Building a Regression Model in BigQuery for AAPL Stock Data

Overview

BigQuery is Google's fully managed, NoOps, low cost analytics database. With BigQuery you can query terabytes and terabytes of data without having any infrastructure to manage, or needing a database administrator.

BigQuery Machine Learning (BQML) is a new feature in BigQuery where data analysts can create, train, evaluate, and predict with machine learning models with minimal coding.

In this lab, you will build and evaluate a simple linear regression model in BQML to predict AAPL stock prices.

Objectives

In this lab, you learn to perform the following tasks:

- Import a file stored in Google Cloud Storage to BigQuery
- Build a linear regression model in BQML
- Evaluate the model in BigQuery

Set up your environment

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.

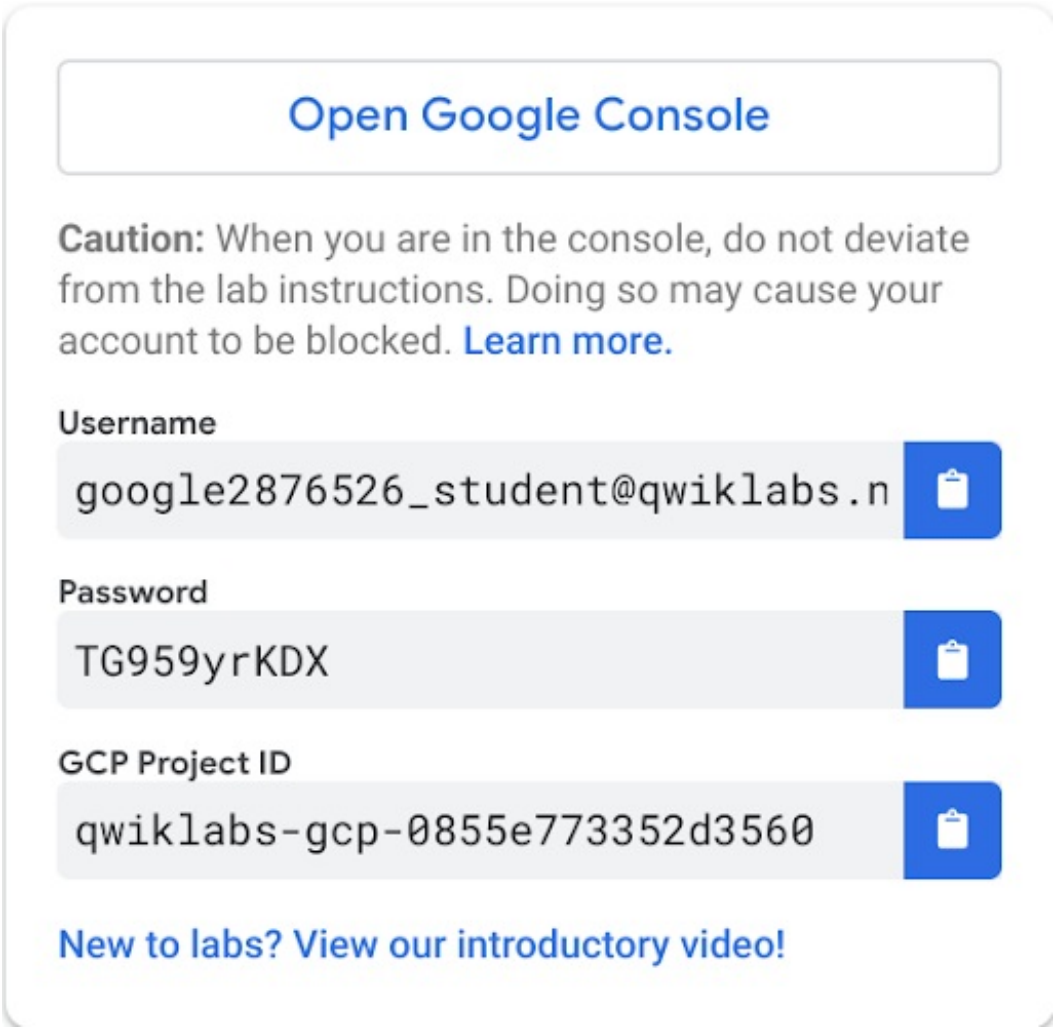
Start your lab

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:

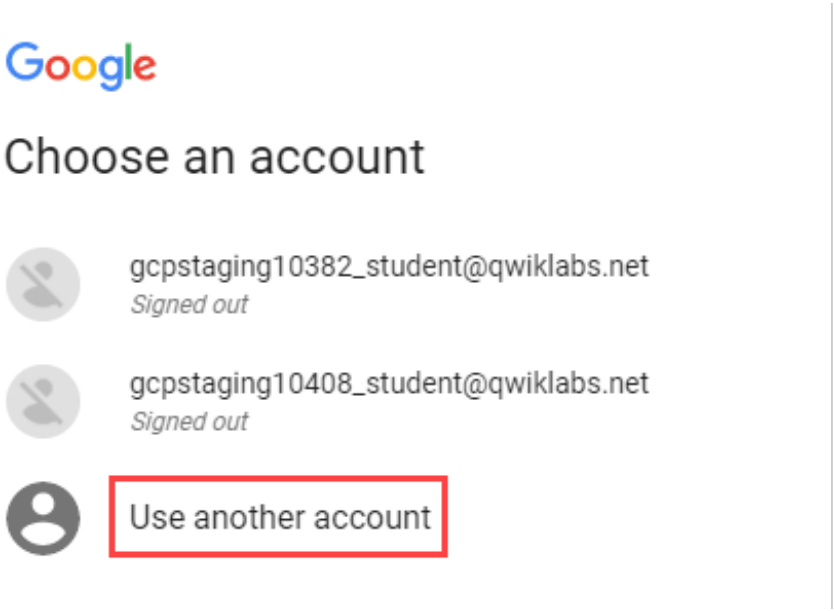


If your lab provides other resource identifiers or connection-related information, it will appear on this panel as well.

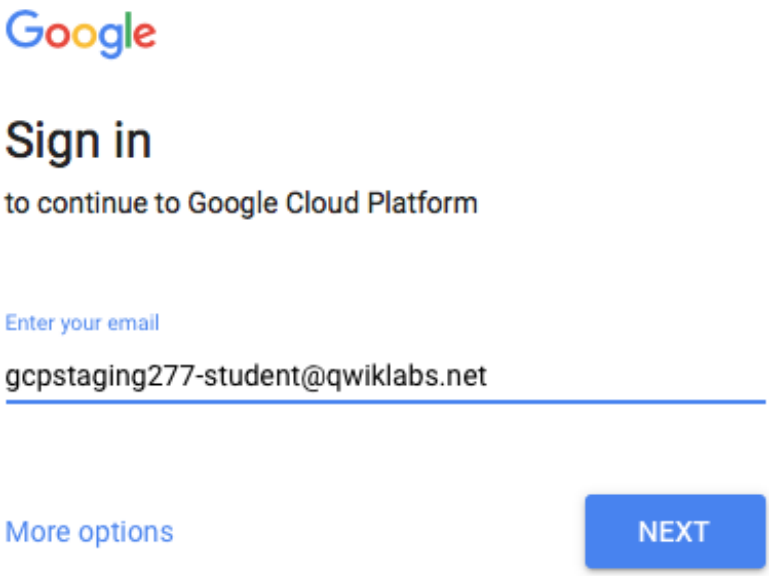
Log in to Google Cloud Console

Using the Qwiklabs browser tab/window or the separate browser you are using for the Qwiklabs session, copy the Username from the Connection Details panel and click the **Open Google Console** button.

You'll be asked to Choose an account. Click **Use another account**.



Paste in the Username, and then the Password as prompted:

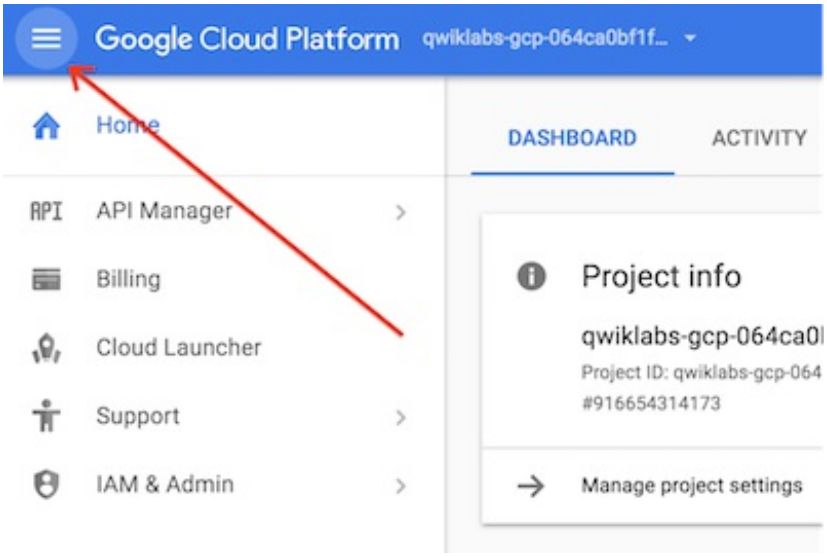


Accept the terms and conditions.

Since this is a temporary account, which you will only have to access for this one lab:

- Do not add recovery options
- Do not sign up for free trials

Note: You can view the list of services by clicking the GCP Navigation menu button at the top-left next to “Google Cloud Platform”.

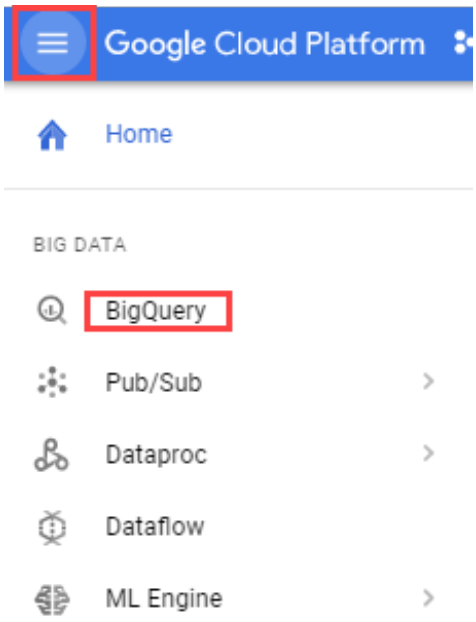


Open BigQuery Console

In the Google Cloud Console, select **Navigation menu** > **BigQuery**:

The **Welcome to BigQuery in the Cloud Console** message box opens. This message box provides a link to the quickstart guide and lists UI updates.

Click **Done**.



Load Data from Google Cloud Storage into BigQuery

1. In the BigQuery navigation menu on the left-hand side select your project id.
2. On the right side click on **CREATE DATASET**.
3. In the menu that results, enter the following values and then click **Create dataset**:

- **Dataset ID**: `ai4f`
- **Default table expiration**: Never
- **Encryption**: Google-managed key

4. Once the dataset is created it will be listed in the navigation menu under your project id. Click on `ai4f`.
5. On the right side click on **CREATE TABLE**.

6. In the menu that results, enter the following values and then click **Create table** (items not specified should be left at their defaults):

- **Create table from**: Google Cloud Storage
- **Select file from GCS bucket**: `cloud-training/ai4f/AAPL10Y.csv`
- **Table name**: AAPL10Y
- **Auto detect**: Schema and input parameters

7. You can view the table's schema by selecting it from the left-hand menu and clicking on the **Schema** tab.

Explore the AAPL Data

Question: What are the minimum and maximum dates present in the dataset?

1. Copy and paste the following SQL code into the Query Editor:

```
SELECT
  MIN(date) AS min_date,
  MAX(date) AS max_date
FROM
  `ai4f.AAPL10Y`
```

2. Click on **Run**. You should receive the following result:

Row	min_date	max_date
1	2009-06-03	2019-06-03

Question: What's the average closing share price for each year?

1. Copy and paste the following SQL code into the Query Editor:

```
SELECT
  EXTRACT(year FROM date) AS year,
  AVG(close) AS avg_close
FROM
  `ai4f.AAPL10Y`
GROUP BY
  year
ORDER BY
  year DESC
```

2. Click on **Run**. You should receive the following result:

Row	year	avg_close
1	2019	180.39666666666665
2	2018	189.05342629482064
3	2017	150.55105577689238
4	2016	104.60400793650794
5	2015	120.03986111111114
6	2014	92.26454999999996
7	2013	67.5192396825397
8	2012	82.29279520000004
9	2011	52.00060039682541
10	2010	37.120328174603216
11	2009	25.025940540540535

Question: Which five dates correspond to the greatest percent increases in AAPL stock?

1. Copy and paste the following SQL code into the Query Editor:

```
SELECT
  date,
  100.0 * close / LAG(close, 1) OVER(ORDER BY date) AS pct_close_change
FROM
  `ai4f.AAPL10Y`
ORDER BY
  pct_close_change DESC
LIMIT
  5
```

2. Click on **Run**. You should receive the following result:

Row	date	pct_close_change
1	2012-04-25	108.87406296851572
2	2014-04-24	108.19816899510835
3	2010-05-10	107.68676007514625
4	2012-11-19	107.2112227639041
5	2018-12-26	107.04215759722125

Build a Linear Regression Model in BigQuery

You will now create a linear regression model in BigQuery to predict the closing price of AAPL stock on any given day. The model will be very simple for the purposes of demonstrating BQML functionality. The only features we'll use as input into the model are the previous day's closing price and a three day trend value. The trend value can only take on two values, either -1 or +1. If the AAPL stock price has increased over any two of the previous three days then the trend will be +1. Otherwise, the trend value will be -1.

1. First, we'll need to generate a table that contains the features to create our regression model. Click on **COMPOSE NEW QUERY**.

2. Copy and paste the following query into the editor window:

```
WITH
  raw AS (
  SELECT
    date,
    close,
    LAG(close, 1) OVER(ORDER BY date) AS min_1_close,
    LAG(close, 2) OVER(ORDER BY date) AS min_2_close,
    LAG(close, 3) OVER(ORDER BY date) AS min_3_close,
    LAG(close, 4) OVER(ORDER BY date) AS min_4_close
  FROM
    `ai4f.AAPL10Y`
  ORDER BY
    date DESC ),
  raw_plus_trend AS (
  SELECT
    date,
    close,
    min_1_close,
    IF (min_1_close - min_2_close > 0, 1, -1) AS min_1_trend,
    IF (min_2_close - min_3_close > 0, 1, -1) AS min_2_trend,
    IF (min_3_close - min_4_close > 0, 1, -1) AS min_3_trend
  FROM
    raw ),
  ml_data AS (
  SELECT
    date,
    close,
    min_1_close AS day_prev_close,
    IF (min_1_trend + min_2_trend + min_3_trend > 0, 1, -1) AS trend_3_day
  FROM
    raw_plus_trend )
SELECT
  *
FROM
  ml_data
```

3. Instead of clicking on **Run**, select the **More** tab then click **Query settings**. In the menu leave all defaults except for the following:

- Select **Set a destination table for query results**.
- **Table name**: model_data

4. Click on **Save**.
5. Run the query by clicking on **Run**.
6. Click on **COMPOSE NEW QUERY**.
7. To build a regression model on the data stored in the table `model_data` execute the following query:

```
CREATE OR REPLACE MODEL `ai4f.aapl_model`
OPTIONS
  ( model_type='linear_reg',
    input_label_cols=['close'],
    data_split_method='seq',
    data_split_eval_fraction=0.3,
    data_split_col='date') AS
SELECT
  date,
  close,
  day_prev_close,
  trend_3_day
FROM
  `ai4f.model_data`
```

Note, this query saves the model into your dataset `ai4f`. The model will be listed along with the tables as `aapl_model`. To evaluate the performance of the model, a sequential split was used in this case, as is common for data with a time element. The split fraction is .3 and split uses the `date` column as the basis for the split.

Evaluate Regression Model Performance

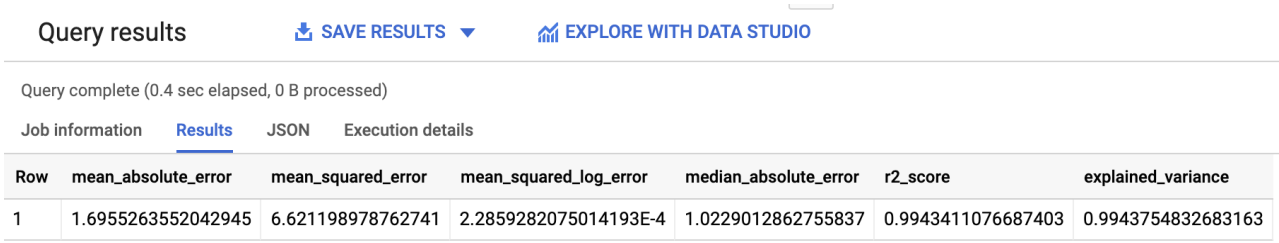
For linear regression models you want to use a loss metric like Root Mean Square Error (RMSE). You want to keep training and improving the model until it has the lowest RMSE while not overfitting.

Note, in BQML, mean_squared_error is a queryable field when evaluating your trained ML model. Add a SQRT() to get RMSE.

1. Click on **COMPOSE NEW QUERY**.
2. To obtain evaluation metrics execute the following query in the editor window:

```
SELECT * FROM ML.EVALUATE(MODEL `ai4f.aapl_model`)
```

Note, the split evaluation data is used to compute the evaluation metrics. Your results should look something like this:



Row	mean_absolute_error	mean_squared_error	mean_squared_log_error	median_absolute_error	r2_score	explained_variance
1	1.6955263552042945	6.621198978762741	2.2859282075014193E-4	1.0229012862755837	0.9943411076687403	0.9943754832683163

Your model RMSE value will vary slightly.

Make Predictions Using Model

1. To make predictions using your trained model enter the following query in the editor window:

```
SELECT
  *
FROM
  ml.PREDICT(MODEL `ai4f.aapl_model`,
    (
    SELECT
      *
    FROM
      `ai4f.model_data`
    WHERE
      date >= '2019-01-01') )
```

This query will generate a new column called `predicted_close` containing the model's predictions for the closing price for all entries in the year 2019.

2. Click on **Run**. Your results should look similar to the following:

Query complete (0.4 sec elapsed, 78.7 KB processed)

Job information | **Results** | JSON | Execution details

Row	predicted_close	date	close	day_prev_close	trend_3_day
1	202.66294728981845	2019-05-09	200.72	202.9	-1
2	195.25620073588766	2019-04-04	195.69	195.35	1
3	205.0396073485694	2019-04-26	204.3	205.28	-1
4	203.02528277667858	2019-04-18	203.86	203.13	1
5	189.86377875384767	2019-04-01	191.24	189.95	1

Congratulations!

You've successfully built and evaluated a linear regression model in BigQuery.

Next Steps / Learn More

- Official documentation for creating ML models in BigQuery: https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-create
- Applying ML to your data on GCP Coursera: https://www.coursera.org/learn/data-insights-gcp-apply-ml
- Getting started with BigQuery ML using the web UI: https://cloud.google.com/bigquery-ml/docs/bigqueryml-web-ui-start

Google Cloud Training & Certification

...helps you make the most of Google Cloud technologies. Our classes include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual options to suit your busy schedule. Certifications help you validate and prove your skill and expertise in Google Cloud technologies.

Manual Last Updated September 4, 2019

Lab Last Tested September 4, 2019