Tags	AutoML, Automated Machine Learning, Machine Learning, BigQuery, Autom ML with BigQuery ML
Meta Description	Meet Automated Machine Learning with BigQuery ML, as an easy start and validation of Machine Learning. Find out AutoML's applicability examples and limitations.
Intro KB	Meet Automated Machine Learning with BigQuery ML as an easy start to the Machine Learning journey. In this article you will find out AutoML models applicability examples and limitations. Validate whether ML will be an efficient solution to the problems that your organization is facing.
Autor	Michał Bryś - Senior ML Engineer and Technical Product Owner
LINK do grafik	

Automated Machine Learning (AutoML) with BigQuery ML. Start Machine Learning easily and validate if ML is worth investing in or not.

Machine learning is becoming increasingly popular in many industries, from finance to marketing to healthcare. But let's face it, that doesn't mean ML will necessarily be a viable solution for every organization. Will the benefit-cost ratio be satisfactory? On the other hand, it would be reckless not to check out the possibilities of ML. Implementing machine learning is a huge process, generating big costs. In this article, we will present a solution that can be an easy start to a machine learning journey and can help you to validate whether ML will be an efficient solution to the problems that your organization is facing.

In this article, we'll **introduce AutoML models, followed by an example of the AutoML Classifier model trained in BigQuery ML**. We will discuss AutoML's **applicability, limitations and application examples.**

What is the idea of Automated Machine Learning (AutoML)?

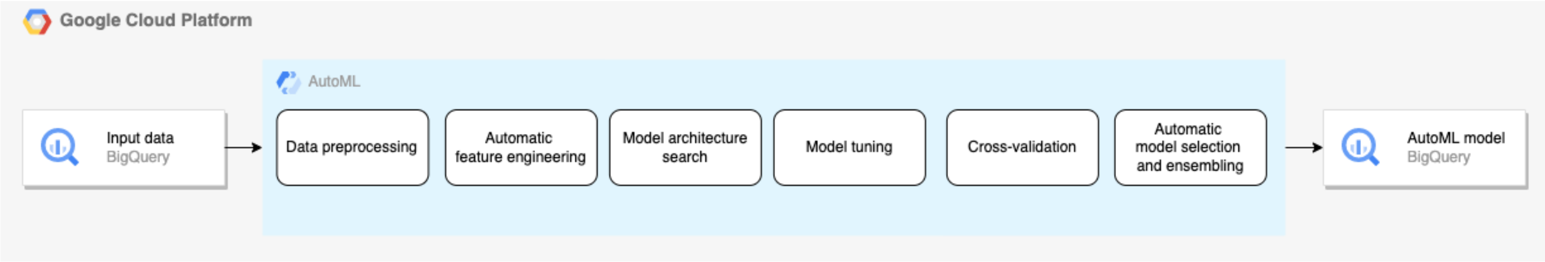
Automated Machine Learning, also called AutoML, is an approach to lower the entry-level into Machine Learning. The main idea is to **automate data preparation, feature engineering,**

model training and model evaluation. As a result, the end user gets a simplified interface to train machine learning models: the raw data as an input and the ready-to-use machine learning model as an output.

AutoML is available through multiple tools and frameworks (i.e., auto-sklearn, AutoKeras, tpot). Since the goal of AutoML is to lower the entry barrier for Machine Learning, in our example we'll leverage the AutoML offered in the Google Cloud Platform as a part of BigQuery ML (BQML). With this serverless setup, we can leverage the AutoML without any complicated configuration and infrastructure maintenance.

Behind the scenes, Auto ML performs the regular Machine Learning workflow:

- data preprocessing,
- automatic feature engineering,
- model architecture search,
- model tuning,
- cross-validation,
- automatic model selection and ensembling.



When are AutoML models useful?

While this black-box approach is not a one-size-fits-all solution to every problem, you may find it helpful to:

- Start working with ML in your organization, even if you don't have enough expertise internally.
- Verify if you can build a valuable ML model using the data you have in place. It can save you lots of time, especially in the prototyping stage by feedback without engaging lots of research work.
- Get the baseline model along with model accuracy. You can use it as a benchmark to evaluate the manually created model's performance.
- Build the proof-of-concept model to see the benefits and possible limitations before you invest more into the target solution.

Automated Machine Learning limitations

- **Dataset size -** the dataset cannot be too big (or too small). In our case using BigQuery, the input data to AutoML models must be between 1000 and 200,000,000 rows and less than 100 GB [source]. This means you can't use it for small data, and you're also limited with the maximum dataset size.
- **Long training** - it takes time to complete the entire workflow including feature engineering and data preprocessing. For example, to use AutoML in BigQuery, you need to set a minimum of 1 hour for the training and you can't set a limit for the rest of the workflow steps [source].
- **Reproducibility and low contro**l **-** because it's a black-box approach, there are many intermediate steps between raw data and the final model, so a slight change in the input data may impact the final result. It may create some hard-to-debug issues with the model stability in time. If you need more control over the AutoML process, you can evaluate Tabular Workflows on Vertex AI.
- **Cost** - when you know how to solve your problem with a suitable ML model, AutoML may not be the most cost-effective way to train it continuously, i.e., daily. [pricing]

However, even with the aforementioned limits, AutoML is still a good approach to start your journey with Machine Learning and check if it's the right solution.

Our test scenario for AutoML in BigQuery

One of the most critical use cases of AutoML is **supervised learning** for **structured data**, including the **features columns** and the **target column** [source]**.**

AutoML in BigQuery offers 2 model types: AutoML Classifier for classification problems and AutoML Regressor for regression problems.

In the following steps, we'll use this scenario to demonstrate how AutoML Classifier (backed by AutoML Tables in Vertex AI) works in BigQuery. We'll build a machine learning **classification model to predict the probability of adding a product to a shopping cart by an e-commerce website user.** The source dataset will be the Google Analytics 4 export to BigQuery (See: A Step-by-Step Guide to Training a Machine Learning Model using BigQueryML (BQML) - GetInData).

Data preparation

First, you need to prepare the source data (see: A Step-by-Step Guide to Training a Machine Learning Model using BigQueryML (BQML) - GetInData) by running the following query: 1_bqml_ga4_dataset.sql

AutoML model training

Now you can create your first AutoML model in BigQuery with the following query:

```
CREATE OR REPLACE MODEL
  bqmlhackathon.ga4.init_model_automl
OPTIONS(MODEL_TYPE = 'AUTOML_CLASSIFIER',
   BUDGET_HOURS = 1.0,
   INPUT_LABEL_COLS = ['addedToCart'],
   OPTIMIZATION_OBJECTIVE = 'MAXIMIZE_AU_ROC'
   )
   AS
SELECT
   *
FROM
 `bqmlhackathon.ga4.ga4_sample_ecommerce`
;
```

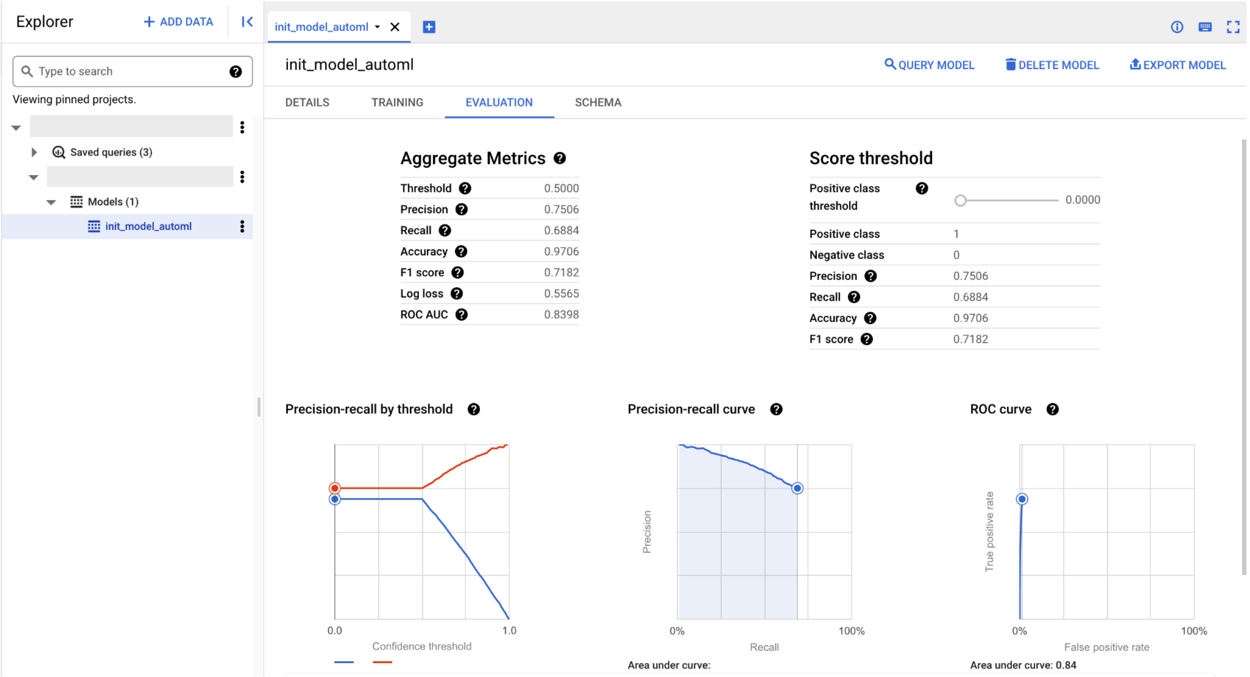
This is a minimal setup, where we only need to specify:

- Model type to *'AUTOML_CLASSIFIER'*
- Budget hours to *1.0* (optional - 1.0 is the default value)
- Input label columns to *['addedToCart']*
- Optimization objective to *'MAXIMIZE_AU_ROC'*

That's all you need for the first model. AutoML will do the rest of the machine learning workflow (but it takes time).

Model evaluation

When the model is ready, you can check its performance on the [Evaluation] tab. Since we're trying to solve a classification problem, our evaluation metric will be ROC AUC (the higher value the better). Note, that it seems like the `budget_hours` limit breaks the training process and thus needs more time to complete.



init_model_automl

Aggregate Metrics

Threshold	0.5000
Precision	0.7506
Recall	0.6884
Accuracy	0.9706
F1 score	0.7182
Log loss	0.5565
ROC AUC	0.8398

Score threshold

Positive class threshold	0.0000
Positive class	1
Negative class	0
Precision	0.7506
Recall	0.6884
Accuracy	0.9706
F1 score	0.7182

AutoML compared with the other model types

In the table below, we compare standard machine learning models and AutoML models performance along with the training time. We measured the elapsed time to create a model on the same data, then measured the model performance using ROC AUC metric.

Model	**Elapsed time**	**ROC AUC**
Standard models		
DNN_CLASSIFIER	25 min 44 sec	0.9893
BOOSTED_TREE_CLASSIFIER	10 min 2 sec	0.9857
LOGISTIC_REG	3 min 11 sec	0.9559
AutoML models		
AUTOML_CLASSIFIER*	7 hr 46 min	0.8398
AUTOML_CLASSIFIER**	1 hr 54 min	0.8387

* BUDGET_HOURS = 4.0
** BUDGET_HOURS = 1.0

As you can see, the model trained by AUTOML_CLASSIFIER gets a high ROC AUC score, but not as high as other models created manually by the data scientists. The results of AutoML may be closer to the expert-created models when we use more training hours (and a higher budget) - the maximum BUDGET_HOURS is 72, but you should keep in mind that the cost of training may not be reasonable to the produced result.

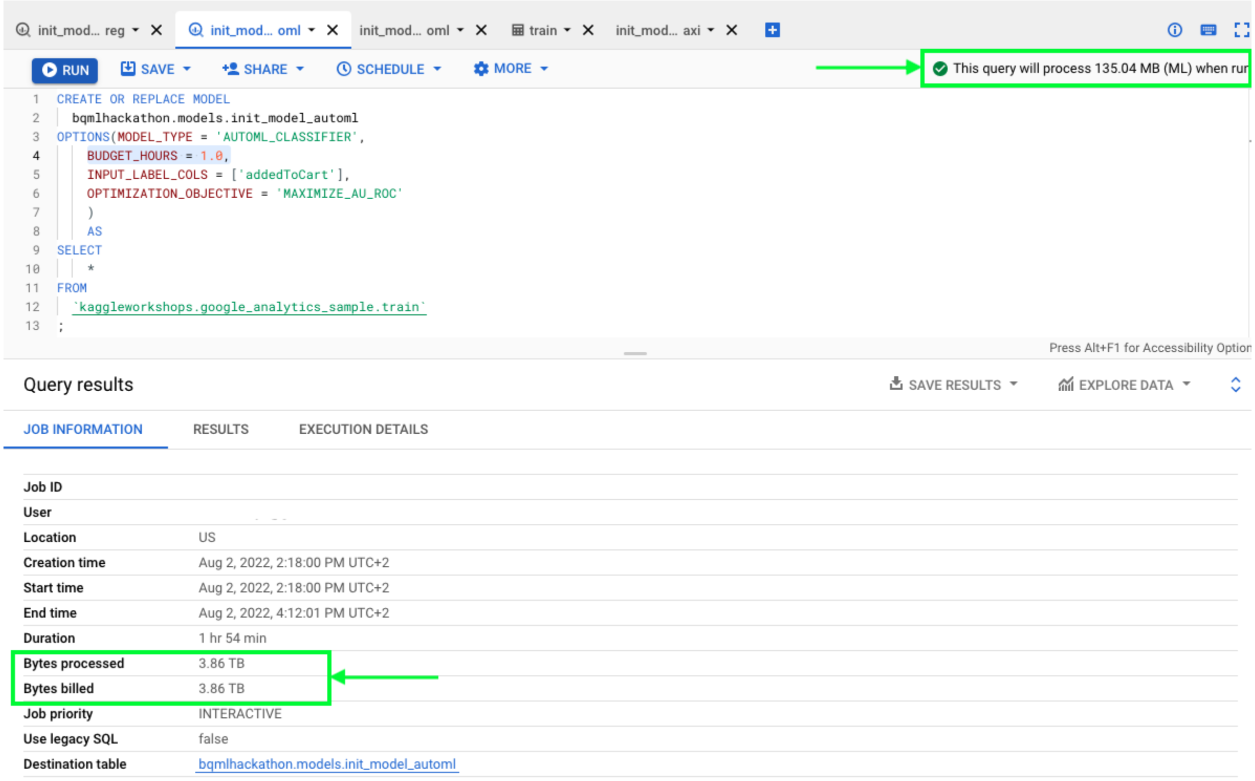
Pricing

1 TB of data processing for AutoML model training in BigQuery costs 5$ plus the Vertex AI training cost [source]. For a more accurate estimate, you should add Vertex AI AutoML pricing - 21$ per 1 hour of training [source].

Be careful with the initial cost estimate visible in the BigQuery console - the preprocessing stage (see: Best practices for creating training data | AutoML Tables | Google Cloud) may explode the dataset size. Also, the preprocessed dataset can be accessed multiple times in parallel to create multiple models during the AutoML training process. This will highly impact the data processed during your BigQueryML training job. Also, the initial cost estimate doesn't seem to include the Vertex AI AutoML pricing.

In our test scenario, the initial estimate for 135 MB turned out to be 3.5 TB, billed for the 1 hour model training. The same training for 4 hours processed 26.69 TB of data.

If you want to hold control over the workflow, but still use AutoML models, you can try Tabular Workflows on Vertex AI, where you can enable/disable auto mode for each workflow step.



This query will process 135.04 MB (ML) when rur

```
CREATE OR REPLACE MODEL
  bqmlhackathon.models.init_model_automl
OPTIONS(MODEL_TYPE = 'AUTOML_CLASSIFIER',
    BUDGET_HOURS = 1.0,
    INPUT_LABEL_COLS = ['addedToCart'],
    OPTIMIZATION_OBJECTIVE = 'MAXIMIZE_AU_ROC'
    )
    AS
SELECT
    *
FROM
  `kaggleworkshops.google_analytics_sample.train`
;
```

Query results

Job ID	
User	
Location	US
Creation time	Aug 2, 2022, 2:18:00 PM UTC+2
Start time	Aug 2, 2022, 2:18:00 PM UTC+2
End time	Aug 2, 2022, 4:12:01 PM UTC+2
Duration	1 hr 54 min
Bytes processed	3.86 TB
Bytes billed	3.86 TB
Job priority	INTERACTIVE
Use legacy SQL	false
Destination table	bqmlhackathon.models.init_model_automl

Next steps

What can you do next with your AutoML model? The two standard ways are to get predictions on the new data within BigQuery, or export the model to use it outside the BigQuery environment (i.e. for online predictions serving in your application).

Predict on the new data

To get the predictions on the new data, BigQuery offers the ML.PREDICT function with the following syntax (See: The ML.PREDICT function | BigQuery ML | Google Cloud)

```
SELECT
  sessionId,
  prob as addedToCart
FROM
  ML.PREDICT(MODEL `bqmlhackathon.ga4.init_model_logistic_reg`,
     (
SELECT *,CONCAT(fullVisitorId, CAST(visitStartTime as string)) as
sessionId
FROM `bqmlhackathon.ga4.ga4_sample_ecommerce_20201231`)
      ),
  UNNEST(predicted_addedToCart_probs)
WHERE label = 1
```

Export the model

You can export your AutoML for use out of BigQuery:
Exporting models | BigQuery ML | Google Cloud
The model export will produce the model artifact in the TensorFlow SavedModel format.

Unfortunately, online serving of AutoML models is not supported in the Vertex AI Endpoints (Exporting models | BigQuery ML | Google Cloud). The reason behind this is that the AutoML models require a custom Docker container for model serving, which is not included in the pre-built containers in VertexAI Endpoints (gcr.io/cloud-automl-tables-public/model_server).
You can still deploy the AutoML model on Vertex AI Endpoints by building the custom Docker container and storing it in the Artifact Registry. However, this scenario introduces an extra complexity which may not be balanced by the benefits of using AutoML.

As mentioned before, AutoML won’t be an ideal approach in every case, but it can be a good point to start the Machine Learning journey. Below you'll find a short list of complementary articles that can help you along the way.

Further reading

- A Step-by-Step Guide to Training a Machine Learning Model using BigQueryML (BQML) - GetInData
- AutoML Tables is now generally available in BigQuery ML | Google Cloud Blog
- Best practices for creating training data | AutoML Tables | Google Cloud