Serverless Machine Learning v1.3

29 hours 39 minutes
7 Credits

Rate Lab

Overview

In this lab, you go from exploring a taxicab dataset to training and deploying a high-accuracy distributed model with Cloud ML Engine.

Objectives

In these labs, you will perform the following tasks:

Lab 1: Explore dataset, create ML datasets, create benchmark

- Sample the dataset and create training, validation, and testing datasets for local development of TensorFlow models
- Create a benchmark to evaluate the performance of ML

Lab 2: Getting Started with TensorFlow

- Explore the TensorFlow Python API
- Building a graph
- Running a graph
- Feeding values into a graph
- Find area of a triangle using TensorFlow

Lab 3: Machine Learning using tf.estimator

- Read from Pandas Dataframe into tf.constant
- Create feature columns for estimator
- Linear Regression with tf.Estimator framework
- Deep Neural Network regression
- Benchmark dataset

Lab 4: Refactoring to add batching and feature-creation

- Refactor the input
- Refactor the way the features are created
- Create and train the model
- Evaluate model

Lab 5: Distributed training and monitoring

- Create features out of input data
- Train and evaluate
- Monitor with Tensorboard

Lab 6: Scaling up ML using Cloud ML Engine

- Package up the code
- Find absolute paths to data
- Run the Python module from the command line

- Run locally using gcloud
- Submit training job using gcloud
- Deploy model
- Prediction
- Train on a larger dataset
- 1-million row dataset

Lab 7: Feature Engineering

- Working with feature columns
- Adding feature crosses in TensorFlow
- Reading data from BigQuery
- Creating datasets using Dataflow
- Using a wide-and-deep model

Lab 8: Demonstration of Hyper-Parameter Tuning and Training

Setup

What you'll need

To complete this lab, you'll need:

- Access to a standard internet browser (Chrome browser recommended).

- Time. Note the lab's **Completion** time in Qwiklabs. This is an estimate of the time it should take to complete all steps. Plan your schedule so you have time to complete the lab. Once you start the lab, you will not be able to pause and return later (you begin at step 1 every time you start a lab).

- The lab's **Access** time is how long your lab resources will be available. If you finish your lab with access time still available, you will be able to explore the Google Cloud Platform or work on any section of the lab that was marked "if you have time". Once the Access time runs out, your lab will end and all resources will terminate.

- You **DO NOT** need a Google Cloud Platform account or project. An account, project and associated resources are provided to you as part of this lab.

- If you already have your own GCP account, make sure you do not use it for this lab.

- If your lab prompts you to log into the console, **use only the student account provided to you by the lab**. This prevents you from incurring charges for lab activities in your personal GCP account.

Start your lab

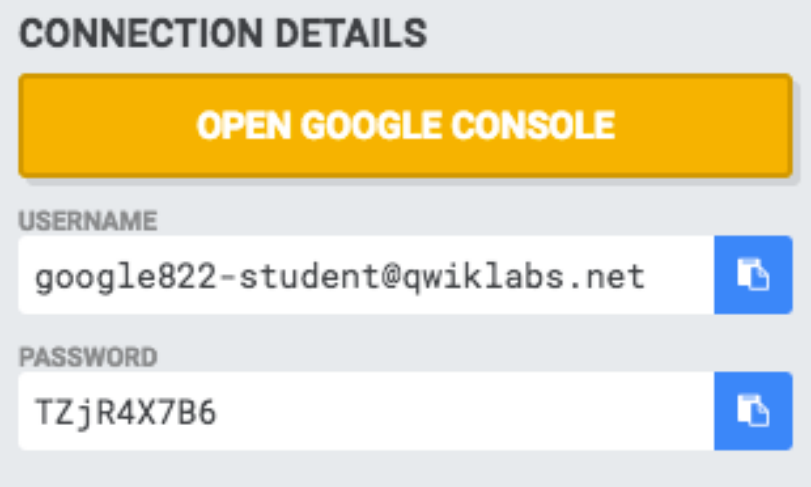
When you are ready, click **Start Lab**. You can track your lab's progress with the status bar at the top of your screen.

Important What is happening during this time? Your lab is spinning up GCP

resources for you behind the scenes, including an account, a project, resources within the project, and permission for you to control the resources needed to run the lab. This means that instead of spending time manually setting up a project and building resources from scratch as part of your lab, you can begin learning more quickly.

Find Your Lab's GCP Username and Password

To access the resources and console for this lab, locate the Connection Details panel in Qwiklabs. Here you will find the account ID and password for the account you will use to log in to the Google Cloud Platform:

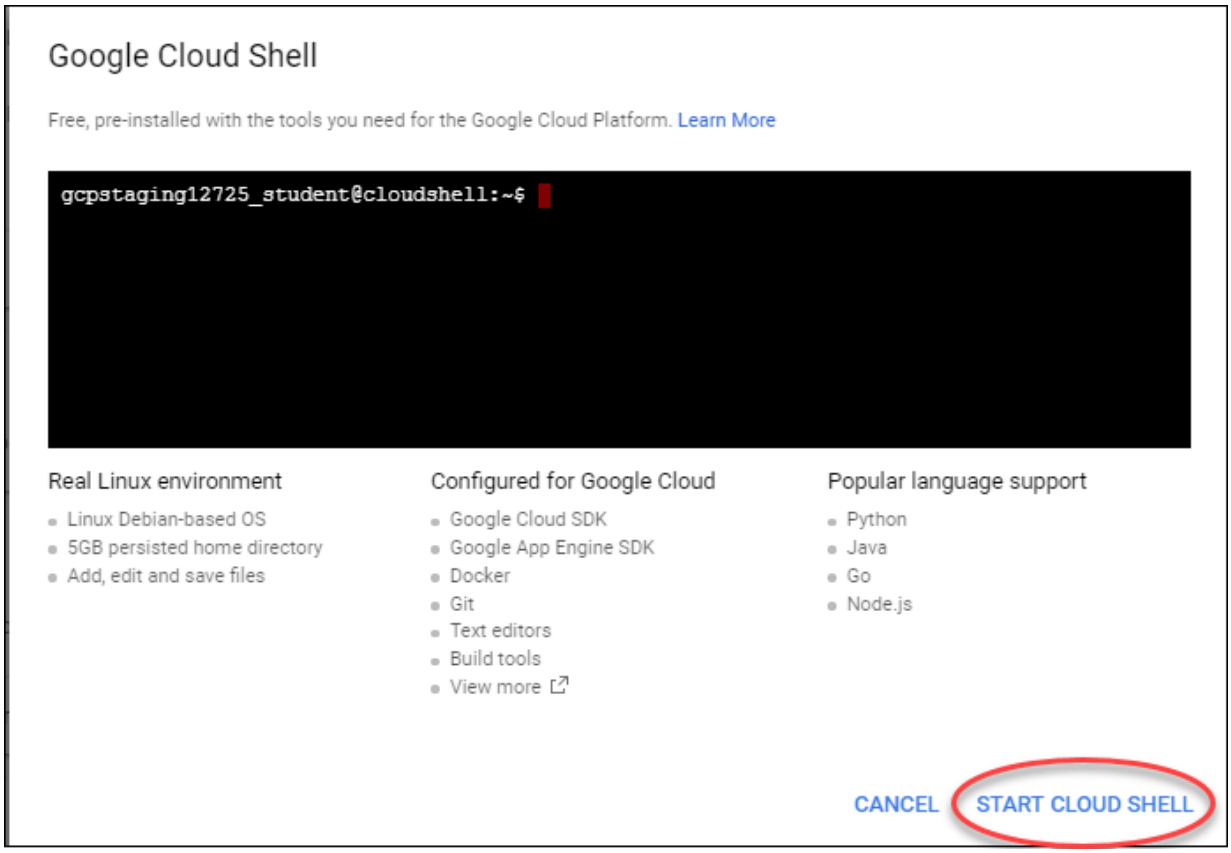


If your lab provides other resource identifiers or connection-related information, it will appear on this panel as well.

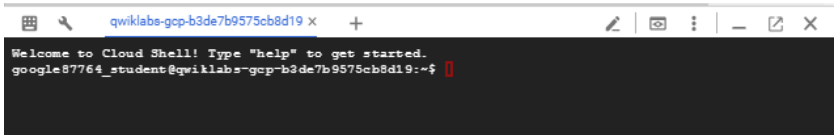
Activate Google Cloud Shell

Google Cloud Shell provides command-line access to your GCP resources.
From the GCP Console click the **Cloud Shell** icon on the top right toolbar:

Then click **START CLOUD SHELL**:



You can click **START CLOUD SHELL** immediately when the dialog comes up instead of waiting in the dialog until the Cloud Shell provisions.
It takes a few moments to provision and connects to the environment:



The Cloud Shell is a virtual machine loaded with all the development tools you'll need. It offers a persistent 5GB home directory, and runs on the Google Cloud, greatly enhancing network performance and authentication.
Once connected to the cloud shell, you'll see that you are already authenticated and the project is set to your *PROJECT_ID*:
gcloud auth list

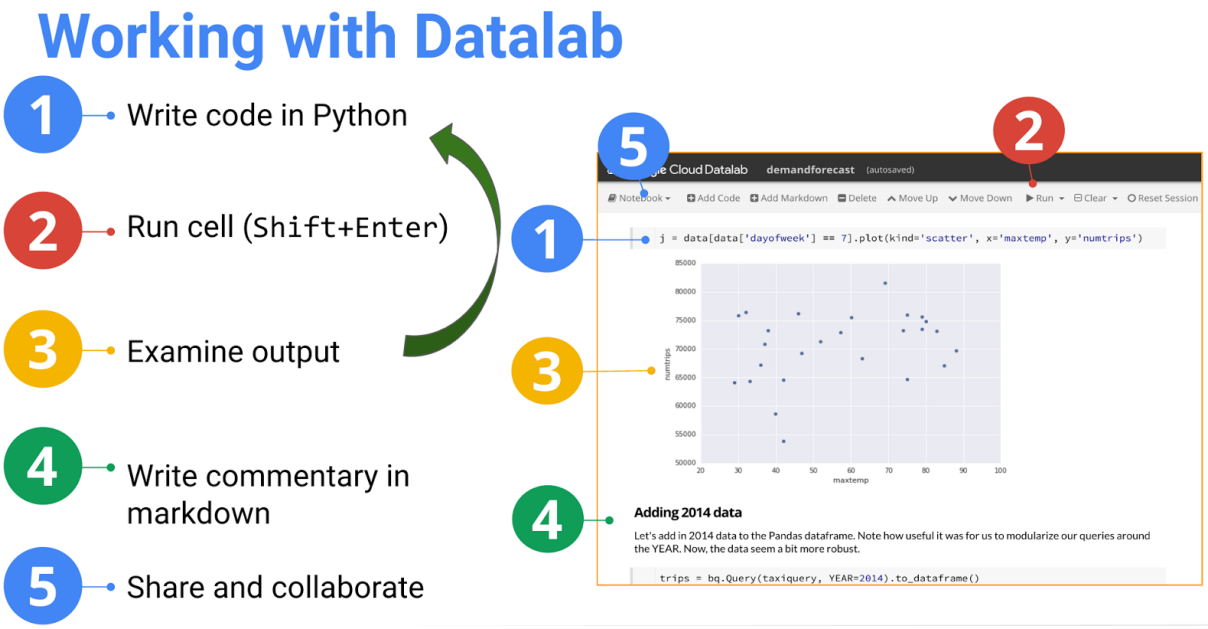
Output:
Credentialed accounts:
 - <myaccount>@<mydomain>.com (active)

Note: gcloud is the powerful and unified command-line tool for Google Cloud Platform. Full documentation is available on Google Cloud gcloud Overview. It comes pre-installed on Cloud Shell and supports tab-completion.
gcloud config list project

Output:
[core]
project = <PROJECT_ID>

Setup Datalab

You will be using Datalab. The following is a graphical cheat sheet for the main Datalab functionality:



- Create and run a query
- Modify the query to add clauses, subqueries, built-in functions and joins.

Task 1. Launch Cloud Datalab

1. Select a zone that is near your location. Enter the following into Cloud Shell to see a list of available zones.

gcloud compute zones list

Please pick a zone in a geographically close region from the following: **us-east1, us-central1, asia-east1, europe-west1**. These are the regions that currently support Cloud ML Engine jobs. Please verify here since this list may have changed after this lab was last updated. For example, if you are in the US, you may choose **us-east1-c** as your zone.

2. Start Datalab:

datalab create dataengvm --zone <ZONE>

Replace <ZONE> with a zone name you picked from the previous step.

3. Datalab setup will prompt you to continue. Enter 'Y'.

Example

Connecting to dataengvm.
This will create an SSH tunnel and may prompt you to create an rsa key pair. To manage these keys, see https://cloud.google.com/compute/docs/instances/adding-removing-ssh-keys
Waiting for Datalab to be reachable at http://localhost:8081/
This tool needs to create the directory [/home/yourprojectid_student/.ssh] before being able to generate SSH keys.
Do you want to continue (Y/n)? Y

4. Datalab setup will ask you for a passphrase. You can press **Enter**twice.

Example

Enter passphrase (empty for no passphrase):
Enter same passphrase again:

5. Datalab will take about five minutes to start. Datalab is ready when you see a message prompting you to do a "Web Preview".

Example

The connection to Datalab is now open and will remain until this command is killed.
Click on the *Web Preview* (square button at top-right), select *Change port > Port 8081*, and start using Datalab.

The connection to your Datalab instance remains open for as long as the datalab command is active. If the cloud shell used for running the datalab command is closed or interrupted, the connection to your Cloud Datalab VM will terminate. If that happens, you may be able to reconnect using the command **datalab connect dataengvm** in your new Cloud Shell.

Task 2. Clone repo into Cloud Datalab

1. Click on the **Web Preview** icon on the top-right corner of the Cloud Shell ribbon.

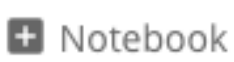
2. Click on **Change Port**.

3. In the **Change Preview Port** dialog, in the **Port Number** box, enter **8081**.

4. Click **Change and Preview**.

Continue in the Cloud Datalab tab

5. In Cloud Datalab home page (browser), open a new notebook using the icon



6. on the top left.

7. In the new notebook, enter the following commands in the cell, and click on Run (on the top navigation bar) to run the commands:

```
%bash
git clone https://github.com/GoogleCloudPlatform/training-data-analyst
cd training-data-analyst
```

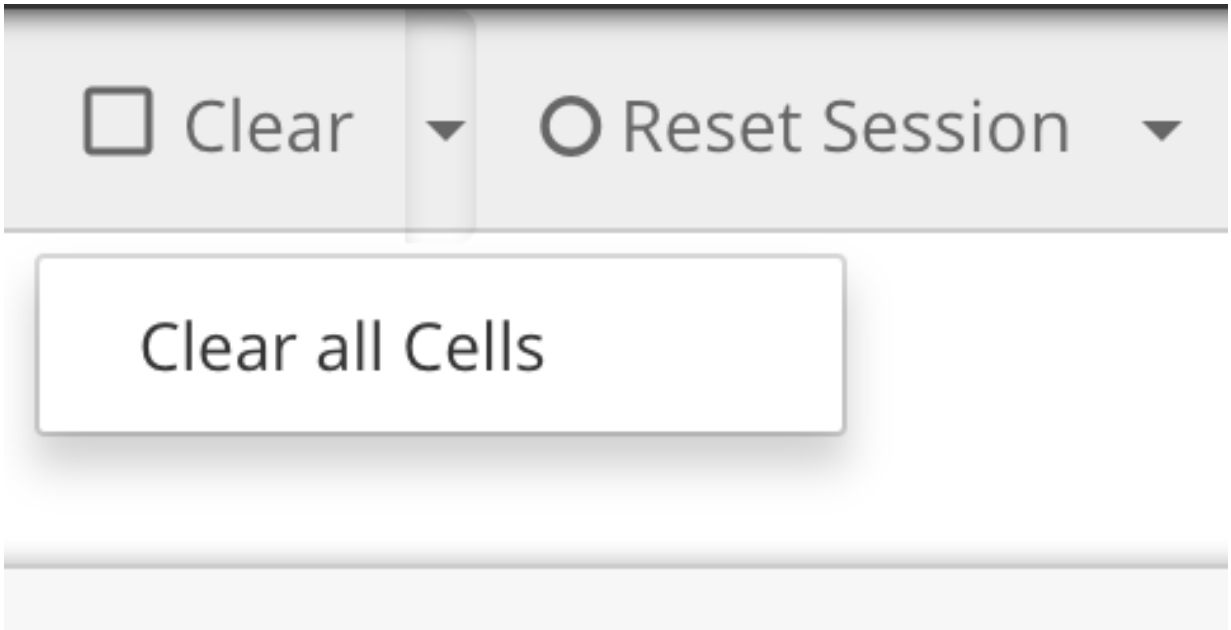
7. Confirm that you have cloned the repo by going back to Datalab browser, and ensure you see the **training-data-analyst** directory.

For the remainder of the labs, you will be loading Datalab python notebooks and performing the lab using the instructions in each notebook.

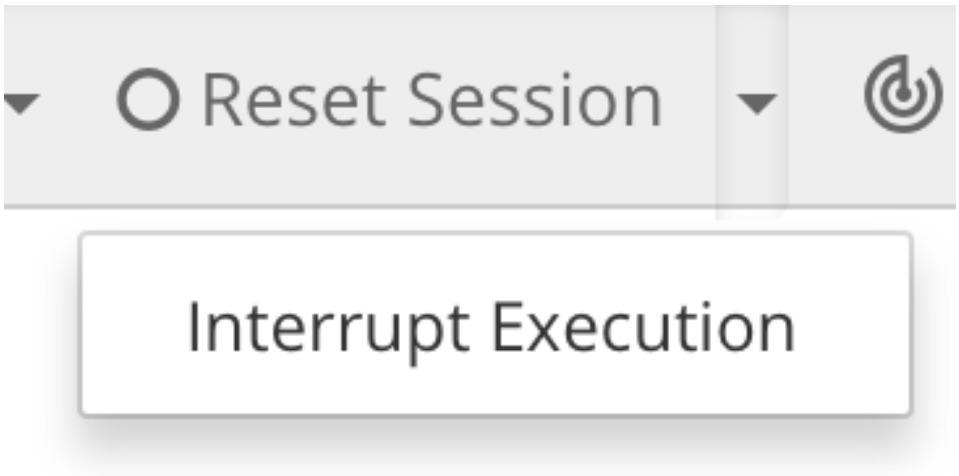
Troubleshooting

What do if a lab hangs or fails:

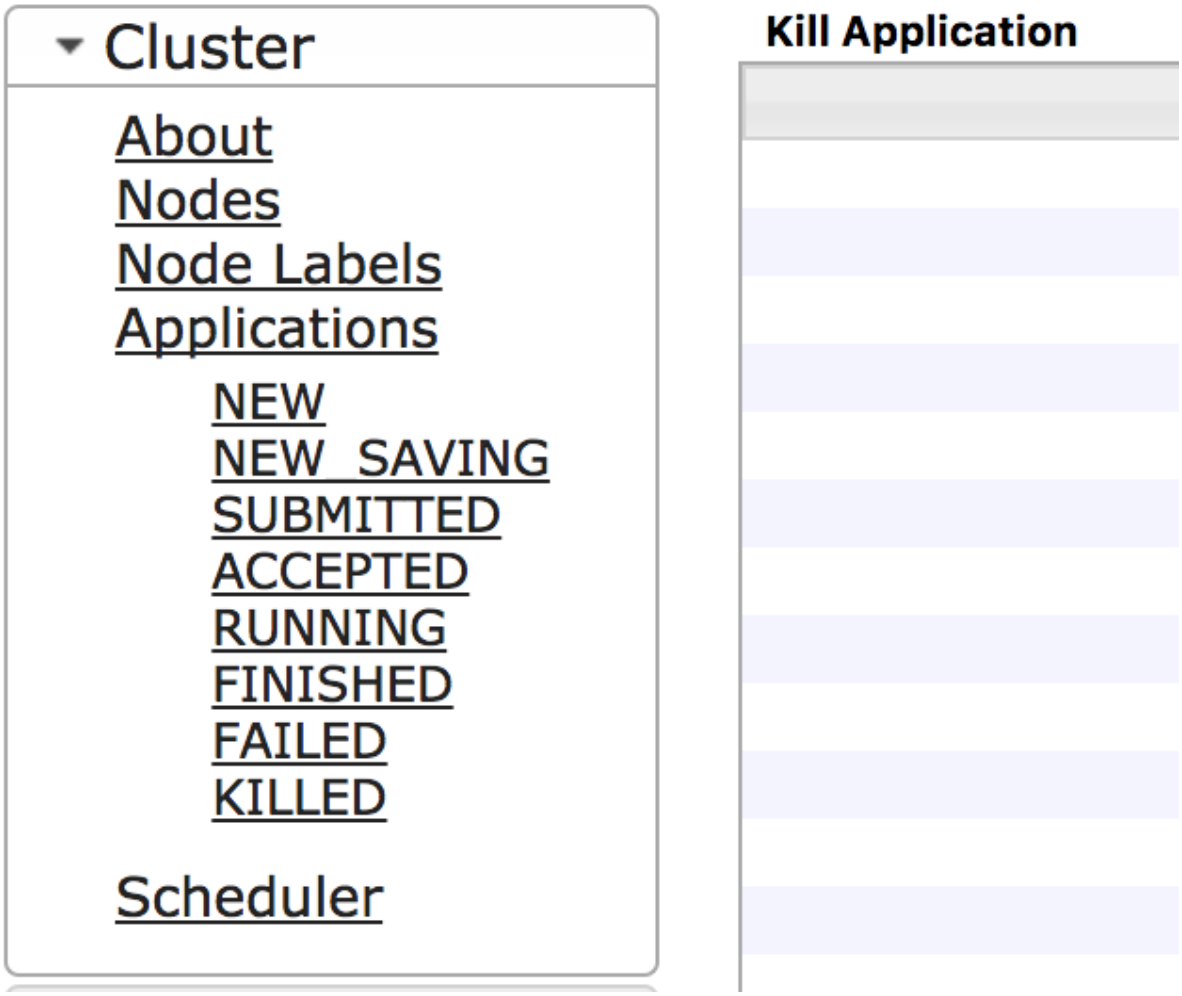
If the lab fails, you can click **Clear** in Datalab to clear the cell output and try again.



In some cases, you may need to reset the Python kernel from within the Datalab. Resetting the kernel causes the job in progress to change state to FINISHED and to have its FinalStatus marked as SUCCEEDED.



If a job is stuck in execution, you can browse to the Hadoop Applications interface, and click on a job that is running. In the upper left corner, there is a link that says "Kill Application".



Task 3: Enable APIs

1. From the GCP console menu, select APIs and services and selectLibrary.

2. For each of the products listed below, search the library and click Enable if they are not already enabled.

- Cloud Machine Learning Engine

- BigQuery API

Lab 1. Explore dataset, create ML datasets, create benchmark

In this lab, you explore a dataset using BigQuery and Datalab.

- Sample the dataset and create training, validation, and testing datasets for local development of TensorFlow models

- Create a benchmark to evaluate the performance of ML

Task 1: Continue the lab in the notebook

1. In Cloud Datalab, click on the Home icon, and then navigate to **datalab/training-data-analyst/courses/machine_learning/datasets/**.

The home icon in Datalab looks like this:

2. Open **create_datasets.ipynb**.

3. In Cloud Datalab, click on **Clear | Clear all Cells** (click on **Clear**, then in the drop-down menu, select **Clear all Cells**).

4. Read the narrative and execute each cell in turn.

Lab 2: Getting Started with TensorFlow

In this lab, you will learn how the TensorFlow Python API works.

- Explore the TensorFlow Python API

- Building a graph

- Running a graph

- Feeding values into a graph

- Find area of a triangle using TensorFlow

Task 1: Continue the lab in the notebook

1. In Cloud Datalab, click on the Home icon, and then navigate to **datalab/training-data-analyst/courses/machine_learning/tensorflow**.

2. Open **a_tfstart.ipynb**.

3. In Cloud Datalab, click on **Clear | Clear all Cells** (click on **Clear**, then in the drop-down menu, select **Clear all Cells**).

4. Read the narrative and execute each cell in turn.

Lab 3: Machine Learning using tf.estimator

In this lab, you will implement a machine learning model using tf.estimator.

- Read from Pandas Dataframe into tf.constant
- Create feature columns for estimator
- Linear Regression with tf.Estimator framework
- Deep Neural Network regression
- Benchmark dataset

Task 1: Continue the lab in the notebook

1. In Cloud Datalab, click on the Home icon, and then navigate to **datalab/training-data-analyst/courses/machine_learning/tensorflow**.

2. Open **b_estimator.ipynb**.

3. In Cloud Datalab, click on **Clear | Clear all Cells** (click on **Clear**, then in the drop-down menu, select **Clear all Cells**).

4. Read the narrative and execute each cell in turn.

Lab 4: Refactoring to add batching and feature-creation

In this lab, you will:

- Refactor the input
- Refactor the way the features are created
- Create and train the model

- Evaluate model

Task 1: Continue the lab in the notebook

1. In Cloud Datalab, click on the Home icon, and then navigate to **datalab/training-data-analyst/courses/machine_learning/tensorflow**.

2. Open **c_batched.ipynb**.

3. In Cloud Datalab, click on **Clear | Clear all Cells** (click on **Clear**, then in the drop-down menu, select **Clear all Cells**).

4. Read the narrative and execute each cell in turn.

Lab 5: Distributed training and monitoring

In this lab, you will learn how to:

- Create features out of input data

- Train and evaluate

- Monitor with Tensorboard

Task 1: Continue the lab in the notebook

1. In Cloud Datalab, click on the Home icon, and then navigate to **datalab/training-data-analyst/courses/machine_learning/tensorflow**.

2. Open **d_traineval.ipynb**.

3. In Cloud Datalab, click on **Clear | Clear all Cells** (click on **Clear**, then in the drop-down menu, select **Clear all Cells**).

4. Read the narrative and execute each cell in turn.

Lab 6: Scaling up ML using Cloud ML Engine

In this lab, you will:

- Package up the code

- Find absolute paths to data

- Run the Python module from the command line

- Run locally using gcloud

- Submit training job using gcloud
- Deploy model
- Prediction
- Train on a larger dataset
- 1-million row dataset

Task 1: Verify that you have a Cloud Storage bucket

The next lab notebook requires you to provide a **Project ID**, **Bucket Name**, and **Bucket Region**.
You can recall the Project ID in Cloud Shell by entering the following:
echo $DEVSHELL_PROJECT_ID

You should have a bucket from the previous lab. If you don't you can follow these instructions to create a bucket.
The bucket must be regional. A Multi-Regional bucket will not work with this lab. If your existing buckets are Multi-Regional, you will need to create a new one for this lab.

1. In the Console, on the **Navigation menu** (

2.), click **Home**.
3. **Select and copy** the Project ID. For simplicity, you will use the Qwiklabs Project ID, which is already globally unique, as the bucket name. If that is already taken, create a globally unique bucket name.
4. In the Console, on the **Navigation menu** (

5.), click **Storage** > **Browser**.
6. Click **Create Bucket**.
7. Specify the following, and leave the remaining settings as their defaults:

Property	**Value** (type value or select option as specified)
Name	<your unique bucket name (Project ID)>

Default storage class	[x] Regional
Location	**<Your location>** **Valid Regions for this lab are** asia-east1, europe-west1, us-central1, us-east1, us-west1

8. 6. Click **Create**.

9. Record the name of your bucket and the location. You will need it in subsequent tasks.

Task 2: Continue the lab in the notebook

1. In Cloud Datalab, click on the Home icon, and then navigate to **datalab/training-data-analyst/courses/machine_learning/cloudmle**.

2. Open **cloudmle.ipynb**.

3. In Cloud Datalab, click on **Clear | Clear all Cells** (click on **Clear**, then in the drop-down menu, select **Clear all Cells**).

4. Read the narrative and execute each cell in turn.

Lab 7: Feature Engineering

In this lab, you will improve the ML model using feature engineering. In the process, you will learn how to:

- Working with feature columns
- Adding feature crosses in TensorFlow
- Reading data from BigQuery
- Creating datasets using Dataflow
- Using a wide-and-deep model

Task 1: Continue the lab in the notebook

1. In Cloud Datalab, click on the Home icon, and then navigate to **datalab/training-data-analyst/courses/machine_learning/feateng**.

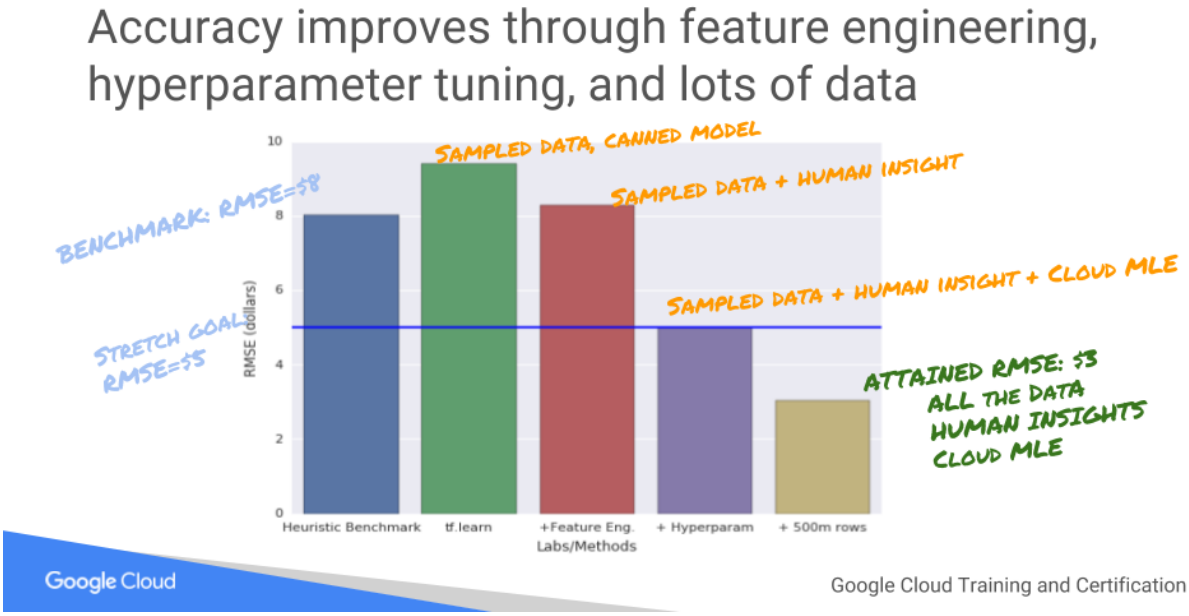
2. Open **feateng.ipynb**.

3. In Cloud Datalab, click on **Clear | Clear all Cells** (click on **Clear**, then in the drop-down menu, select **Clear all Cells**).

4. Read the narrative and execute each cell in turn.

Lab 8: Demonstration of Hyper-Parameter Tuning and Training

Your instructor will demonstrate notebooks that contain hyper-parameter tuning and training on 500 million rows of data. The changes to the model are minor -- essentially just command-line parameters, but the impact on model accuracy is huge.



End your lab

When you have completed your lab, click **End Lab**. Qwiklabs removes the resources you've used and cleans the account for you.

You will be given an opportunity to rate the lab experience. Select the applicable number of stars, type a comment, and then click **Submit**.

The number of stars indicates the following:

- 1 star = Very dissatisfied
- 2 stars = Dissatisfied
- 3 stars = Neutral
- 4 stars = Satisfied
- 5 stars = Very satisfied

You can close the dialog box if you don't want to provide feedback.

For feedback, suggestions, or corrections, please use the **Support** tab.

Last Updated Date: 2018-10-01

Last Tested Date: 2018-10-01

respective companies with which they are associated.