Data Analytics Products (https://cloud.google.com/products/big-data/)
BigQuery ML (https://cloud.google.com/bigquery-ml/docs/)
Documentation (https://cloud.google.com/bigquery-ml/docs/) Guides

Getting started with BigQuery ML using the web UI

This tutorial introduces users to BigQuery ML using the BigQuery web UI.

BigQuery ML enables users to create and execute machine learning models in BigQuery by using SQL queries. The goal is to democratize machine learning by enabling SQL practitioners to build models using their existing tools and to increase development speed by eliminating the need for data movement.

In this tutorial, you use the sample Google Analytics sample dataset for BigQuery (https://support.google.com/analytics/answer/7586738?hl=en&ref_topic=3416089) to create a model that predicts whether a website visitor will make a transaction. For information on the schema of the Analytics dataset, see BigQuery export schema (https://support.google.com/analytics/answer/3437719) in the Google Analytics Help Center.

Objectives

In this tutorial, you use:

- BigQuery ML to create a binary logistic regression model using the `CREATE MODEL` statement
- The `ML.EVALUATE` function to evaluate the ML model
- The `ML.PREDICT` function to make predictions using the ML model

Costs

This tutorial uses billable components of Cloud Platform, including:

- BigQuery
- BigQuery ML

For more information on BigQuery costs, see the BigQuery pricing (https://cloud.google.com/bigquery/pricing) page.

For more information on BigQuery ML costs, see the BigQuery ML pricing (https://cloud.google.com/bigquery-ml/pricing) page.

Before you begin

1. Sign in (https://accounts.google.com/Login) to your Google Account.

   If you don't already have one, sign up for a new account (https://accounts.google.com/SignUp).

2. In the Cloud Console, on the project selector page, select or create a Cloud project.

   Note: If you don't plan to keep the resources that you create in this procedure, create a project instead of selecting an existing project. After you finish these steps, you can delete the project, removing all resources associated with the project.

   GO TO THE PROJECT SELECTOR PAGE (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/PROJECTSELECT

3. Make sure that billing is enabled for your Google Cloud project. Learn how to confirm billing is enabled for your project (https://cloud.google.com/billing/docs/how-to/modify-project).

4. BigQuery is automatically enabled in new projects. To activate BigQuery in a pre-existing project, go to Enable the BigQuery API.

   ENABLE THE API (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/FLOWS/ENABLEAPI?APIID=BIGQUERY)

Step one: Create your dataset

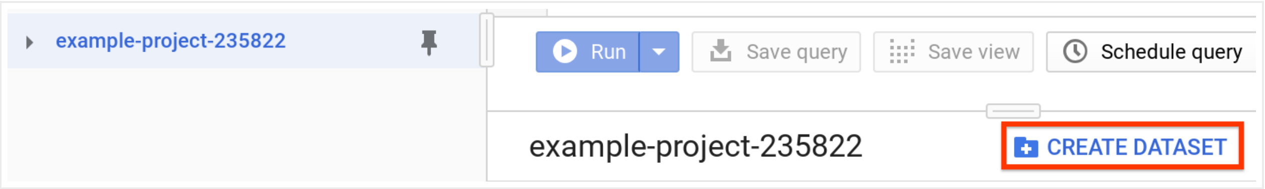
The first step is to create a BigQuery dataset to store your ML model. To create your dataset:

1. Go to the BigQuery web UI in the Cloud Console.

   GO TO THE BIGQUERY WEB UI (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/BIGQUERY)

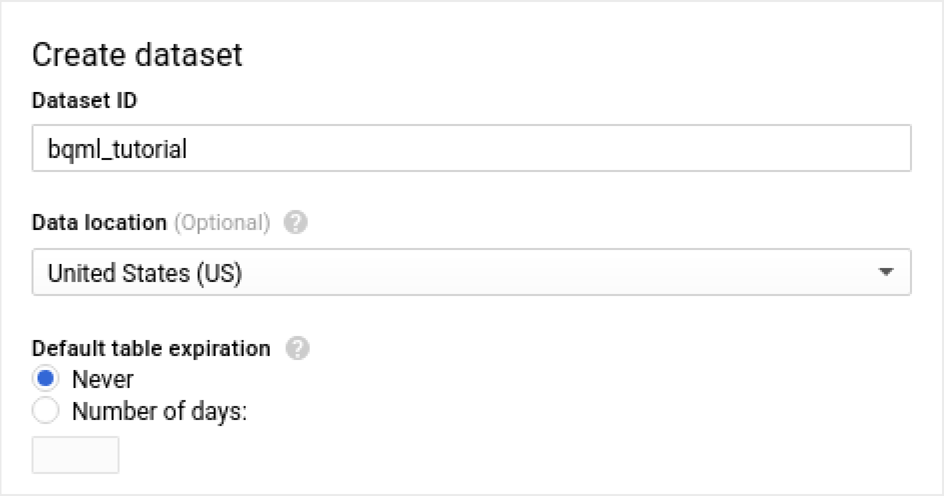
2. In the navigation panel, in the **Resources** section, click your project name.

3. On the right side, in the details panel, click **Create dataset**.



4. On the **Create dataset** page:

   - For **Dataset ID**, enter `bqml_tutorial`.
   - For **Data location**, choose **United States (US)**. Currently, the public datasets are stored in the `US` multi-region location (https://cloud.google.com/bigquery/docs/locations). For simplicity, you should place your dataset in the same location.



Create dataset

Dataset ID

bqml_tutorial

Data location (Optional)

United States (US)

Default table expiration

● Never

○ Number of days:

5. Leave all of the other default settings in place and click **Create dataset**.

Step two: Create your model

Next, you create a logistic regression model using the Google Analytics sample dataset for BigQuery. The following standard SQL query is used to create the model you use to predict whether a website visitor will make a transaction.

```
#standardSQL
CREATE MODEL `bqml_tutorial.sample_model`
OPTIONS(model_type='logistic_reg') AS
SELECT
  IF(totals.transactions IS NULL, 0, 1) AS label,
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(geoNetwork.country, "") AS country,
  IFNULL(totals.pageviews, 0) AS pageviews
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20160801' AND '20170630'
```

In addition to creating the model, running a query that contains the `CREATE MODEL` statement trains the model using the data retrieved by your query's `SELECT` statement.

Query details

The `CREATE MODEL` (https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-create) clause is used to create and train the model named `bqml_tutorial.sample_model`.

The `OPTIONS(model_type='logistic_reg')` clause indicates that you are creating a logistic regression (https://en.wikipedia.org/wiki/Logistic_regression) model. A logistic regression model tries to split input data into two classes and gives the probability the data is in one of the classes. Usually, what you are trying to detect (such as whether an email is spam) is represented by 1 and everything else is represented by 0. If the logistic regression model outputs 0.9, there is a 90% probability the input is what you are trying to detect (the email is spam).

This query's `SELECT` statement retrieves the following columns that are used by the model to predict the probability a customer will complete a transaction:

- `totals.transactions` — The total number of ecommerce transactions within the session. If the number of transactions is `NULL`, the value in the `label` column is set to `0`. Otherwise, it is set to `1`. These values represent the possible outcomes. Creating an alias named `label` is an alternative to setting the `input_label_cols=` option in the `CREATE MODEL` statement.
- `device.operatingSystem` — The operating system of the visitor's device.
- `device.isMobile` — Indicates whether the visitor's device is a mobile device.
- `geoNetwork.country` — The country from which the sessions originated, based on the IP address.
- `totals.pageviews` — The total number of page views within the session.

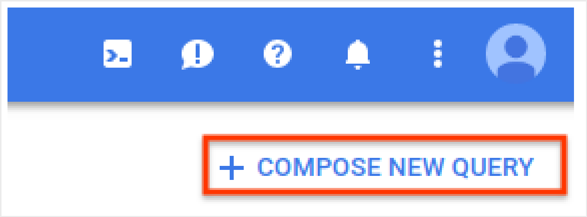
The `FROM` clause — `bigquery-public-data.google_analytics_sample.ga_sessions_*` — indicates that you are querying the Google Analytics sample dataset. This dataset is in the `bigquery-public-data` project. You are querying a set of tables sharded by date. This is represented by the wildcard in the table name: `google_analytics_sample.ga_sessions_*`.

The `WHERE` clause — `_TABLE_SUFFIX BETWEEN '20160801' AND '20170630'` — limits the number of tables scanned by the query. The date range scanned is August 1, 2016 to June 30, 2017.

Run the `CREATE MODEL` query

To run the `CREATE MODEL` query to create and train your model:

1. In the BigQuery web UI, click the **Compose new query** button. If this text is greyed-out, then the **Query editor** is already open.



2. Enter the following standard SQL query in the **Query editor** text area.

```
#standardSQL
CREATE MODEL `bqml_tutorial.sample_model`
OPTIONS(model_type='logistic_reg') AS
SELECT
  IF(totals.transactions IS NULL, 0, 1) AS label,
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(geoNetwork.country, "") AS country,
  IFNULL(totals.pageviews, 0) AS pageviews
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20160801' AND '20170630'
```

3. Click **Run**.

   The query takes several minutes to complete. After the first iteration is complete, your model (`sample_model`) appears in the navigation panel of the BigQuery web UI. Because the query uses a `CREATE MODEL` statement to create a model, you do not see query results.

   You can observe the model as it's being trained by viewing the **Model stats** tab in the BigQuery web UI. As soon as the first iteration completes, the tab is updated. The stats continue to update as each iteration completes.

(Optional) Step three: Get training statistics

To see the results of the model training, you can use the `ML.TRAINING_INFO` (https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-train) function, or you can view the statistics in the BigQuery web UI. In this tutorial, you use the BigQuery web UI.

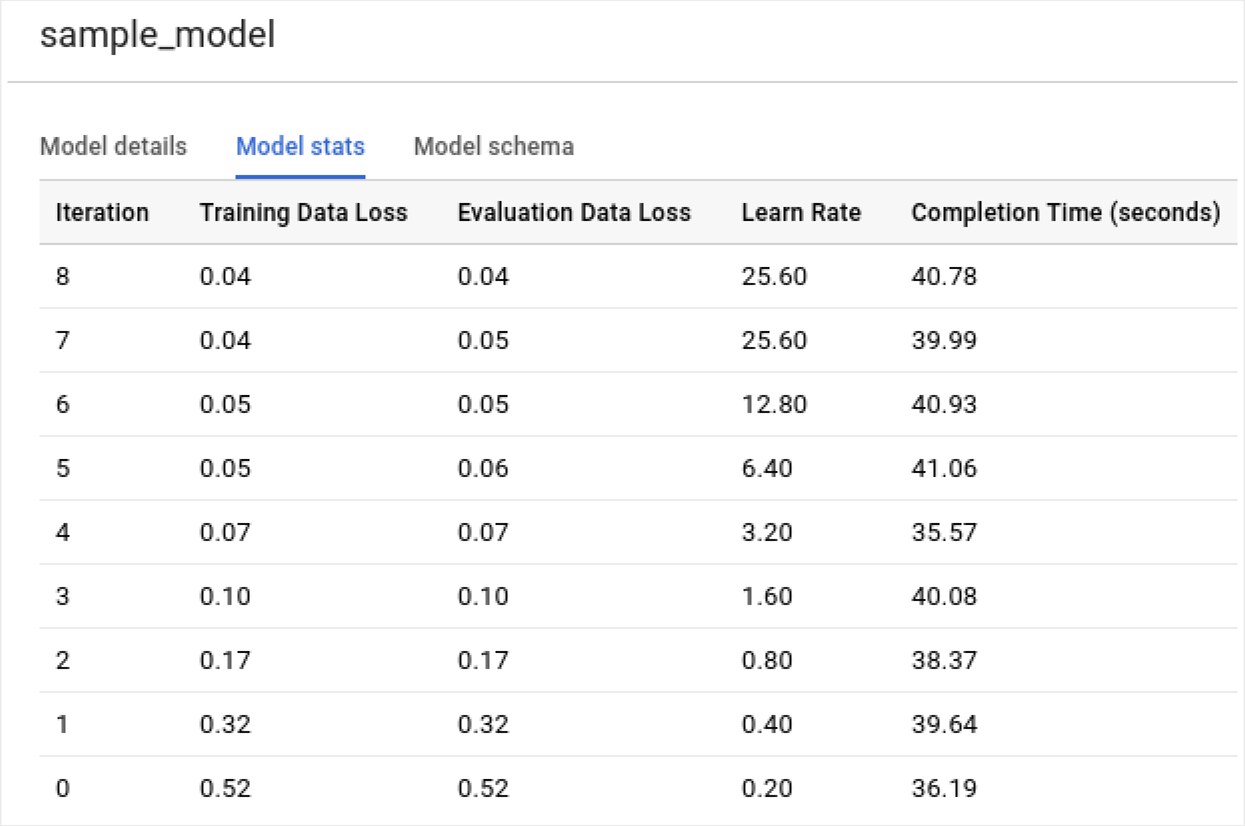
Machine learning is about creating a model that can use data to make a prediction. The model is essentially a function that takes inputs and applies calculations to the inputs to produce an output — a prediction.

Machine learning algorithms work by taking several examples where the prediction is already known (such as the historical data of user purchases) and iteratively adjusting various weights in the model so that the model's predictions match the true values. It does this by minimizing how wrong the model is using a metric called loss.

The expectation is that for each iteration, the loss should be decreasing (ideally to zero). A loss of zero means the model is 100% accurate.

To see the model training statistics that were generated when you ran the `CREATE MODEL` query:

1. In the BigQuery web UI, in the **Resources** section, expand **[PROJECT_ID] > bqml_tutorial** and then click **sample_model**.
2. Click the **Model stats** tab. The results should look like the following:



Iteration	Training Data Loss	Evaluation Data Loss	Learn Rate	Completion Time (seconds)
8	0.04	0.04	25.60	40.78
7	0.04	0.05	25.60	39.99
6	0.05	0.05	12.80	40.93
5	0.05	0.06	6.40	41.06
4	0.07	0.07	3.20	35.57
3	0.10	0.10	1.60	40.08
2	0.17	0.17	0.80	38.37
1	0.32	0.32	0.40	39.64
0	0.52	0.52	0.20	36.19

The **Training Data Loss** column represents the loss metric calculated after the given iteration on the training dataset. Since you performed a logistic regression, this column is the log loss (https://en.wikipedia.org/wiki/Cross_entropy#Cross-entropy_error_function_and_logistic_regression). The **Evaluation Data Loss** column is the same loss metric calculated on the holdout dataset (data that is held back from training to validate the model).

BigQuery ML automatically splits your input data into a training set and a holdout set to avoid overfitting (https://en.wikipedia.org/wiki/Overfitting) the model. This is necessary so that the training algorithm doesn't so closely tailor to the known data that it doesn't generalize to unseen, new examples.

Training Data Loss and Evaluation Data Loss are average loss values, averaged over all examples in the respective sets.

For more details on the `ML.TRAINING_INFO` function, see the BigQuery ML Syntax Reference (https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-train).

Step four: Evaluate your model

After creating your model, you evaluate the performance of the classifier using the `ML.EVALUATE` function. The `ML.EVALUATE` function evaluates the predicted values against the actual data. You can also use the `ML.ROC_CURVE` (https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-roc) function for logistic regression specific metrics.

In this tutorial you are using a binary classification model that detects transactions. The two classes are the values in the `label` column: `0` (no transactions) and `1` (transaction made).

The query used to evaluate the model is as follows:

```
#standardSQL
SELECT
  *
FROM
  ML.EVALUATE(MODEL `bqml_tutorial.sample_model`, (
SELECT
  IF(totals.transactions IS NULL, 0, 1) AS label,
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(geoNetwork.country, "") AS country,
  IFNULL(totals.pageviews, 0) AS pageviews
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20170701' AND '20170801'))
```

Query details

The top-most `SELECT` statement retrieves the columns from your model.

The `FROM` clause uses the `ML.EVALUATE` (https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-evaluate) function against your model: `bqml_tutorial.sample_model`.

This query's nested `SELECT` statement and `FROM` clause are the same as those in the `CREATE MODEL` query.

The `WHERE` clause — `_TABLE_SUFFIX BETWEEN '20170701' AND '20170801'` — limits the number of tables scanned by the query. The date range scanned is July 1, 2017 to August 1, 2017. This is the data you're using to evaluate the predictive performance of the model. It was collected in the month immediately following the time period spanned by the training data.

Run the `ML.EVALUATE` query

To run the `ML.EVALUATE` query that evaluates the model:

1. In the BigQuery web UI, click the **Compose new query** button.
2. Enter the following standard SQL query in the **Query editor** text area.

```
#standardSQL
SELECT
  *
FROM
  ML.EVALUATE(MODEL `bqml_tutorial.sample_model`, (
SELECT
  IF(totals.transactions IS NULL, 0, 1) AS label,
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(geoNetwork.country, "") AS country,
  IFNULL(totals.pageviews, 0) AS pageviews
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20170701' AND '20170801'))
```

3. Click **Run**.
4. When the query is complete, click the **Results** tab below the query text area. The results should look like the following:

```
+--------------------+---------------------+--------------------+-------------
|     precision      |       recall        |      accuracy      |      f1_score
+--------------------+---------------------+--------------------+-------------
| 0.4451901565995526 | 0.08879964301651048 | 0.9716829479411401 | 0.14806547619
+--------------------+---------------------+--------------------+-------------
```

Because you performed a logistic regression, the results include the following columns:

- `precision` (https://developers.google.com/machine-learning/glossary/#precision) — A metric for classification models. Precision identifies the frequency with which a model was correct when predicting the positive class.

- `recall` (https://developers.google.com/machine-learning/glossary/#recall) — A metric for classification models that answers the following question: Out of all the possible positive labels, how many did the model correctly identify?
- `accuracy` (https://developers.google.com/machine-learning/glossary/#accuracy) — Accuracy is the fraction of predictions that a classification model got right.
- `f1_score` (https://en.wikipedia.org/wiki/F1_score) — A measure of the accuracy of the model. The f1 score is the harmonic average of the precision and recall. An f1 score's best value is 1. The worst value is 0.
- `log_loss` (https://en.wikipedia.org/wiki/Cross_entropy#Cross-entropy_error_function_and_logistic_regression) — The loss function used in a logistic regression. This is the measure of how far the model's predictions are from the correct labels.
- `roc_auc` (https://developers.google.com/machine-learning/glossary/#AUC) — The area under the ROC (https://developers.google.com/machine-learning/glossary/#ROC) curve. This is the probability that a classifier is more confident that a randomly chosen positive example is actually positive than that a randomly chosen negative example is positive. For more information, see Classification (https://developers.google.com/machine-learning/crash-course/classification/video-lecture) in the Machine Learning Crash Course.

Step five: Use your model to predict outcomes

Now that you have evaluated your model, the next step is to use it to predict an outcome. You use your model to predict the number of transactions made by website visitors from each country.

The query used to predict the outcome is as follows:

```
#standardSQL
SELECT
  country,
  SUM(predicted_label) as total_predicted_purchases
FROM
  ML.PREDICT(MODEL `bqml_tutorial.sample_model`, (
SELECT
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(totals.pageviews, 0) AS pageviews,
  IFNULL(geoNetwork.country, "") AS country
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20170701' AND '20170801'))
GROUP BY country
ORDER BY total_predicted_purchases DESC
LIMIT 10
```

Query details

The top-most `SELECT` statement retrieves the `country` column and sums the `predicted_label` column. This column is generated by the `ML.PREDICT` function. When you use the `ML.PREDICT` function the output column name for the model is `predicted_<label_column_name>`. For linear regression models, `predicted_label` is the estimated value of `label`. For logistic regression models, `predicted_label` is the most likely label, which in this case is either `0` or `1`.

> **Note:** A more refined use of `ML.PREDICT` would sum the predicted probabilities of each label in the `predicted_label_probs` array. For more informations, see the `ML.PREDICT` (https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-predict) syntax reference.

The `ML.PREDICT` (https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-predict) function is used to predict results using your model: `bqml_tutorial.sample_model`.

This query's nested `SELECT` statement and `FROM` clause are the same as those in the `CREATE MODEL` query.

The `WHERE` clause — `_TABLE_SUFFIX BETWEEN '20170701' AND '20170801'` — limits the number of tables scanned by the query. The date range scanned is July 1, 2017 to August 1, 2017. This is the data for which you're making predictions. It was collected in the month immediately following the time period spanned by the training data.

The `GROUP BY` and `ORDER BY` clauses group the results by country and order them by the sum of the predicted purchases in descending order.

The `LIMIT` clause is used here to display only the top 10 results.

Run the `ML.PREDICT` query

To run the query that uses the model to predict an outcome:

1. In the BigQuery web UI, click the **Compose new query** button.
2. Enter the following standard SQL query in the **Query editor** text area.

```
#standardSQL
SELECT
  country,
  SUM(predicted_label) as total_predicted_purchases
FROM
  ML.PREDICT(MODEL `bqml_tutorial.sample_model`, (
SELECT
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(totals.pageviews, 0) AS pageviews,
  IFNULL(geoNetwork.country, "") AS country
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20170701' AND '20170801'))
GROUP BY country
ORDER BY total_predicted_purchases DESC
LIMIT 10
```

3. Click **Run**.
4. When the query is complete, click the **Results** tab below the query text area. The results should look like the following:

```
+----------------+---------------------------+
|    country     | total_predicted_purchases |
+----------------+---------------------------+
| United States  |                       209 |
| Taiwan         |                         6 |
| Canada         |                         4 |
| Turkey         |                         2 |
| India          |                         2 |
| Japan          |                         2 |
| Indonesia      |                         1 |
| United Kingdom |                         1 |
| Guyana         |                         1 |
+----------------+---------------------------+
```

(Optional) Predict purchases per user

In this example, you try to predict the number of transactions each website visitor will make. This query is identical to the previous query except for the `GROUP BY` clause. Here the `GROUP BY` clause — `GROUP BY fullVisitorId` — is used to group the results by visitor ID.

To run the query:

1. In the BigQuery web UI, click the **Compose new query** button.
2. Enter the following standard SQL query in the **Query editor** text area.

```
#standardSQL
SELECT
  fullVisitorId,
  SUM(predicted_label) as total_predicted_purchases
FROM
  ML.PREDICT(MODEL `bqml_tutorial.sample_model`, (
SELECT
  IFNULL(device.operatingSystem, "") AS os,
  device.isMobile AS is_mobile,
  IFNULL(totals.pageviews, 0) AS pageviews,
  IFNULL(geoNetwork.country, "") AS country,
  fullVisitorId
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20170701' AND '20170801'))
GROUP BY fullVisitorId
ORDER BY total_predicted_purchases DESC
LIMIT 10
```

3. Click **Run**.
4. When the query is complete, click the **Results** tab below the query text area. The results should look like the following:

```
+---------------------+---------------------------+
|    fullVisitorId    | total_predicted_purchases |
+---------------------+---------------------------+
| 9417857471295131045 |                         4 |
| 2158257269735455737 |                         3 |
| 5073919761051630191 |                         3 |
| 7104098063250586249 |                         2 |
| 4668039979320382648 |                         2 |
| 1280993661204347450 |                         2 |
| 7701613595320832147 |                         2 |
| 0376394056092189113 |                         2 |
| 9097465012770697796 |                         2 |
| 4419259211147428491 |                         2 |
+---------------------+---------------------------+
```

Cleaning up

To avoid incurring charges to your Google Cloud Platform account for the resources used in this tutorial:

- You can delete the project you created.
- Or you can keep the project and delete the dataset.

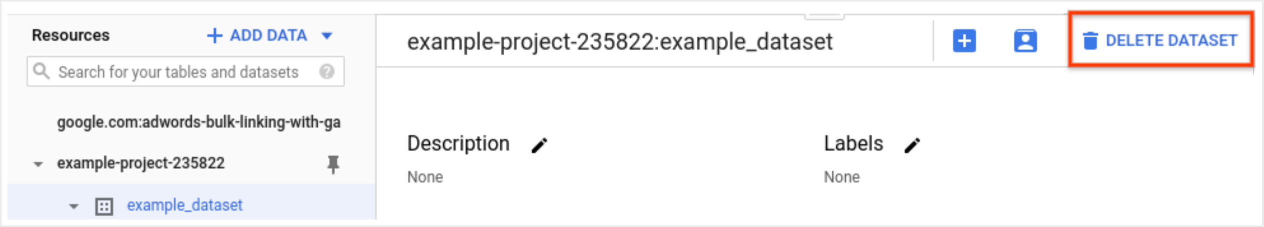
Deleting your dataset

Deleting your project removes all datasets and all tables in the project. If you prefer to reuse the project, you can delete the dataset you created in this tutorial:

1. If necessary, open the BigQuery web UI.

   GO TO THE BIGQUERY WEB UI (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/BIGQUERY)

2. In the navigation, select the **bqml_tutorial** dataset you created.
3. Click **Delete dataset** on the right side of the window. This action deletes the dataset, the table, and all the data.


Resources
+ ADD DATA
Search for your tables and datasets
google.com:adwords-bulk-linking-with-ga
example-project-235822
example_dataset
example-project-235822:example_dataset
DELETE DATASET
Description
None
Labels
None

4. In the **Delete dataset** dialog box, confirm the delete command by typing the name of your dataset (`bqml_tutorial`) and then click **Delete**.

Deleting your project

To delete the project:

Caution: Deleting a project has the following effects:

- **Everything in the project is deleted.** If you used an existing project for this tutorial, when you delete it, you also delete any other work you've done in the project.
- **Custom project IDs are lost.** When you created this project, you might have created a custom project ID that you want to use in the future. To preserve the URLs that use the project ID, such as an `appspot.com` URL, delete selected resources inside the project instead of deleting the whole project.

If you plan to explore multiple tutorials and quickstarts, reusing projects can help you avoid exceeding project quota limits.

1. In the Cloud Console, go to the **Manage resources** page.

   GO TO THE MANAGE RESOURCES PAGE (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/IAM-ADMIN/PRO

2. In the project list, select the project you want to delete and click **Delete** 🗑 .
3. In the dialog, type the project ID, and then click **Shut down** to delete the project.

What's next

- To learn more about machine learning, see the Machine learning crash course (https://developers.google.com/machine-learning/crash-course/).
- For an overview of BigQuery ML, see Introduction to BigQuery ML (https://cloud.google.com/bigquery-ml/docs/bigqueryml-intro).
- To learn more about the BigQuery web UI, see Using the BigQuery web UI (https://cloud.google.com/bigquery/bigquery-web-ui).

Last updated December 4, 2019.