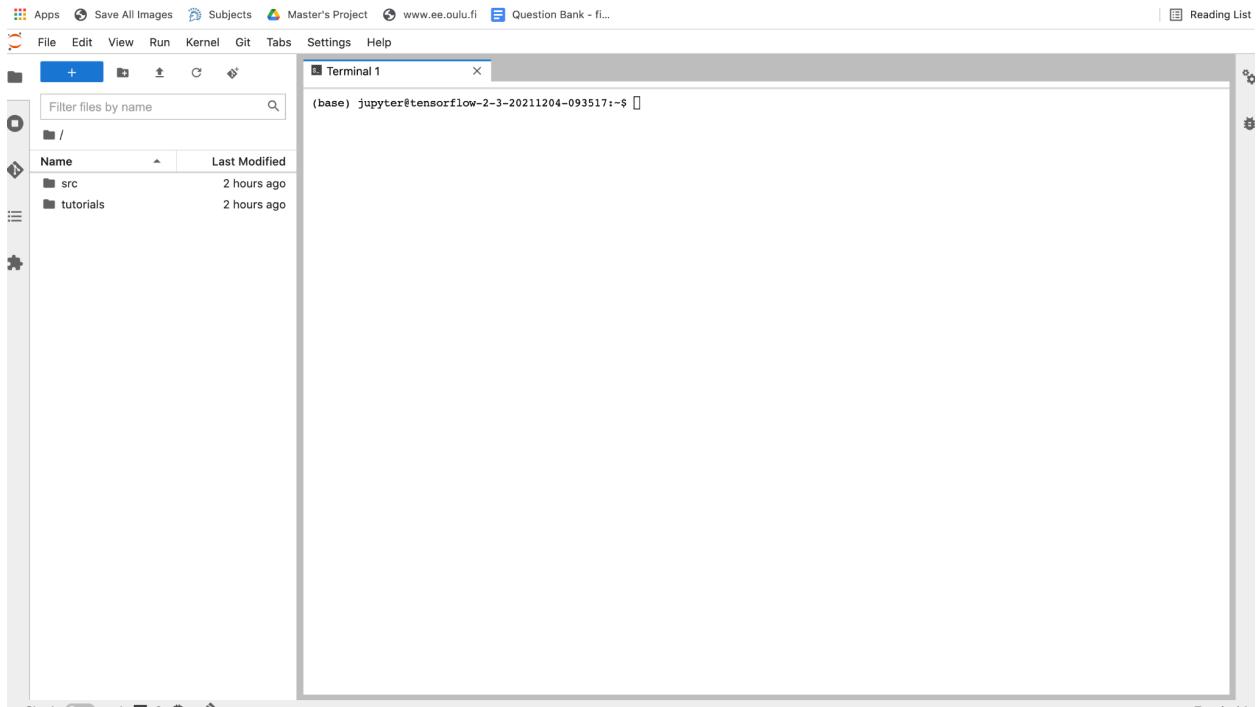
REFRESH

Region: us-central1 (Iowa)

Filter: Enter a property name

Name	ID	Status	Job type	Model type	Created	Elapsed time	Labels
fraud_detection_2021124174133	1123262178447589376	Training	Training pipeline	Tabular classification	Dec 4, 2021, 9:43:26 AM	1 hr 32 min	-

Part c) [https://codelabs.developers.google.com/vertex\\_custom\\_training\\_prediction#0](https://codelabs.developers.google.com/vertex_custom_training_prediction#0)



```

Epoch 44/1000
8/8 [=====] - 0s 6ms/step - loss: 7.2625 - mae: 1.8994 - mse: 7.2625 - val_loss: 8.8492 - val_mae: 2.2638 - val_mse: 8.8492
Epoch 45/1000
8/8 [=====] - 0s 6ms/step - loss: 7.2998 - mae: 1.9054 - mse: 7.2998 - val_loss: 8.7706 - val_mae: 2.2936 - val_mse: 8.7706
Epoch 46/1000
8/8 [=====] - 0s 7ms/step - loss: 7.0895 - mae: 1.9006 - mse: 7.0895 - val_loss: 9.4730 - val_mae: 2.2412 - val_mse: 9.4730
Epoch 47/1000
8/8 [=====] - 0s 6ms/step - loss: 7.2303 - mae: 1.8992 - mse: 7.2303 - val_loss: 8.9592 - val_mae: 2.2868 - val_mse: 8.9592
Epoch 48/1000
8/8 [=====] - 0s 7ms/step - loss: 7.1156 - mae: 1.8647 - mse: 7.1156 - val_loss: 8.8877 - val_mae: 2.3046 - val_mse: 8.8877
Epoch 49/1000
8/8 [=====] - 0s 6ms/step - loss: 6.9258 - mae: 1.8454 - mse: 6.9258 - val_loss: 8.8305 - val_mae: 2.2941 - val_mse: 8.8305
2021-12-04 20:01:46.981112: W tensorflow/python/util/util.cc:48] Sets are not currently considered sequences, but this may change in the future. Code depends on this behavior.
(base) jupyter@tensorflow-2-3-20211204-093517:~/mpg$ docker push $IMAGE_URI
The push refers to repository [gcr.io/haystack1/mpg]
73f984dd6466: Pushed
2699358e44b2: Mounted from deeplearning-platform-release/tf2-cpu.2-6
393bdeff6b5: Mounted from deeplearning-platform-release/tf2-cpu.2-6
cf3738cf85bb: Mounted from deeplearning-platform-release/tf2-cpu.2-6
1b3c3815787f: Mounted from deeplearning-platform-release/tf2-cpu.2-6
244680cd2339: Mounted from deeplearning-platform-release/tf2-cpu.2-6
2d3bbea2c63d: Mounted from deeplearning-platform-release/tf2-cpu.2-6
b538f5dfbd26: Mounted from deeplearning-platform-release/tf2-cpu.2-6
539edf6bfe8: Mounted from deeplearning-platform-release/tf2-cpu.2-6
f212849e96cc: Mounted from deeplearning-platform-release/tf2-cpu.2-6
7121dccc04ae: Mounted from deeplearning-platform-release/tf2-cpu.2-6
c789a84d37e: Mounted from deeplearning-platform-release/tf2-cpu.2-6
9166f4ad9608: Mounted from deeplearning-platform-release/tf2-cpu.2-6
2fbcb400914c0: Mounted from deeplearning-platform-release/tf2-cpu.2-6
b69e0b09e4c0: Mounted from deeplearning-platform-release/tf2-cpu.2-6
54363f142f4d: Mounted from deeplearning-platform-release/tf2-cpu.2-6
5f70ff18a086: Layer already exists
v1: digest: sha256:27fc0864233ebab2d0442c95690d0277bf296e196a25ae415a024c245e40647 size: 4713
(base) jupyter@tensorflow-2-3-20211204-093517:~/mpg$ 

```

## Part d) <https://codelabs.developers.google.com/vertex-workbench-intro#0>

Time of inactivity before shutdown (Minutes) \*  
60  
Must be integer: 1-600

**Networking**  
The network must have outbound connection to the internet. [Learn more](#) about the networking options below.

- Google-managed networks  
No configuration required
- Networks in this project  
Private service access required
- Networks shared with me  
Private service access required

**Security**

- Enable nbconvert [?](#)
- Enable file downloading from Notebook UI

**Permission**  
Managed notebooks currently only support single user only mode. Single user only mode restricts access to the user specified below.

Owner \*  
shreya.goyal@sjsu.edu

Quota exceeded for quota metric 'Create Runtime API requests' and limit 'Create Runtime API requests per minute' of service 'notebooks.googleapis.com' for consumer 'project\_number:840891042430'

← → C https://console.cloud.google.com/vertex-ai/workbench/create-managed?project=haytsack1

Apps Save All Images Subjects Master's Project www.ee.oulu.fi Question Bank - fi...

Google Cloud Platform Haytsack1 Search products and resources

Vertex AI Create a managed notebook

Notebook name \* london-bike  
Name must be 63 characters or less, must start with a letter and include only lowercase letters, digits, or '-'.

Region \* us-central1 (Iowa)

Advanced settings

CREATE CANCEL

Dashboard Datasets Features Labeling tasks Workbench Pipelines Training Experiments Models Endpoints Batch predictions Metadata Marketplace

This screenshot shows the Google Cloud Platform Vertex AI Workbench interface. On the left, there is a sidebar with various AI-related services like Dashboard, Datasets, Features, Labeling tasks, Pipelines, Training, Experiments, Models, Endpoints, Batch predictions, and Metadata. Below this is a Marketplace section. The main area is titled 'Create a managed notebook' and contains fields for 'Notebook name' (set to 'london-bike') and 'Region' (set to 'us-central1 (Iowa)'). There is also an 'Advanced settings' dropdown and a 'CREATE' button at the bottom. The URL in the browser bar is https://console.cloud.google.com/vertex-ai/workbench/create-managed?project=haytsack1.

## Part f) <https://codelabs.developers.google.com/vertex-hyperparameter-tuning#0>

← → C https://2147ef85652e67ca-dot-us-west1.notebooks.googleusercontent.com/lab/tree/horses\_or\_humans/Dockerfile

File Edit View Run Kernel Git Tabs Settings Help

Terminal 1 Dockerfile Dockerfile

```
1 FROM gcr.io/deeplearning-platform-release/tf2-gpu.2-5
2
3 WORKDIR /
4
5 # Installs hypertune library
6 RUN pip install cloudml-hypertune
7
8 # Copies the trainer code to the docker image.
9 COPY trainer /trainer
10
11 # Sets up the entry point to invoke the trainer.
12 ENTRYPOINT ["python", "-m", "trainer.task"]
```

This screenshot shows a Jupyter Notebook interface. The top navigation bar includes File, Edit, View, Run, Kernel, Git, Tabs, Settings, and Help. The tabs are labeled Terminal 1, Dockerfile, and Dockerfile. The Dockerfile tab is active, displaying a text editor with a Dockerfile content. The Dockerfile defines a base image, sets the working directory to /, installs the cloudml-hypertune library, copies the 'trainer' code to the /trainer directory, and sets the entry point to 'python -m trainer.task'. The URL in the browser bar is https://2147ef85652e67ca-dot-us-west1.notebooks.googleusercontent.com/lab/tree/horses\_or\_humans/Dockerfile.

The screenshot shows a Jupyter Notebook interface with two tabs open: `task.py` and `Dockerfile`. The `task.py` tab contains Python code for creating a neural network and performing hyperparameter tuning. The `Dockerfile` tab shows a Docker build process.

```

44
45     def create_model(num_neurons, learning_rate, momentum):
46         """Defines and compiles model."""
47
48         inputs = tf.keras.Input(shape=(150, 150, 3))
49         x = tf.keras.layers.Conv2D(16, (3, 3), activation='relu')(inputs)
50         x = tf.keras.layers.MaxPooling2D((2, 2))(x)
51         x = tf.keras.layers.Conv2D(32, (3, 3), activation='relu')(x)
52         x = tf.keras.layers.MaxPooling2D((2, 2))(x)
53         x = tf.keras.layers.Conv2D(64, (3, 3), activation='relu')(x)
54         x = tf.keras.layers.MaxPooling2D((2, 2))(x)
55         x = tf.keras.layers.Flatten()(x)
56         x = tf.keras.layers.Dense(num_neurons, activation='relu')(x)
57         outputs = tf.keras.layers.Dense(1, activation='sigmoid')(x)
58         model = tf.keras.Model(inputs, outputs)
59         model.compile(
60             loss='binary_crossentropy',
61             optimizer=tf.keras.optimizers.SGD(learning_rate=learning_rate, momentum=momentum),
62             metrics=['accuracy'])
63
64     return model
65
66
67 def main():
68     args = get_args()
69     train_data, validation_data = create_dataset()
70     model = create_model(args.num_neurons, args.learning_rate, args.momentum)
71     history = model.fit(train_data, epochs=NUM_EPOCHS, validation_data=validation_data)
72
73     # DEFINE METRIC
74     hp_metric = history.history['val_accuracy'][-1]
75
76     hpt = hypertune.HyperTune()
77     hpt.report_hyperparameter_tuning_metric(
78         hyperparameter_metric_tag='accuracy',
79         metric_value=hp_metric,
80         global_step=NUM_EPOCHS)
81
82
83 if __name__ == "__main__":
84     main()
85
86
87
88
89
90
91
92
93
94

```

```

80208bd6f7fb: Pull complete
b83c16bef138: Pull complete
9d1427033824: Pull complete
c0028679f003: Pull complete
09c222efff04: Pull complete
ae6048a3aace1: Pull complete
lced637de50b: Pull complete
762ff1eb7f161: Pull complete
5f8f4a2e0000: Pull complete
59c44922aa: Pull complete
b6c7eb38f366: Pull complete
5209e8017b4d: Pull complete
Digest: sha256:fb35899e63342df5fe603b7a567bcac8a8fc0744bec058fd1a1cd9d90637
Status: Downloaded newer image for gcr.io/deeplearning-platform-release/tf2-gpu.2-5:latest
--> 307b41bleac7
Step 2/5 : WORKDIR /
--> Running in c8449e5d0393
Removing intermediate container c8449e5d0393
--> 8b3a03037ef0
Step 3/5 : RUN pip install cloudml-hypertune
--> Running in 3dc5f09c009e
Collecting cloudml-hypertune
  Downloading https://pypi.org/packages/0a/00/5e59d4356cff5da4ed6b3d53e7c7002ed/cloudml_hypertune-0.1.0.dev6.tar.gz (3.2 kB)
Building wheels for collected packages: cloudml-hypertune
  Building wheel for cloudml-hypertune (setup.py): started
  Building wheel for cloudml-hypertune (setup.py): finished with status 'done'
    Created wheel for cloudml-hypertune: filename=cloudml_hypertune-0.1.0.dev6-py3-none-any.whl size=3988 sha256=5efe987639a549d70fd2efaf73939
6581fe6c4356cff5da4ed6b3d53e7c7002ed
  Stored in directory: /root/.cache/pip/wheels/a7/f8/87/e7bed0c2741fe219b3d6da67c2431d7f7fdb183032e00f81e
Successfully built cloudml-hypertune
Successfully installed cloudml-hypertune-0.1.0.dev6
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypy.io/warnings/venv
Removing intermediate container 3dc5f09c009e
--> 3dc5f09c009e
Step 4/5 : COPY trainer /trainer
--> 7088b2a80f6c
Step 5/5 : ENTRYPOINT ['python', "-m", "trainer.task"]
--> Running in 2c19ced348e6
Removing intermediate container 2c19ced348e6
--> 2d13402ade2b
Successfully built 2d13402ade2b
Successfully tagged gcr.io/haystack1/horse-human:hypertune
(base) jupyter@tensorflow-2-3-20211204-093517:~/horses_or_humans$ 

```

Part g) <https://codelabs.developers.google.com/vertex-mlmd-pipelines#0>

The screenshot shows the Google Cloud Platform Vertex AI Workbench interface. On the left, there's a sidebar with various options like Dashboard, Datasets, Features, Labeling tasks, Workbench (which is selected), Pipelines, Training, Experiments, Models, Endpoints, Batch predictions, and Metadata. The main area is titled "Create a managed notebook". It has fields for "Notebook name" (set to "managed-notebook-1638572301") and "Region" (set to "us-central1 (Iowa)"). Below these are "Advanced settings" and two buttons: "CREATE" and "CANCEL". A modal window is displayed with the message: "Quota exceeded for quota metric 'Create Runtime API requests' and limit 'Create Runtime API requests per minute' of service 'notebooks.googleapis.com' for consumer 'project\_number:310606389043'." There's also a "Marketplace" link at the bottom of the sidebar.

The screenshot shows the Google Cloud Platform Vertex AI Model page. The sidebar includes options like Dashboard, Datasets, Features, Labeling tasks, Workbench, Pipelines, Training, Experiments, Models (selected), Endpoints, Batch predictions, and Metadata. The main content area shows a model named "fraud\_detection\_2021123195457". It has tabs for EVALUATE, DEPLOY & TEST (selected), BATCH PREDICTIONS, and MODEL PROPERTIES. Under "Deploy your model", it says "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." A "DEPLOY TO ENDPOINT" button is present. Below this, a table lists an endpoint named "endpointfraud" with ID 141731446866837504, Status Active, Models 0, Region us-central1, Monitoring Disabled, and Last updated Dec 3, 2021, 2:34:54 PM. An "API" column shows "Sample request". Under "Test your model", there's a "PREVIEW" section with a table for inputting feature values. The columns are Feature column name, Type, Required or optional, Value, and Local feature. The rows are Time (Numerical, Required, Value 84883), V1 (Numerical, Required, Value 0.0218339018594994), V2 (Numerical, Required, Value 0.0672124249198848), and V3 (Numerical, Required, Value -). To the right of the preview table, there are sections for "Predicted column not yet known" and "Prediction result".

## Part h)

<https://codelabs.developers.google.com/codelabs/automl-forecasting-with-vertex-ai#0>

← → C https://console.cloud.google.com/vertex-ai/datasets?project=haytsack1

Apps Save All Images Subjects Master's Project www.ee.oulu.fi Question Bank - fi...

Google Cloud Platform Haytsack1 Search products and resources

Vertex AI Datasets + CREATE

Dashboard Datasets Features Labeling tasks Workbench Pipelines Training Experiments Models Endpoints Batch predictions Metadata Marketplace

Managed datasets contain data used to train a machine learning model. [Learn more](#)

Region us-central1 (Iowa) Filter Enter a property name

Name	ID	Status	Region	Type	Items	Last updated	Labels
iowa_daily	3395001635502555136	Ready	us-central1	Tabular Forecasting	—	December 4, 2021	
untitled_1637354613458	2386265687715741696	Ready	us-central1	Text	—	November 19, 2021	

https://console.cloud.google.com/vertex-ai/locations/us-central1/datasets/3395001635502555136/analyze?project=haytsack1

Apps Save All Images Subjects Master's Project www.ee.oulu.fi Question Bank - fi...

Google Cloud Platform Haytsack1 Search products and resources

Vertex AI iowa\_daily

Dashboard Datasets Features Labeling tasks Workbench Pipelines Training Experiments Models Endpoints Batch predictions Metadata Marketplace

SOURCE ANALYZE

Dataset info Summary

Created: Dec 04, 2021 12:40 PM Total columns: 4  
Dataset format: CSV Total rows:—  
Dataset location(s): gs://automl-demo-2.../iowa\_daily.csv

Series identifier column:  Timestamp column:

GENERATE STATISTICS

Filter Enter property name or value

Column name	Missing % (count)	Distinct values
ds	—	—
holiday	—	—
id	—	—
y	—	—

Training jobs and models

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

TRAIN NEW MODEL

Google Cloud Platform

**Train new model**

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

Model name \* iowa\_daily\_2021124204347

Target column \* y

Series identifier column \* id

Timestamp column \* ds

Forecasting configuration

Data granularity \* Daily

The granularity level of the timestamp column. Granularity must be the same for all rows. For example, if 'days' is selected, timestamps must be within one day of each other. Data granularity also sets the time period granularity for the forecast horizon and context window.

Forecast horizon \* 7

The number of time periods into the future for which forecasts will be created. Future periods start from the most recent timestamp in the dataset.

Context window \* 7

Defines the input lags to the model for each time series. For most use cases, the context window is between 0-5 times the forecast horizon value. For a starting point, try setting the context window equal to the forecast horizon value. [Learn more](#)

Export test dataset to BigQuery

**Data validation options**

AutoML Forecast can perform multiple validations on your dataset to ensure the best model quality. [?](#)

START TRAINING CANCEL

Google Cloud Platform

**iowa\_daily**

SOURCE ANALYZE

**Dataset Info**

Created: Dec 04, 2021 12:40 PM  
Dataset format: CSV  
Dataset location(s): [gs://automl-demo-2/or/iowa\\_daily.csv](gs://automl-demo-2/or/iowa_daily.csv)

**Summary**

Total columns: 4  
Total rows: -  
Series identifier column: id  
Timestamp column: ds

**Training jobs and models**

iowa\_daily\_2021124204846  
Training model...

**TRAIN NEW MODEL**

**GENERATE STATISTICS**

**Filter** Enter property name or value

Column name ↑	Missing % (count) ?	Distinct values ?
ds	-	-
holiday	-	-
id	-	-
y	-	-

Part i) [https://codelabs.developers.google.com/vertex\\_multiworker\\_training#0](https://codelabs.developers.google.com/vertex_multiworker_training#0)

A screenshot of a Jupyter Notebook interface. The top navigation bar includes 'File', 'Edit', 'View', 'Run', 'Kernel', 'Git', 'Tabs', 'Settings', and 'Help'. A tab labeled 'Dockerfile' is active. The main area shows a code editor with the following Dockerfile content:

```
1 FROM gcr.io/deeplearning-platform-release/tf2-gpu.2-5
2
3 WORKDIR /
4
5 # Copies the trainer code to the docker image.
6 COPY trainer /trainer
7
8 # Sets up the entry point to invoke the trainer.
9 ENTRYPOINT ["python", "-m", "trainer.task"]
10
```

The sidebar on the left shows a file tree with a single file named 'Dockerfile' under the directory '/cassava/'. The bottom status bar indicates 'Ln 10, Col 1 Spaces: 4 Dockerfile'.

A screenshot of a Jupyter Notebook interface. The top navigation bar includes 'File', 'Edit', 'View', 'Run', 'Kernel', 'Git', 'Tabs', 'Settings', and 'Help'. A tab labeled 'task.py' is active. The main area shows a code editor with the following Python script content:

```
1 import tensorflow as tf
2 import tensorflow_datasets as tfds
3 import os
4
5 PER_REPLICA_BATCH_SIZE = 64
6 EPOCHS = 2
7
8 # TODO: replace {your-gcs-bucket} with the name of the Storage bucket you created earlier
9 BUCKET = 'gs://(your-gcs-bucket)/mwms'
10
11 def preprocess_data(image, label):
12     '''Resizes and scales images.'''
13
14     image = tf.image.resize(image, (300,300))
15     return tf.cast(image, tf.float32) / 255., label
16
17
18 def create_dataset(batch_size):
19     '''Loads Cassava dataset and preprocesses data.'''
20
21     data, info = tfds.load(name='cassava', as_supervised=True, with_info=True)
22     number_of_classes = info.features['label'].num_classes
23     train_data = data['train'].map(preprocess_data,
24                                   num_parallel_calls=tf.data.experimental.AUTOTUNE)
25     train_data = train_data.shuffle(1000)
26     train_data = train_data.batch(batch_size)
27     train_data = train_data.prefetch(tf.data.experimental.AUTOTUNE)
28
29     # Set AutoShardPolicy
30     options = tf.data.Options()
31     options.experimental_distribute.auto_shard_policy = tf.data.experimental.AutoShardPolicy.DATA
32     train_data = train_data.with_options(options)
33
34     return train_data, number_of_classes
35
36
37
38 def create_model(number_of_classes):
39     '''Creates and compiles pretrained ResNet50 model.'''
40
41
```

The sidebar on the left shows a file tree with a single file named 'task.py' under the directory '/cassava / trainer/'. The bottom status bar indicates 'Ln 105, Col 11 Spaces: 4 task.py'.

```

(base) jupyter@tensorflow-2-3-20211204-093517:~/horses_or_humans$ cd
(base) jupyter@tensorflow-2-3-20211204-093517:~$ mkdir cassava
(base) jupyter@tensorflow-2-3-20211204-093517:~$ cd cassava
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ touch Dockerfile
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ PROJECT_ID="haystack1"
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ BUCKET="gs://$PROJECT_ID-bucket"
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ gsutil mb -l us-central1 $BUCKET
Creating gs://haystack1-bucket...
ServiceException: 409 A Cloud Storage bucket named 'haystack1-bucket' already exists. Try another name. Bucket names must be globally unique across all Google Cloud projects, including those outside of your organization.
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ gsutil mb -l us-central1 $BUCKET
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ touch trainer/task.py
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ IMAGE_URI="gcr.io/$PROJECT_ID/multiworker:cassava"
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ docker build ./ -t $IMAGE_URI
Sending build context to Docker daemon 12.8kB
Step 1/4 : FROM gcr.io/deeplearning-platform-release/tf2-gpu.2-5
--> 307b41blaec7
Step 2/4 : WORKDIR /
--> Using cache
--> 8b3a03037ef0
Step 3/4 : COPY trainer /trainer
--> 935de06950e9
Step 4/4 : ENTRYPOINT ["python", "-m", "trainer.task"]
--> Running in cc716f9f527b
Removing intermediate container cc716f9f527b
--> c1642c1a406
Successfully built c1642c1a406
Successfully tagged gcr.io/haystack1/multiworker:cassava
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$
```

```

--> Using cache
--> 8b3a03037ef0
Step 3/4 : COPY trainer /trainer
--> 935de06950e9
Step 4/4 : ENTRYPOINT ["python", "-m", "trainer.task"]
--> Running in cc716f9f527b
Removing intermediate container cc716f9f527b
--> c1642c1a406
Successfully built c1642c1a406
Successfully tagged gcr.io/haystack1/multiworker:cassava
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$ docker push $IMAGE_URI
The push refers to repository [gcr.io/haystack1/multiworker]
2a3cc1ddbad2: Pushed
5bb1aa5df10d: Mounted from deeplearning-platform-release/tf2-gpu.2-5
f028010939aa: Mounted from deeplearning-platform-release/tf2-gpu.2-5
dc99c4ea3a81: Mounted from deeplearning-platform-release/tf2-gpu.2-5
37b508e5711b: Mounted from deeplearning-platform-release/tf2-gpu.2-5
756ab564e194: Mounted from deeplearning-platform-release/tf2-gpu.2-5
2ae68080a3d1: Mounted from deeplearning-platform-release/tf2-gpu.2-5
1dccbdcb2fb557: Mounted from deeplearning-platform-release/tf2-gpu.2-5
cfcd8dc2b748: Mounted from deeplearning-platform-release/tf2-gpu.2-5
937abf29c2e1: Mounted from deeplearning-platform-release/tf2-gpu.2-5
5d17b2f192e1: Mounted from deeplearning-platform-release/tf2-gpu.2-5
5d63a86e5e51: Mounted from deeplearning-platform-release/tf2-gpu.2-5
6474a8e117f7: Mounted from deeplearning-platform-release/tf2-gpu.2-5
fe498124ed57: Mounted from deeplearning-platform-release/tf2-gpu.2-5
d545470bb3d: Mounted from deeplearning-platform-release/tf2-gpu.2-5
fb896ef24b4b: Mounted from deeplearning-platform-release/tf2-gpu.2-5
5087113f67c8: Mounted from deeplearning-platform-release/tf2-gpu.2-5
2a92857a1d48: Mounted from deeplearning-platform-release/tf2-gpu.2-5
0ded97864c52: Mounted from deeplearning-platform-release/tf2-gpu.2-5
b50bbacac3e32: Mounted from deeplearning-platform-release/tf2-gpu.2-5
262ea1af4c10: Mounted from deeplearning-platform-release/tf2-gpu.2-5
b420a466ca49: Mounted from deeplearning-platform-release/tf2-gpu.2-5
608c0579d1: Mounted from deeplearning-platform-release/tf2-gpu.2-5
0760cd642691: Mounted from deeplearning-platform-release/tf2-gpu.2-5
75555a1111: Mounted from deeplearning-platform-release/tf2-gpu.2-5
22cf034da6: Mounted from deeplearning-platform-release/tf2-gpu.2-5
8bee4f3fc085: Mounted from deeplearning-platform-release/tf2-gpu.2-5
3b129ca2db46: Mounted from deeplearning-platform-release/tf2-gpu.2-5
64chclal1930ab: Mounted from deeplearning-platform-release/tf2-gpu.2-5
6006ff5a3f1f: Mounted from deeplearning-platform-release/tf2-gpu.2-5
8f8f0266f834: Layer already exists
cassava: digest: sha256:35829f3c34d6861d8d56ab1e1889ca8bb4dfc8b9e8ee09df1fc1d3eb11831dfb size: 6836
(base) jupyter@tensorflow-2-3-20211204-093517:~/cassava$
```

Part j) <https://codelabs.developers.google.com/codelabs/bqml-vertex-prediction#0>

Google Cloud Platform Haytsack1

Search products and resources

FEATURES & INFO SHORTCUT DISABLE EDITOR TABS

Editor

RUN SAVE SCHEDULE MORE

Type to search

Viewing pinned projects.

- haytsack1
- bqpublicdata

Create dataset

Project ID haytsack1 CHANGE

Dataset ID \* Letters, numbers, and underscores allowed

Data location

Default table expiration

Enable table expiration

Default maximum table age Days

Encryption

Google-managed encryption key No configuration required

Customer-managed encryption key (CMEK) Manage via Google Cloud Key Management Service

CREATE DATASET CANCEL

PERSONAL HISTORY PROJECT HISTORY SAVED QUERIES

Google Cloud Platform Haytsack1

Search products and resources

FEATURES & INFO SHORTCUT DISABLE EDITOR TABS

Editor

RUN SAVE SCHEDULE MORE

\*UNSAV... CC\_DEF... COMPOSE NEW QUERY

Type to search

Viewing pinned projects.

- haytsack1
  - cc\_default
    - Models (1)
      - logistic\_model
    - MORE RESULTS
  - bqpublicdata

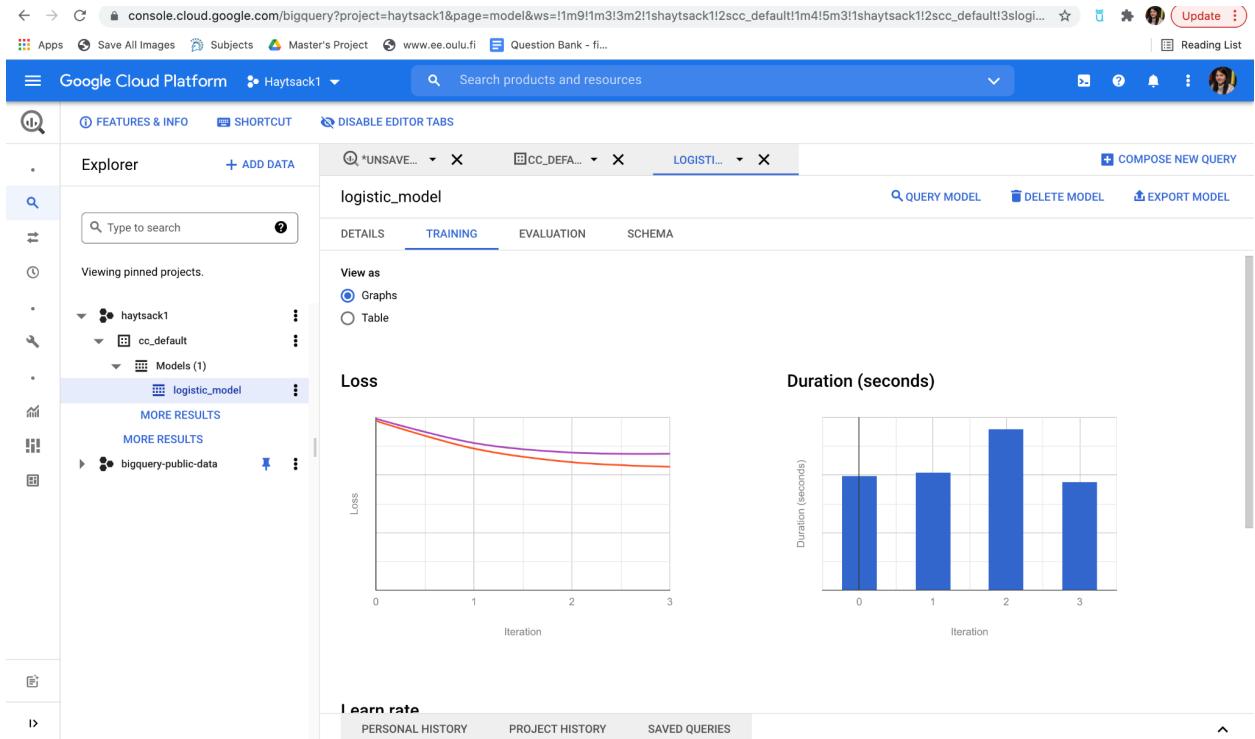
Query results

CREATE OR REPLACE MODEL `haytsack1.cc\_default.logistic\_model` OPTIONS(model\_type='LOGISTIC\_REG', input\_label\_cols=['default\_payment\_next\_month']) AS  
SELECT  
 limit\_balance,  
 sex,  
 education\_level,  
 marital\_status,  
 age,  
 pay\_0,  
 pay\_2,  
 pay\_3,  
 pay\_4,  
 pay\_5,

Query complete (40.0 sec elapsed, 471 KB (ML) processed)

Job information Results Execution details

This statement will create a new model named haytsack1.cc\_default.logistic\_model. Depending on the type of model, this may take several hours to complete. Go to model



Part k) <https://codelabs.developers.google.com/vertex-xgb-wit#0>

The screenshot shows a Jupyter Notebook interface with a terminal window. The terminal window has the title 'Terminal 2' and contains the following pip3 command and its output:

```
(base) jupyter@tensorflow-2-3-20211204-093517:~$ pip3 install xgboost==1.2
Collecting xgboost==1.2
  Downloading xgboost-1.2.0-py3-none-manylinux2010_x86_64.whl (148.9 MB)
    148.9 MB 30 kB/s
Requirement already satisfied: scipy in /opt/conda/lib/python3.7/site-packages (from xgboost==1.2) (1.1.3)
Requirement already satisfied: numpy in /opt/conda/lib/python3.7/site-packages (from xgboost==1.2) (1.19.5)
Installing collected packages: xgboost
Successfully installed xgboost-1.2.0
(base) jupyter@tensorflow-2-3-20211204-093517:~$
```

The left side of the interface shows a file browser with a list of files: 'cassava' (modified 17 minutes ago), 'src' (modified 4 hours ago), 'tutorials' (modified 4 hours ago), and 'Untitled.ipynb' (modified seconds ago).

File Edit View Run Kernel Git Tabs Settings Help

Terminal 2 Untitled.ipynb Python 3

```
[4]: data = pd.read_csv('mortgage-small.csv', index_col=False, dtype=COLUMN_NAMES)
data = data.dropna()
data = shuffle(data, random_state=2)
data.head()
```

	as_of_year	agency_code	loan_type	property_type	loan_purpose	occupancy	loan_amt_thousands	preapproval	county_code	applicant
310650	2016	Consumer Financial Protection Bureau (CFPB)	Conventional (any loan other than FHA, VA, FSA...)	One to four-family (other than manufactured ho...)	Refinancing	1	110.0	Not applicable	119.0	
630129	2016	Department of Housing and Urban Development (HUD)	Conventional (any loan other than FHA, VA, FSA...)	One to four-family (other than manufactured ho...)	Home purchase	1	480.0	Not applicable	33.0	
715484	2016	Federal Deposit Insurance Corporation (FDIC)	Conventional (any loan other than FHA, VA, FSA...)	One to four-family (other than manufactured ho...)	Refinancing	2	240.0	Not applicable	59.0	
887708	2016	Office of the Comptroller of the Currency (OCC)	Conventional (any loan other than FHA, VA, FSA...)	One to four-family (other than manufactured ho...)	Refinancing	1	76.0	Not applicable	65.0	
719598	2016	National Credit Union Administration (NCUA)	Conventional (any loan other than FHA, VA, FSA...)	One to four-family (other than manufactured ho...)	Refinancing	1	100.0	Not applicable	127.0	

[ ]:

Simple 2 1 Python 3 | Idle Mode: Edit Ln 5, Col 2 Untitled.ipynb

File Edit View Run Kernel Git Tabs Settings Help

Terminal 2 Untitled.ipynb Python 3

```
[5]: # Class labels - 0: denied, 1: approved
print(data['approved'].value_counts())

labels = data['approved'].values
data = data.drop(columns=['approved'])

1    665389
0    334610
Name: approved, dtype: int64
```

```
[6]: dummy_columns = list(data.dtypes[data.dtypes == 'category'].index)
data = pd.get_dummies(data, columns=dummy_columns)

data.head()
```

	as_of_year	occupancy	loan_amt_thousands	county_code	applicant_income_thousands	population	ffiec_median_fam_income	tract_to_msa_j
310650	2016	1	110.0	119.0	55.0	5930.0	64100.0	
630129	2016	1	480.0	33.0	270.0	4791.0	90300.0	
715484	2016	2	240.0	59.0	96.0	3439.0	105700.0	
887708	2016	1	76.0	65.0	85.0	3952.0	61300.0	
719598	2016	1	100.0	127.0	70.0	2422.0	46400.0	

5 rows x 44 columns

[ ]:

Simple 2 1 Python 3 | Idle Mode: Edit Ln 1, Col 1 Untitled.ipynb

2147ef85652e67ca-dot-us-west1.notebooks.googleusercontent.com/lab/workspaces/auto-B/tree/Untitled.ipynb

File Edit View Run Kernel Git Tabs Settings Help

Terminal 2 Untitled.ipynb Python 3

```
[6]:
```

	as_of_year	occupancy	loan_amt_thousands	county_code	applicant_income_thousands	population	ffiec_median_fam_income	tract_to_msa_i
310650	2016	1	110.0	119.0	55.0	5930.0	64100.0	
630129	2016	1	480.0	33.0	270.0	4791.0	90300.0	
715484	2016	2	240.0	59.0	96.0	3439.0	105700.0	
887708	2016	1	76.0	65.0	85.0	3952.0	61300.0	
719598	2016	1	100.0	127.0	70.0	2422.0	46400.0	

5 rows x 44 columns

```
[7]: x,y = data.values,labels
x_train,x_test,y_train,y_test = train_test_split(x,y)

[*]: model = xgb.XGBClassifier(
    objective='reg:logistic'
)

[*]: model.fit(x_train, y_train)
y_pred = model.predict(x_test)
acc = accuracy_score(y_test, y_pred.round())
print(acc, '\n')

[*]: model.save_model('model.bst')

[ ]:
```

2147ef85652e67ca-dot-us-west1.notebooks.googleusercontent.com/lab/workspaces/auto-B/tree/Untitled.ipynb

File Edit View Run Kernel Git Tabs Settings Help

Terminal 2 Untitled.ipynb Python 3

```
[7]: x,y = data.values,labels
x_train,x_test,y_train,y_test = train_test_split(x,y)

[10]: model = xgb.XGBClassifier(
    objective='reg:logistic'
)

[11]: model.fit(x_train, y_train)
y_pred = model.predict(x_test)
acc = accuracy_score(y_test, y_pred.round())
print(acc, '\n')

0.873704

[12]: model.save_model('model.bst')

[13]: num_wit_examples = 500
test_examples = np.hstack((x_test[:num_wit_examples],y_test[:num_wit_examples].reshape(-1,1)))
config_builder = (WitConfigBuilder(test_examples.tolist(), data.columns.tolist() + ['mortgage_status'])
    .set_custom_predict_fn(model.predict_proba)
    .set_target_feature('mortgage_status')
    .set_label_vocab(['denied', 'approved']))
WitWidget(config_builder, height=800)
```

Datapoint editor      Performance & Fairness      Features      TypeError('Input data can not be a list.') ⚙️ ?

Visualize	Binning I ...	Binning I ...	Color By	Label By	Scatter I ...	Scatter I ...
<input checked="" type="radio"/> Datapoints <input type="radio"/> Partial dependence plots	(none) <span style="font-size: small;">▼</span>	(none) <span style="font-size: small;">▼</span>	(none) <span style="font-size: small;">▼</span>	(defau) <span style="font-size: small;">▼</span>	(defau) <span style="font-size: small;">▼</span>	(defau) <span style="font-size: small;">▼</span>
Nearest counterfactual <span style="font-size: small;"> ⓘ</span>						
<input checked="" type="radio"/> L1 <input type="radio"/> L2						

Datapoints and their inference results will be displayed here.

## Part I)

```
[1]: import tensorflow as tf
import tensorflow_datasets as tfds
import tensorflow_hub as hub

[ ]: data, info = tfds.load(name='deep_weeds', as_supervised=True, with_info=True)
NUM_CLASSES = info.features['label'].num_classes
DATASET_SIZE = info.splits['train'].num_examples

[ ]: def preprocess_data(image, label):
    image = tf.image.resize(image, (300, 300))
    return tf.cast(image, tf.float32) / 255., label

[ ]: # Create train/validation splits

# Shuffle dataset
dataset = data['train'].shuffle(1000)

train_split = 0.8
val_split = 0.2
train_size = int(train_split * DATASET_SIZE)
val_size = int(val_split * DATASET_SIZE)

train_data = dataset.take(train_size)
train_data = train_data.map(preprocess_data)
train_data = train_data.batch(64)

validation_data = dataset.skip(train_size)
validation_data = validation_data.map(preprocess_data)
validation_data = validation_data.batch(64)

[ ]: feature_extractor_model = "inception_v3"

[ ]: tf_hub_uri = f"https://tfhub.dev/google/imagenet/{feature_extractor_model}/feature_vector/5"
```

```
[ ]: # Shuffle dataset
dataset = data['train'].shuffle(1000)

train_split = 0.8
val_split = 0.2
train_size = int(train_split * DATASET_SIZE)
val_size = int(val_split * DATASET_SIZE)

train_data = dataset.take(train_size)
train_data = train_data.map(preprocess_data)
train_data = train_data.batch(64)

validation_data = dataset.skip(train_size)
validation_data = validation_data.map(preprocess_data)
validation_data = validation_data.batch(64)

[ ]: feature_extractor_model = "inception_v3"

[ ]: tf_hub_uri = f"https://tfhub.dev/google/imagenet/{feature_extractor_model}/feature_vector/5"

[ ]: feature_extractor_layer = hub.KerasLayer(
    tf_hub_uri,
    trainable=False)

[ ]: model = tf.keras.Sequential([
    feature_extractor_layer,
    tf.keras.layers.Dense(units=NUM_CLASSES)
])

[ ]: model.compile(
    optimizer=tf.keras.optimizers.Adam(),
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
    metrics=['acc'])

model.fit(train_data, validation_data=validation_data, epochs=20)
```

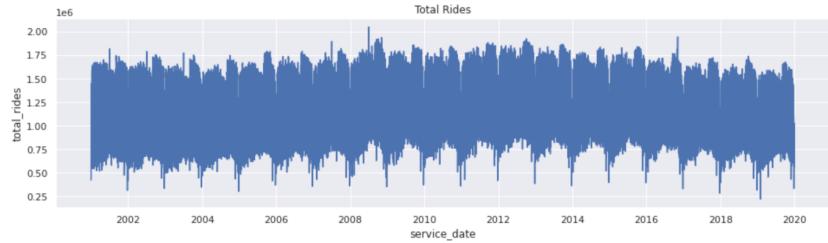
Part m)

<https://codelabs.developers.google.com/codelabs/time-series-forecasting-with-cloud-ai-platform#0>

git clone https://github.com/GoogleCloudPlatform/training-data-analyst  
 Cloning into 'training-data-analyst'...  
 remote: Enumerating objects: 100% (376/376), done.  
 remote: Counting objects: 100% (247/247), done.  
 remote: Compressing objects: 100% (147/147), done.  
 remote: Total 54506 (delta 147), reused 266 (delta 83), pack-reused 54130  
 Receiving objects: 100% (54506/54506), 627.26 MiB | 31.69 MiB/s, done.  
 Resolving deltas: 100% (34524/34524), done.  
 Checking out files: 100% (12334/12334), done.  
 (base) jupyter@tensorflow-2-3-20211204-093517:~\$

[9]: # Initialize plotting  
 register\_matplotlib\_converters() # Addresses a warning  
 sns.set(rc={'figure.figsize':(16,4)})

[10]: # Explore total rides over time  
 sns.lineplot(data=df, x=df.index, y=df[target]).set\_title('Total Rides')  
 fig = plt.show()



[11]: # Explore rides by day type: Weekday (W), Saturday (A), Sunday/Holiday (U)  
 sns.lineplot(data=df, x=df.index, y=df[target], hue=df['day\_type']).set\_title('Total Rides by Day Type')  
 fig = plt.show()



