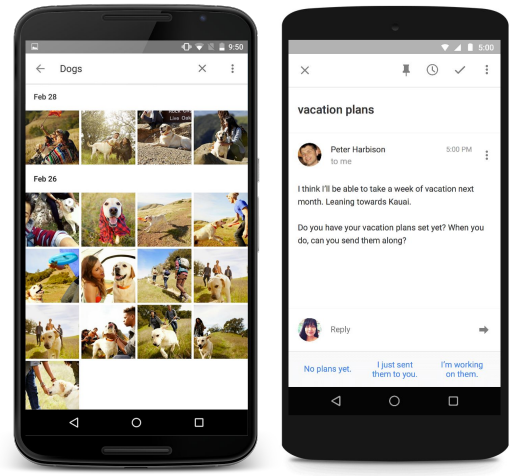
Introduction to BigQuery ML

When you hear “AI or ML,” you probably think of:

Image models

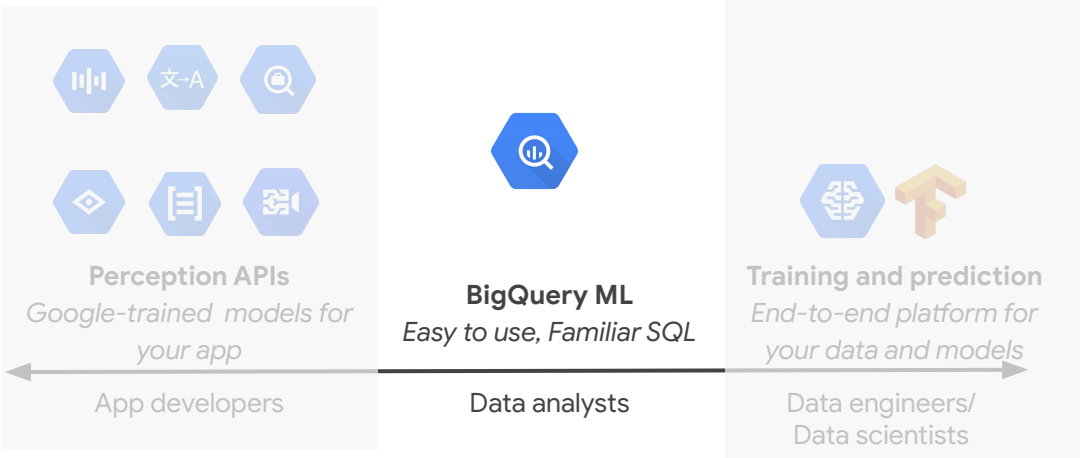
Sequence models

Neural networks



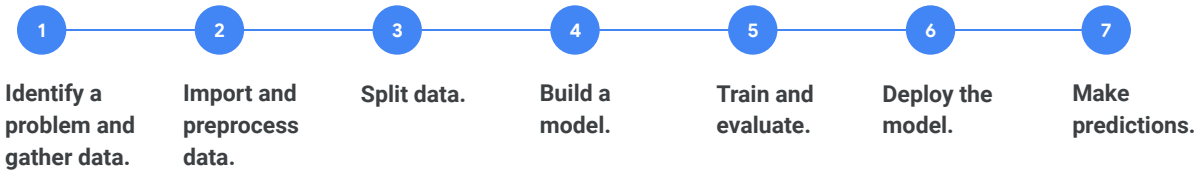
When people think of AI or Machine Learning they generally think of the advanced models like those you saw earlier for Google Photos video stabilization and Smart Reply in Gmail.
But did you know that at Google the majority of models deployed are models that operate on structured data?

BigQuery ML is a way to easily build machine learning models



BQML is middle ground between using pre trained models and rolling your own with TF, Vertex AI.

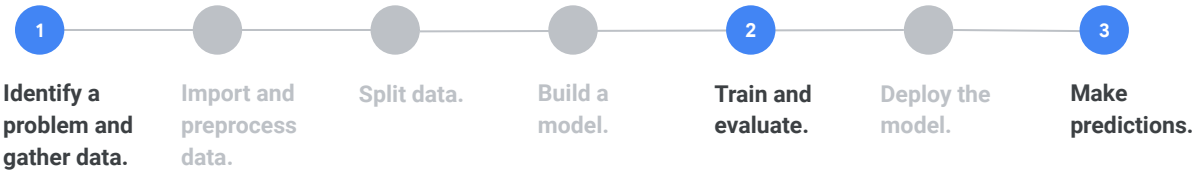
Custom model without BigQuery ML



Custom modeling without BQML

- Increases complexity — Multiple tools are required.
- Reduces speed — Moving and formatting large amounts data for Python-based ML frameworks takes longer than model training in BigQuery.
- Requires multiple steps to export data from the warehouse, restricting the ability to experiment on your data.
- Can be prevented by legal restrictions (such as HIPAA guidelines).

Custom model with BigQuery ML



BQML speeds up time to production, makes development work much easier, and automates of a number of the steps in the ML workflow.

With BQML all you really have to do is have the data in BQ, identify a use case, write a tiny bit of SQL code, and you're good to go.

BigQuery ML

Execute ML models without moving data from BigQuery.

Iterate on models in SQL in BigQuery to increase development speed.

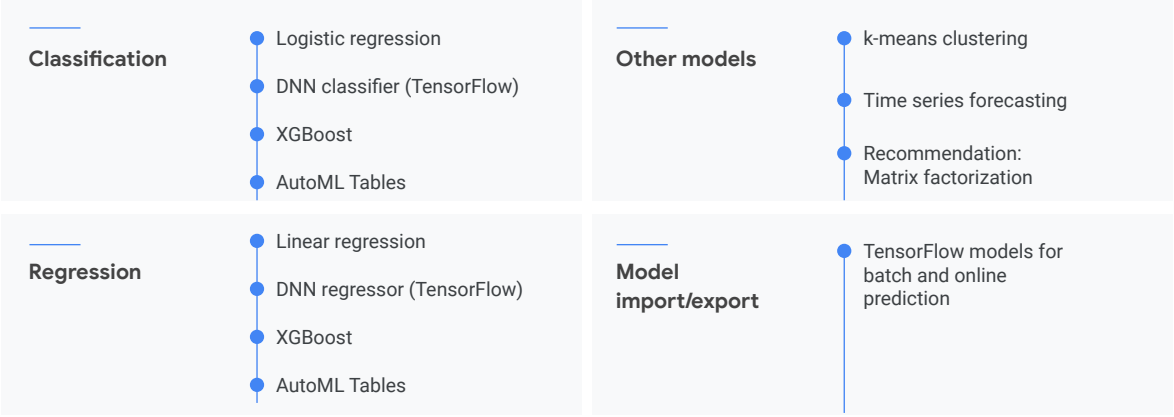
Automate common ML tasks and hyperparameter tuning.

With BQML you can train, evaluate, and serve model predictions without your data leaving BQ itself.

You can automate a lot of the tedious tasks involved with ML.

This enables fast, reliable, and scalable ML experimentation.

Supported BigQuery ML models



BigQuery supports many different model types for classification and regression. And this list is still growing.
Other types of modeling, such as clustering, recommendation systems and forecasting are also available.
Finally you can import your TensorFlow models into BQML for serving predictions/export certain models to use elsewhere.

BigQuery model inference

- ML.PREDICT
- ML.FORECAST
- ML.RECOMMEND
- ML.DETECT_ANOMALIES
- ML.RECONSTRUCTION_LOSS

https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-inference-overview

Machine learning inference in BigQuery ML includes not only machine learning tasks such as label prediction, but also high-level application domains such as forecasting, recommendation, and anomaly detection.

- ML.PREDICT is used to predict outcomes using the model.
- ML.FORECAST forecasts a time series based on a trained time series model.
- ML.RECOMMEND generates a predicted rating for every user-item row combination for a matrix factorization model.
- ML.DETECT_ANOMALIES provides anomaly detection for BigQuery ML.
- ML.RECONSTRUCTION_LOSS compute the reconstruction losses between the input and output data of an Autoencoder model. (Autoencoder model is an unsupervised learning model. https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-e2e-journey#model_creation_phase)

Key phases of a BigQuery ML project

01 Extract, transform, and load data into BigQuery.

02 Select and preprocess features.

03 Create the model inside BigQuery.

04 Evaluate the performance of the trained model.

05 Use the model to make predictions.

BigQuery ML

Here are the five key phases to directly build ML models in BigQuery.

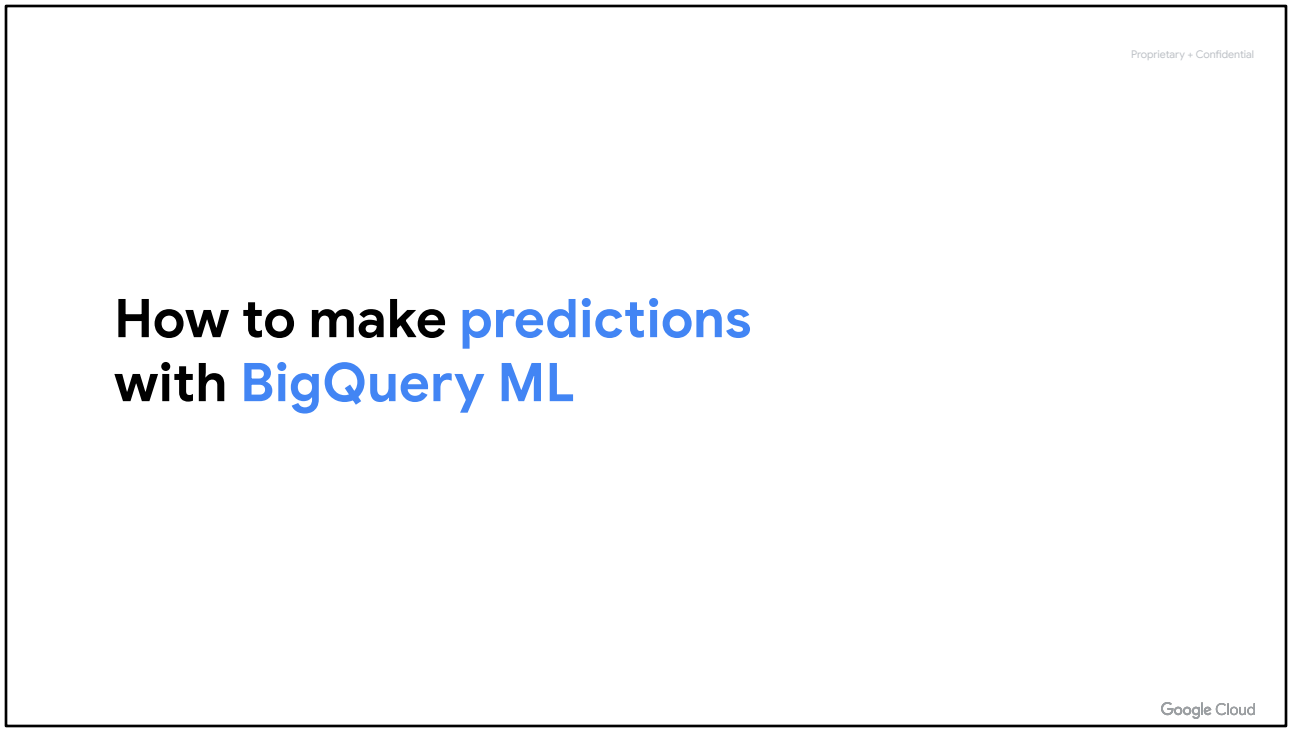
In **phase 1,** you **extract, transform, and load data into BigQuery**, if it isn't there already.

In **phase 2**, you **select and preprocess features**. You can use SQL to create the training dataset for the model to learn from.

In **phase 3**, you **create the ML model** inside BigQuery.

In **phase 4**, after your model is trained, you can execute an ML.EVALUATE query to **evaluate the performance of the trained model** on your evaluation dataset.

And in **phase 5**, the final phase, when you're happy with your model performance, you can then use it to **make predictions**.



Let's start with a simple machine learning function and see how to make predictions.

How to Make Predictions with BigQuery ML

Let's start with a simple machine learning function and see how to make predictions.

Table info

Table ID	nyc-tlc.yellow.trips
Table size	129.72 GB
Long-term storage size	129.72 GB
Number of rows	1,108,779,463

pickup_datetime	dropoff_datetime	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude	rate_code	passenger_count
2010-03-04 00:35:16 UTC	2010-03-04 00:35:47 UTC	-74.035201	40.721548	-74.035201	40.721548	1	1
2010-03-15 17:18:34 UTC	2010-03-15 17:18:35 UTC	0.0	0.0	0.0	0.0	1	1
2015-03-18 01:07:02 UTC	2015-03-18 01:07:07 UTC	0.0	0.0	0.0	0.0	1	5
2015-03-09 18:24:03 UTC	2015-03-09 18:25:37 UTC	-73.93724822998047	40.758201599121094	-73.93726348876953	40.7581901550293	1	1
2010-03-06 06:33:41 UTC	2010-03-06 06:36:06 UTC	-73.785514	40.6454	-73.784564	40.648681	1	2
2013-08-07 00:42:45 UTC	2013-08-07 00:58:43 UTC	-74.025817	40.763044	-74.046752	40.78324	5	1
2015-04-26 02:56:37 UTC	2015-04-26 03:00:01 UTC	-73.98765563964844	40.77165603637695	-73.98755645751953	40.771751403808594	1	1
2015-04-29 18:45:03 UTC	2015-04-29 18:49:01 UTC	0.0	0.0	0.0	0.0	1	1
2010-03-11 21:24:48 UTC	2010-03-11 21:46:51 UTC	-74.571511	40.9108	-74.628928	40.964321	1	1
2013-08-24 01:58:23 UTC	2013-08-24 01:58:23 UTC	-73.972171	40.759439	0.0	0.0	5	4

Taxi photo:
@alexanderredl
https://unsplash.com/photos/S9qxkJN0f4Q

Google Cloud

We have a public dataset of 1.1 billion taxi rides from NYC yellow cabs already loaded into BigQuery.
We will use this dataset to build a model in place in BigQuery.
Our goal will be to train a model to predict the taxi fare for a ride

Select data

Photo from Unsplash

```
SELECT
  fare_amount,
  pickup_longitude,
  pickup_latitude,
  dropoff_longitude,
  dropoff_latitude,
  passenger_count

FROM
  `nyc-tlc.yellow.trips`
```

First let's select the data we want to use for training.
We will select the fare_amount, the label that we are trying to predict, and the pickup location which is latitude and longitude, dropoff location and the passenger count.

https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-create

Build and train with
CREATE MODEL

Photo from Unsplash

```
CREATE OR REPLACE MODEL
  mydataset.model_linreg

OPTIONS(
  input_label_cols=['fare_amount'],
  model_type='linear_reg') AS

SELECT
  fare_amount,
  pickup_longitude,
  pickup_latitude,
  dropoff_longitude,
  dropoff_latitude,
  passenger_count

FROM
  `nyc-tlc.yellow.trips`
```

To create a model, we only need a few more lines of code!
We use the CREATE MODEL statement to create the model in one of our BQ datasets.
The input_label_cols OPTION specifies the column we wish to predict and then we specify the model_type and any of the options associated with that model type.

https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-create

Evaluate with ML.EVALUATE

Photo from Unsplash

```
SELECT
  *

FROM
  ML.EVALUATE(
    MODEL mydataset.model_linreg
    )
```

We can evaluate our models by using the ML.EVALUATE function.
We pass in our model name and the dataset we wish to evaluate on.
https://cloud.google.com/bigquery-ml/docs/reference/standard-sql/bigqueryml-syntax-create

Use the model with ML.PREDICT

Photo from Unsplash

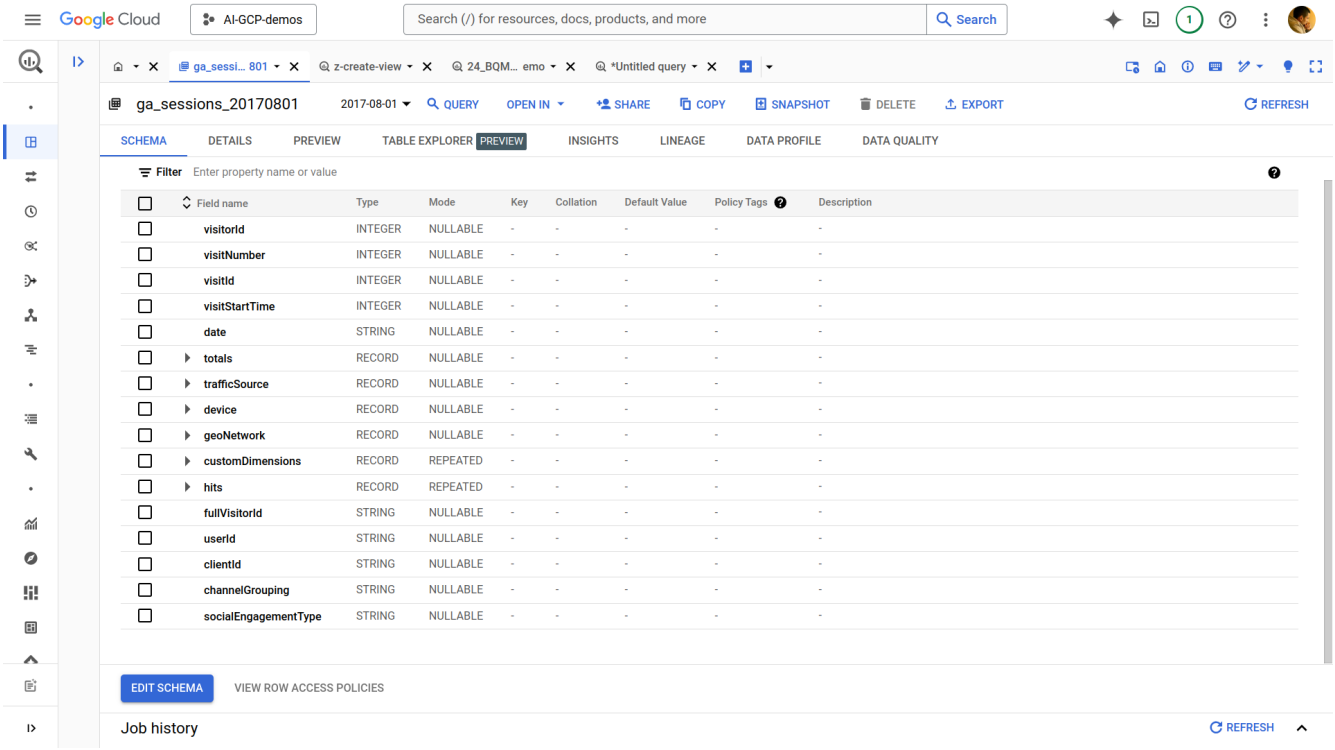
```
SELECT
  *

FROM
  ML.PREDICT(MODEL mydataset.model_linreg,
    (
    SELECT
      fare_amount,
      pickup_longitude,
      pickup_latitude,
      dropoff_longitude,
      dropoff_latitude,
      passenger_count
    FROM
      `nyc-tlc.yellow.trips`
  ))
```

Then we use ML.PREDICT function to predict the outcome of the model.

BQML Example

The Dataset Schema



1) Get the training data

```
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 '20170631'
LIMIT 10000;
```

2) Save the training data in a table called “training_data” in a dataset called “bqml”

3) Create the model

```
CREATE OR REPLACE MODEL `bqml.sample_model`
OPTIONS(model_type='logistic_reg') AS
SELECT * from `bqml.training_data`;
```

4) Evaluate the model

```
SELECT *
FROM ml.EVALUATE(MODEL `bqml.sample_model`);
```

5) Get the prediction data (the data for which we will predict the label)

```
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,
  fullVisitorId
FROM
  `bigquery-public-data.google_analytics_sample.ga_sessions_*`
WHERE
  _TABLE_SUFFIX BETWEEN '20170701' AND '20170801';
```

6) Save the prediction data in a table called “july_data” in a dataset called “bqml”

7) Predict Example

```
SELECT
  country,
  SUM(predicted_label) as total_predicted_purchases
FROM
  ml.PREDICT(
    MODEL `bqml.sample_model`,
    (SELECT * FROM `bqml.july_data`)
  )
GROUP BY country
ORDER BY total_predicted_purchases DESC
LIMIT 10;
```