Start Lab 01:15:00

Building a Regression Model in BigQuery for AAPL Stock Data

1 hour 15 minutes Free ★★★★☆

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

For each lab, you get a new Google Cloud project and set of resources for a fixed time at no cost.

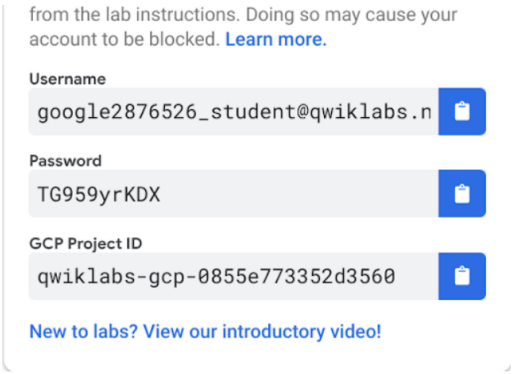
1. Make sure you signed into Qwiklabs using an **incognito window**.
2. Note the lab's access time (for example, 02:00:00 and make sure you can finish in that time block.

   There is no pause feature. You can restart if needed, but you have to start at the beginning.

3. When ready, click START LAB.
4. Note your lab credentials. You will use them to sign in to the Google Cloud Console.

   Open Google Console

   Caution: When you are in the console, do not deviate



5. Click **Open Google Console**.

6. Click **Use another account** and copy/paste credentials for **this** lab into the prompts.

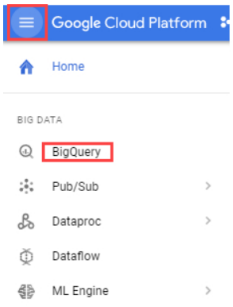
> If you use other credentials, you'll get errors or **incur charges**.

7. Accept the terms and skip the recovery resource page.

> Do not click **End Lab** unless you are finished with the lab or want to restart it. This clears your work and removes the project.

Open BigQuery Console

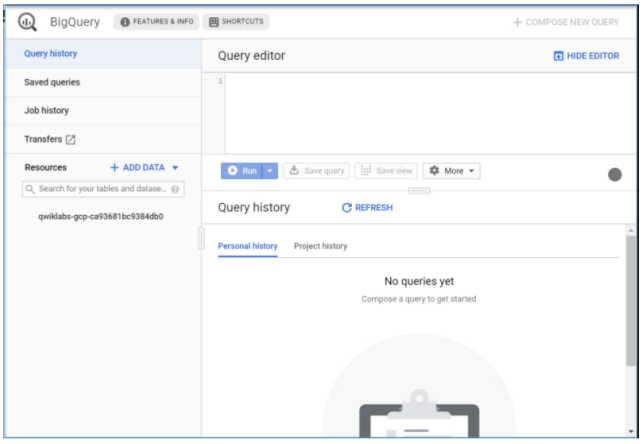
In the Google Cloud Console, select **Navigation menu** > **BigQuery**:



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**.

The BigQuery console opens.



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. Click on **COMPOSE NEW QUERY**. 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. Click on **COMPOSE NEW QUERY**. 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

What does the BigQuery LAG function do?

- ◯ Returns the value of an expression on a subsequent row
- ◯ Identifies the percentile a given row falls into within a partition
- ◯ Assigns a rank to each row within a partition of a result set
- ◯ Returns the value of an expression on a preceding row

Submit

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.

The past 4 values of a stock's closing price are sequentially 80, 81, 74, and 77. What is the trend_3_day value?

- ◯ +1
- ◯ -1

Submit

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:


Query results SAVE RESULTS EXPLORE WITH DATA STUDIO

Query complete (0.4 sec elapsed, 0 B processed)

Job information Results JSON Execution details

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. Click on **COMPOSE NEW QUERY**. 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

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.

©2020 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.