

[544] BigQuery Machine Learning

Tyler Caraza-Harter

Learning Objectives

- use AI.CLASSIFY to integrate LLMs into queries
- write "CREATE MODEL" queries to train models on BigQuery query results
- use a BigQuery TRANSFORM clause to pre-process data prior to training
- use BigQuery's "ML.???" tabular functions to inspect models, make predictions, and evaluate performance

Outline

LLMs

BigQuery ML Basics

Background: Feature Transformation

TopHat+Demos...

Example: AI.CLASSIFY

```
SELECT value, AI.CLASSIFY(value, ["A", "B", "C"]) FROM ...
```

```
SELECT value, AI.CLASSIFY(value, ["A", "B", "C"], endpoint => 'gemini-2.5-flash')...
```

```
SELECT value, AI.CLASSIFY(value, [("A", "description"), ...])...
```

Demo...

Outline

LLMs

BigQuery ML Basics

Background: Feature Transformation

TopHat+Demos...

Train/Test Split

BigQuery provides a `DATA_SPLIT_METHOD` config, but its a bit unusual.

Default behavior depends on dataset

- <500 rows: 100% training data
- <50K rows: 80% training data
- bigger: 10K rows for test, rest for training

Documentation: "When there is a data split, you can find the temporary split results (Training Data, Evaluation Data) on the Model Details page in the BigQuery Console and the model API `data_split_result` field. **These split tables will be saved for 48 hours.** If you will need them for longer than 48 hours, copy them out of the anonymous dataset for longer retention."

Recommendation:

- split manually using `rand()<ratio` in SQL (`rand` gives num between 0 and 1)
- disable BigQuery splitting: **`DATA_SPLIT_METHOD="NO_SPLIT"`**

Training

Step 1: write a query to select both features and label

```
SELECT yesterday_temp, humidity, temp
FROM weather
```

features label
(to predict)

Training

Step 2: choose a model name and create it

```
CREATE OR REPLACE MODEL myproj.mydataset.mymodel
OPTIONS (...)
```

AS

```
SELECT yesterday_temp, humidity, temp
FROM weather
```

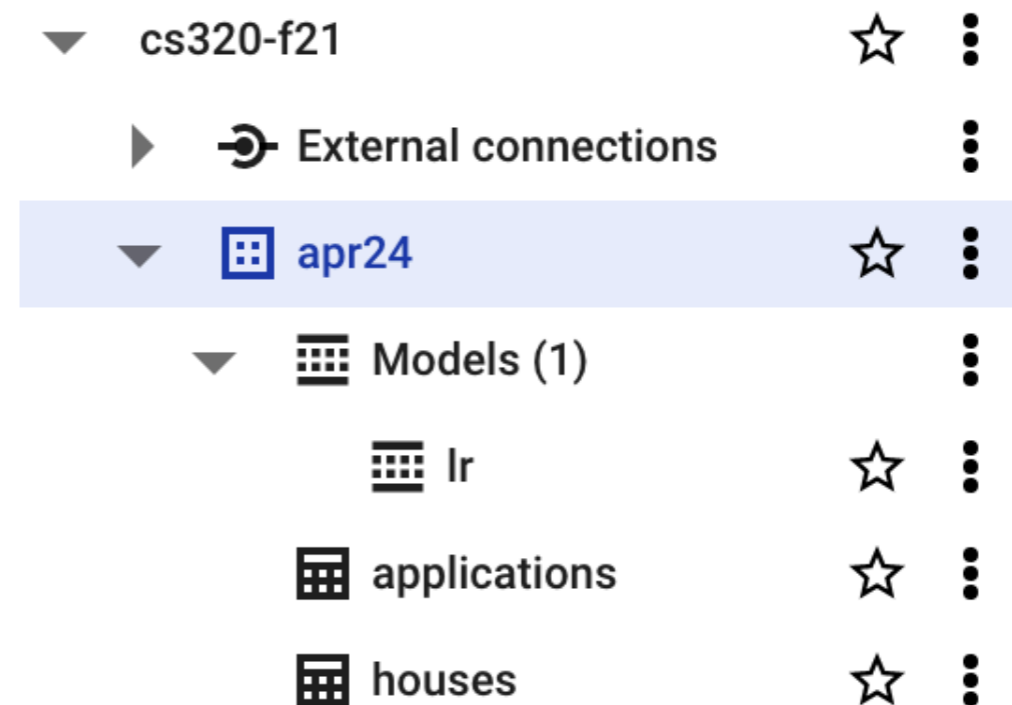
hierarchy:

projects

 datasets

 tables

 models



Training

Step 3: choose type of model

```
CREATE OR REPLACE MODEL myproj.mydataset.mymodel  
OPTIONS (MODEL_TYPE='LINEAR_REG')
```

AS

```
SELECT yesterday_temp, humidity, temp  
FROM weather
```

Options: LINEAR_REG, LOGISTIC_REG, KMEANS, MATRIX_FACTORIZATION,
PCA, AUTOENCODER, AUTOML_CLASSIFIER, AUTOML_REGRESSOR,
BOOSTED_TREE_CLASSIFIER, BOOSTED_TREE_REGRESSOR,
RANDOM_FOREST_CLASSIFIER, RANDOM_FOREST_REGRESSOR,
DNN_CLASSIFIER, DNN_REGRESSOR, DNN_LINEAR_COMBINED_CLASSIFIER,
DNN_LINEAR_COMBINED_REGRESSOR, ARIMA_PLUS, ARIMA_PLUS_XREG,
TENSORFLOW, TENSORFLOW_LITE, ONNX, XGBOOST


Training

Step 4: indicate label column (others are assumed features)

```
CREATE OR REPLACE MODEL myproj.mydataset.mymodel
OPTIONS (MODEL_TYPE='LINEAR_REG',
        INPUT_LABEL_COLS=['temp'])
```

AS

```
SELECT yesterday_temp, humidity, temp
FROM weather
```



Using Trained Models

Each of these functions return a table related to a model.

what are the coefficients used to multiply features?

ML.WEIGHTS(MODEL ????)

what are the predictions given the features?

ML.PREDICT(MODEL ????, (????))

SQL query to get features



how well do we predict (various metrics) given the features+label?

ML.EVALUATE(MODEL ????, (????))

SQL query to get features and label



Using Trained Models

Each of these functions return a table related to a model.

what are the coefficients used to multiply features?

ML.WEIGHTS(MODEL ????)

example:

```
SELECT *  
FROM ML.WEIGHTS (MODEL mymodel)
```

Outline

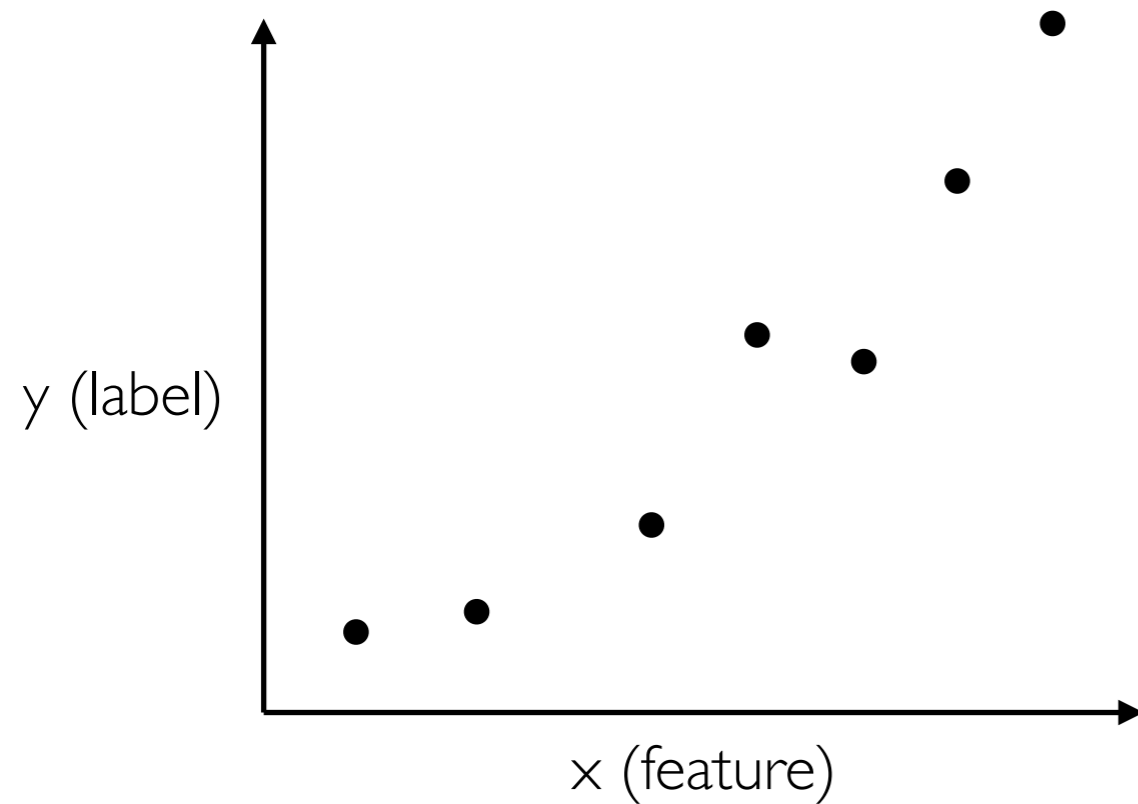
LLMs

BigQuery ML Basics

Background: Feature Transformation

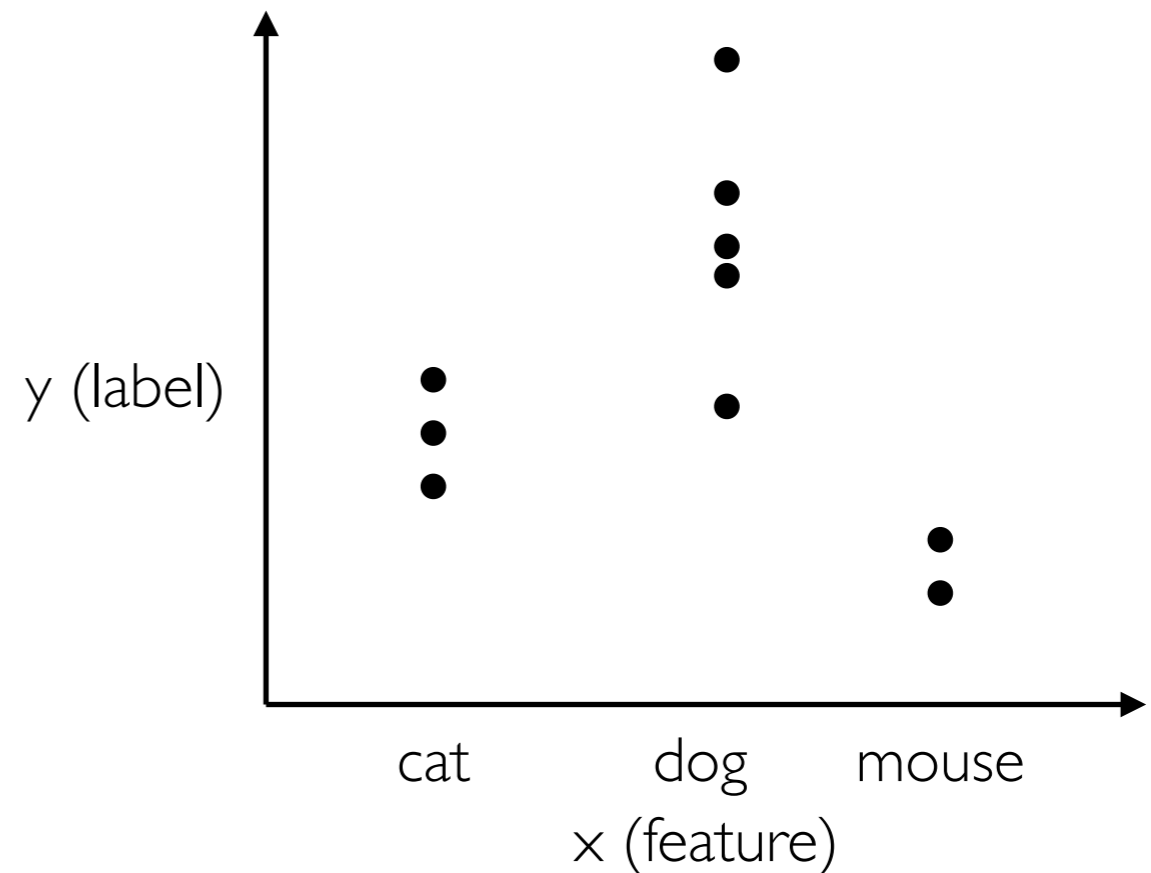
TopHat+Demos...

Patterns and Features



non-linear patterns

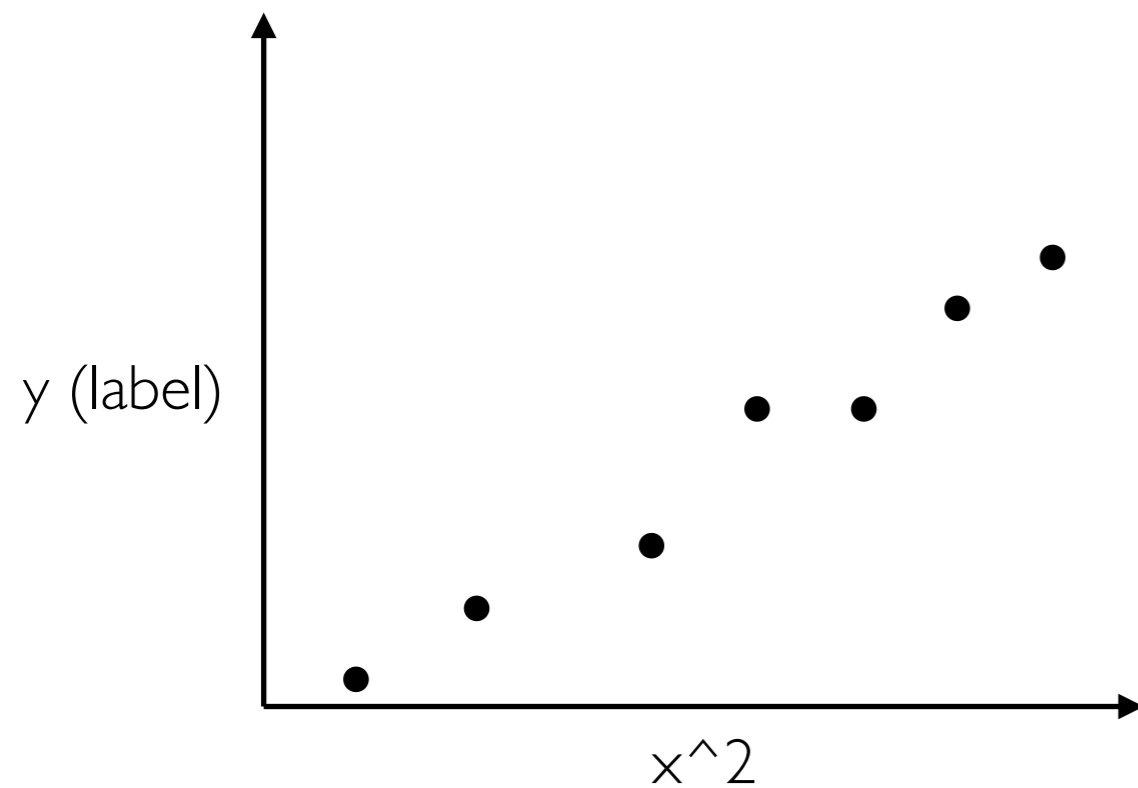
- some models (e.g., DNNs) naturally handle this
- others (e.g., LinearRegression) do not



categorical features

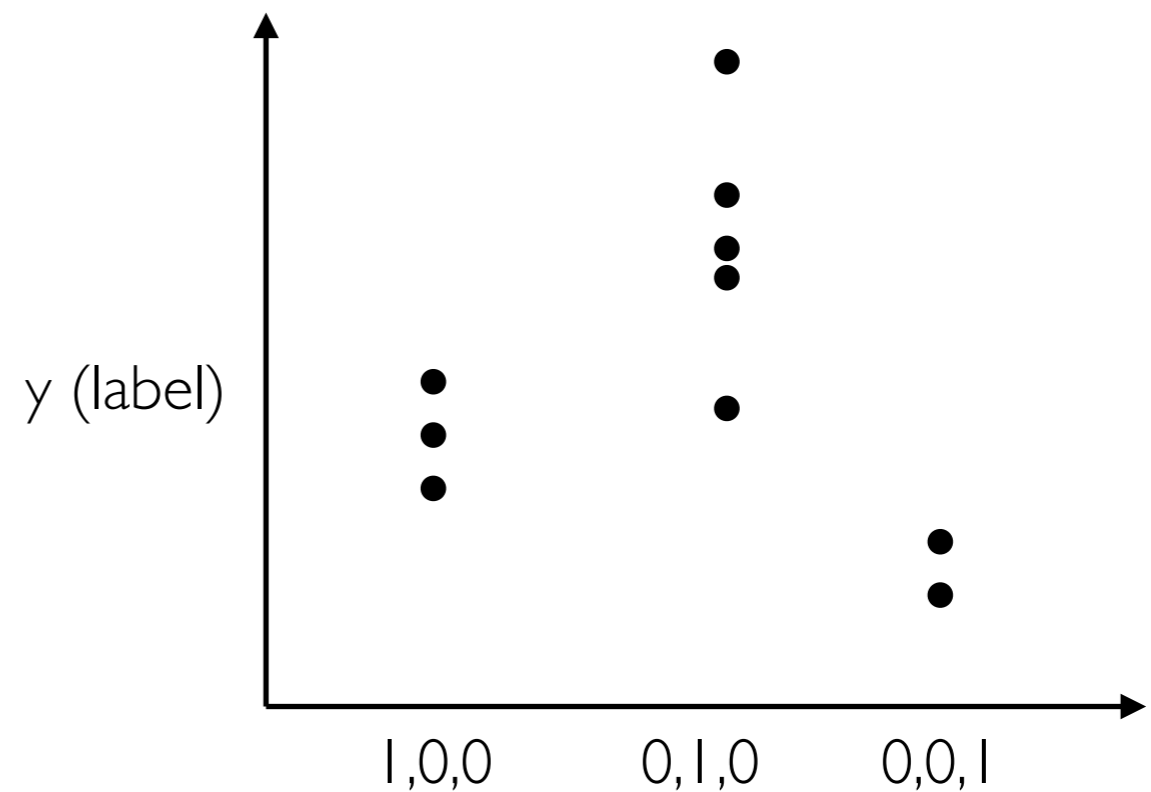
- some models (e.g., DTs) naturally handle this
- others (e.g., LinearRegression) do not

Feature Transformation



non-linear patterns

- can introduce new features than are computed as functions of originals (e.g., $x_2 = x^2$)
- a linear model over the new features corresponds to a non-linear model over the originals



categorical features

- encode categorical features as numbers (e.g., as matrix of zeros and ones for OneHot encoding)

Outline

LLMs

BigQuery ML Basics

Background: Feature Transformation

TopHat+Demos...