  -- Create a cleaned table for modelling

CREATE or replace TABLE

stage1energy.dataset.table\_clean

 as

select ML.LABEL\_ENCODER(BuildingType) OVER () AS BuildingType,

ML.LABEL\_ENCODER(PrimaryPropertyType) OVER () AS PrimaryPropertyType,

ML.LABEL\_ENCODER(Neighborhood) OVER () AS Neighborhood, PropertyGFATotal, PropertyGFABuilding\_s\_,

ML.LABEL\_ENCODER(ListOfAllPropertyUseTypes) OVER () AS ListOfAllPropertyUseTypes,

ML.LABEL\_ENCODER(LargestPropertyUseType) OVER () AS LargestPropertyUseType, LargestPropertyUseTypeGFA,

ML.LABEL\_ENCODER(SecondLargestPropertyUseType) OVER () AS SecondLargestPropertyUseType,

ML.LABEL\_ENCODER(ThirdLargestPropertyUseType) OVER () AS ThirdLargestPropertyUseType, Electricity\_kWh\_, NaturalGas\_therms\_, TotalGHGEmissions

FROM `stage1energy.dataset.table\_all\_cols`

WHERE Electricity\_kWh\_>0

OR TotalGHGEmissions>0;

  -- Checking for anomalies in Y, NANs treatment (Examples of codes, NULLs are treated automatically by BQ ML)

select min(Electricity\_kWh\_), min(TotalGHGEmissions)

from stage1energy.dataset.table\_clean;

select count(\*)

FROM `stage1energy.dataset.table\_clean`

where BuildingType is null;

select distinct(BuildingType), count(BuildingType)

FROM `stage1energy.dataset.table\_clean`

group by 1

order by 2 desc;

UPDATE `stage1energy.dataset.table\_clean\_test`

set SecondLargestPropertyUseType=

(select APPROX\_TOP\_COUNT(SecondLargestPropertyUseType, 1)[OFFSET(0)].value

from `stage1energy.dataset.table\_clean\_test`

where SecondLargestPropertyUseType is not null

limit 1)

where SecondLargestPropertyUseType is null;

UPDATE `stage1energy.dataset.table\_clean\_test`

set LargestPropertyUseTypeGFA=

cast((select PERCENTILE\_CONT(LargestPropertyUseTypeGFA, 0.5) over()

from `stage1energy.dataset.table\_clean\_test`

where LargestPropertyUseTypeGFA is not null limit 1) as int)

where LargestPropertyUseTypeGFA is null;

UPDATE `stage1energy.dataset.table\_clean\_test\_num`

set NaturalGas\_therms\_=

cast((select PERCENTILE\_CONT(NaturalGas\_therms\_, 0.5) over()

from `stage1energy.dataset.table\_clean\_test\_num`

limit 1) as int)

where NaturalGas\_therms\_=0;

-- Y1 ('TotalGHGEmissions')

-- Linear Regression model ('TotalGHGEmissions')

CREATE OR REPLACE MODEL

  stage1energy.dataset.emission\_lr\_model

OPTIONS

  ( model\_type='LINEAR\_REG',

    enable\_global\_explain=TRUE,

    input\_label\_cols=['TotalGHGEmissions'],

    max\_iterations=15,

    DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT')

AS SELECT \* except(Electricity\_kWh\_)

FROM stage1energy.dataset.table\_clean;

-- Random Forest model ('TotalGHGEmissions')

CREATE OR REPLACE MODEL

  stage1energy.dataset.emission\_rf\_model

OPTIONS(MODEL\_TYPE='RANDOM\_FOREST\_REGRESSOR',

        enable\_global\_explain=TRUE,

        NUM\_PARALLEL\_TREE = 50,

        TREE\_METHOD = 'HIST',

        EARLY\_STOP =TRUE,

        INPUT\_LABEL\_COLS = ['TotalGHGEmissions'],

        DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT')

AS SELECT \* except(Electricity\_kWh\_)

FROM stage1energy.dataset.table\_clean;

  -- Deep Neural Network (DNN) model ('TotalGHGEmissions')

CREATE OR REPLACE MODEL stage1energy.dataset.emission\_dnn\_model

OPTIONS(MODEL\_TYPE='DNN\_REGRESSOR',

        enable\_global\_explain=TRUE,

        ACTIVATION\_FN = 'RELU',

        BATCH\_SIZE = 16,

        DROPOUT = 0.1,

        EARLY\_STOP = TRUE,

        HIDDEN\_UNITS = [128, 128, 128],

        INPUT\_LABEL\_COLS = ['TotalGHGEmissions'],

        DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT',

        LEARN\_RATE=0.001,

        MAX\_ITERATIONS = 25,

        OPTIMIZER = 'ADAM')

AS SELECT \* except(Electricity\_kWh\_)

FROM stage1energy.dataset.table\_clean;

  -- Boosted Trees model ('TotalGHGEmissions') / Example of different codes to retrieve the results of the modelling and make predictions

CREATE OR REPLACE MODEL

  stage1energy.dataset.emission\_bt\_model

OPTIONS

      ( MODEL\_TYPE='BOOSTED\_TREE\_REGRESSOR',

        enable\_global\_explain=TRUE,

        BOOSTER\_TYPE = 'GBTREE',

        NUM\_PARALLEL\_TREE = 1,

        MAX\_ITERATIONS = 50,

        TREE\_METHOD = 'HIST',

        EARLY\_STOP = TRUE,

        INPUT\_LABEL\_COLS = ['TotalGHGEmissions'],

        DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT')

AS SELECT \* except(Electricity\_kWh\_)

FROM stage1energy.dataset.table\_clean;

-- evaluate the model on test data ('TotalGHGEmissions')

SELECT

  \*

FROM

  ML.EVALUATE (MODEL `stage1energy.dataset.emission\_bt\_model`);

-- globally explain the model ('TotalGHGEmissions')

SELECT

  \*

FROM

  ML.GLOBAL\_EXPLAIN(MODEL `stage1energy.dataset.emission\_bt\_model`);

  -- the model prediction ('TotalGHGEmissions')

SELECT

  \*

FROM

  ML.PREDICT (MODEL `stage1energy.dataset.emission\_bt\_model`,

    (

      SELECT \* except(Electricity\_kWh\_)

      FROM stage1energy.dataset.table\_clean

     )

  );

  -- explain the model prediction ('TotalGHGEmissions')

SELECT

\*

FROM

ML.EXPLAIN\_PREDICT(MODEL `stage1energy.dataset.emission\_bt\_model`,

  (

  SELECT \* except(Electricity\_kWh\_)

  FROM stage1energy.dataset.table\_clean

  ),

  STRUCT(3 as top\_k\_features));

-- Y2 ('Electricity\_kWh\_')

-- Linear Regression model ('Electricity\_kWh\_')

CREATE OR REPLACE MODEL

  stage1energy.dataset.electricity\_lr\_model

OPTIONS

  ( model\_type='LINEAR\_REG',

    enable\_global\_explain=TRUE,

    input\_label\_cols=['Electricity\_kWh\_'],

    max\_iterations=15,

    DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT')

AS SELECT \* except(TotalGHGEmissions)

FROM stage1energy.dataset.table\_clean;

-- Random Forest model ('Electricity\_kWh\_')

CREATE OR REPLACE MODEL

  stage1energy.dataset.electricity\_rf\_model

OPTIONS(MODEL\_TYPE='RANDOM\_FOREST\_REGRESSOR',

        enable\_global\_explain=TRUE,

        NUM\_PARALLEL\_TREE = 50,

        TREE\_METHOD = 'HIST',

        EARLY\_STOP =TRUE,

        INPUT\_LABEL\_COLS = ['Electricity\_kWh\_'],

        DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT')

AS SELECT \* except(TotalGHGEmissions)

FROM stage1energy.dataset.table\_clean;

  -- Deep Neural Network (DNN) model ('Electricity\_kWh\_')

CREATE OR REPLACE MODEL stage1energy.dataset.electricity\_dnn\_model

OPTIONS(MODEL\_TYPE='DNN\_REGRESSOR',

        enable\_global\_explain=TRUE,

        ACTIVATION\_FN = 'RELU',

        BATCH\_SIZE = 16,

        DROPOUT = 0.1,

        EARLY\_STOP = TRUE,

        HIDDEN\_UNITS = [128, 128, 128],

        INPUT\_LABEL\_COLS = ['Electricity\_kWh\_'],

        DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT',

        LEARN\_RATE=0.001,

        MAX\_ITERATIONS = 25,

        OPTIMIZER = 'ADAM')

AS SELECT \* except(TotalGHGEmissions)

FROM stage1energy.dataset.table\_clean;

  -- Boosted Trees model ('Electricity\_kWh\_') / Example of different codes to retrieve the results of the modelling and make predictions

CREATE OR REPLACE MODEL

  stage1energy.dataset.electricity\_bt\_model

OPTIONS

      ( MODEL\_TYPE='BOOSTED\_TREE\_REGRESSOR',

        enable\_global\_explain=TRUE,

        BOOSTER\_TYPE = 'GBTREE',

        NUM\_PARALLEL\_TREE = 1,

        MAX\_ITERATIONS = 50,

        TREE\_METHOD = 'HIST',

        EARLY\_STOP = TRUE,

        INPUT\_LABEL\_COLS = ['Electricity\_kWh\_'],

        DATA\_SPLIT\_METHOD = 'AUTO\_SPLIT')

AS SELECT \* except(TotalGHGEmissions)

FROM stage1energy.dataset.table\_clean;

-- evaluate the model on test data ('Electricity\_kWh\_')

SELECT

  \*

FROM

  ML.EVALUATE (MODEL `stage1energy.dataset.electricity\_bt\_model`);

-- globally explain the model ('Electricity\_kWh\_')

SELECT

  \*

FROM

  ML.GLOBAL\_EXPLAIN(MODEL `stage1energy.dataset.electricity\_bt\_model`);

  -- the model prediction ('Electricity\_kWh\_')

SELECT

  \*

FROM

  ML.PREDICT (MODEL `stage1energy.dataset.electricity\_bt\_model`,

    (

      SELECT \* except(Electricity\_kWh\_)

      FROM stage1energy.dataset.table\_clean

     )

  );

  -- explain the model prediction ('Electricity\_kWh\_')

SELECT

\*

FROM

ML.EXPLAIN\_PREDICT(MODEL `stage1energy.dataset.electricity\_bt\_model`,

  (

  SELECT \* except(Electricity\_kWh\_)

  FROM stage1energy.dataset.table\_clean

  ),

  STRUCT(3 as top\_k\_features));