

Data Analysis Project in BigQuery

Denis Storozhuk / Data Science project

Stages of the project

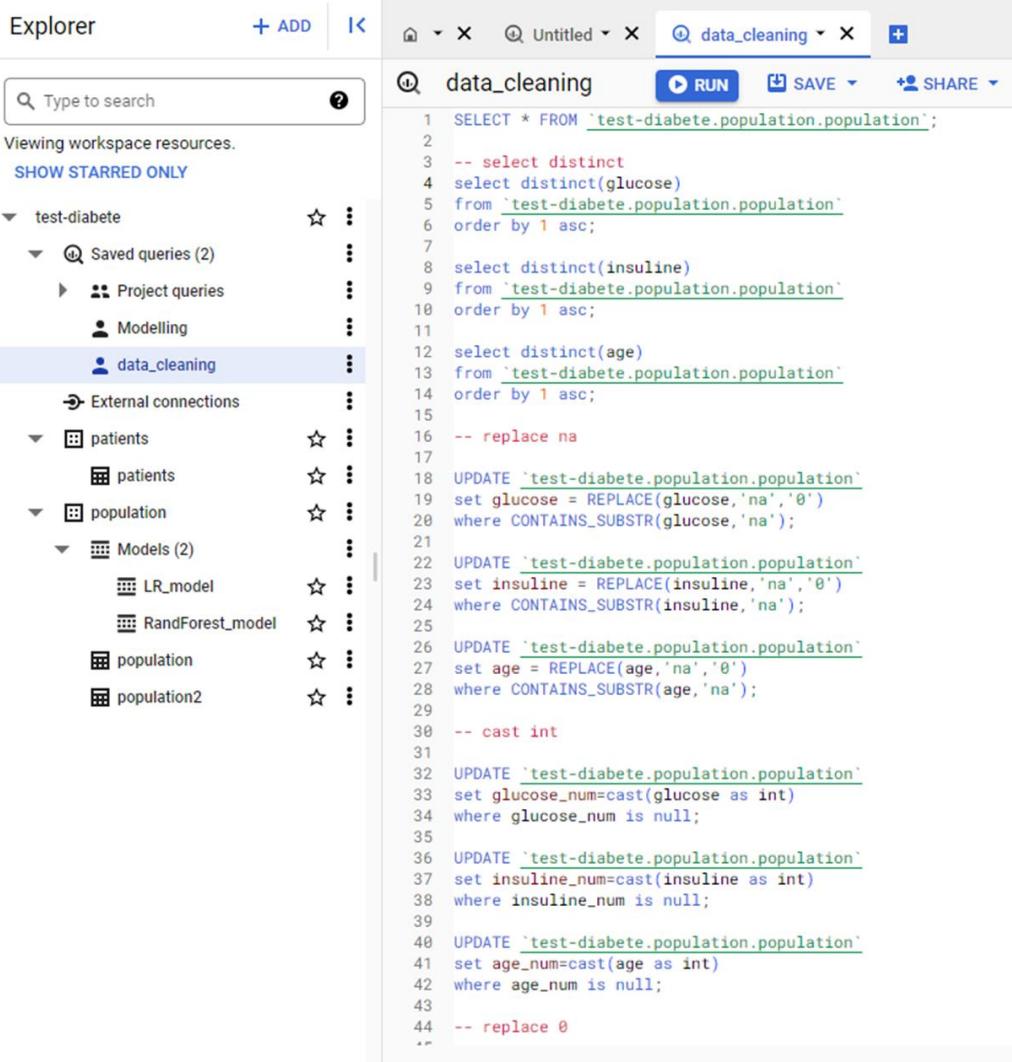
1. Brief / Debriefing
2. Load, Transform, Export data
3. EDA (Exploratory Data Analysis)
4. Modelling
 1. Logistic Regression
 2. Random Forrest Classifier
 3. Deep Neural Networks
 4. Boosted Trees
5. Hyperparameter Tuning
6. Summary / Key Learnings

Data Cleaning

Stages of data cleaning

1. Load data
2. Transform data
 1. Replace NaNs
 2. Check/change datatypes
 3. Replace Zeros with Medians
 4. Drop non-used columns
3. Ready to export data (e.g. for EDA)

The code in SQL



The screenshot shows a SQL editor interface with the following details:

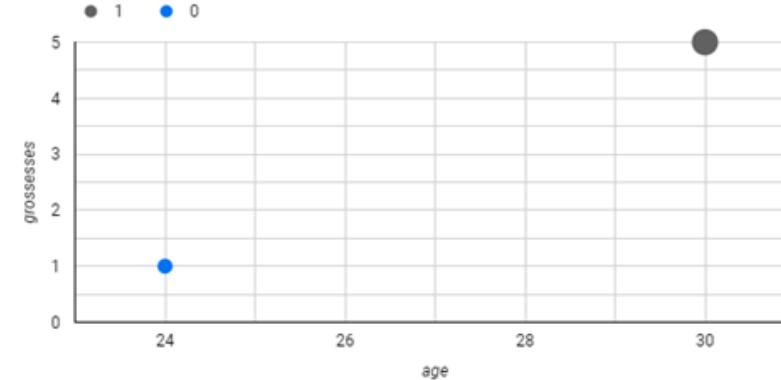
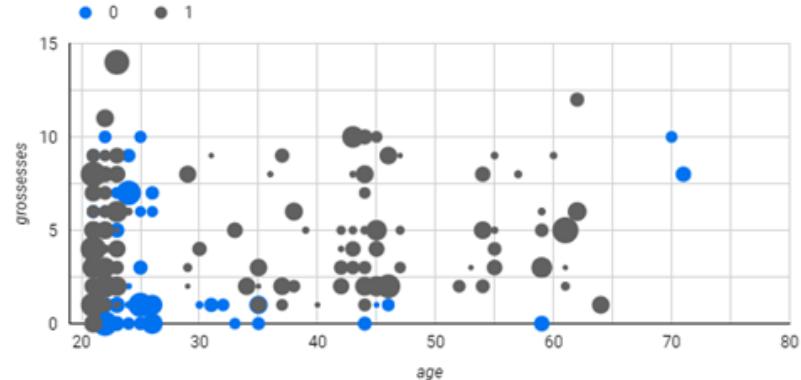
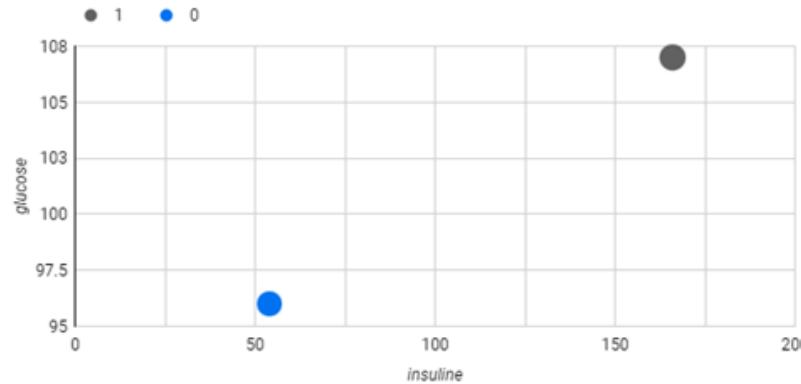
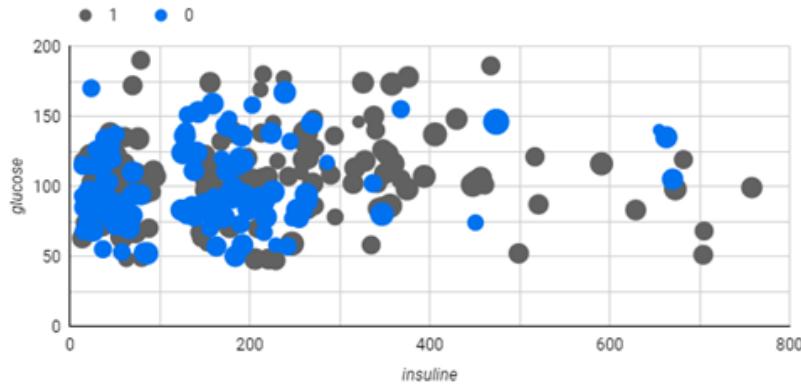
- Explorer:** On the left, it lists workspace resources under "test-diabete". The "data_cleaning" query is selected.
- Query Tab:** The current tab is "data_cleaning".
- Code Area:** The code is as follows:

```
1  SELECT * FROM `test-diabete.population.population`;
2
3  -- select distinct
4  select distinct(glucose)
5  from `test-diabete.population.population`
6  order by 1 asc;
7
8  select distinct(insuline)
9  from `test-diabete.population.population`
10 order by 1 asc;
11
12 select distinct(age)
13 from `test-diabete.population.population`
14 order by 1 asc;
15
16 -- replace na
17
18 UPDATE `test-diabete.population.population`
19 set glucose = REPLACE(glucose,'na','0')
20 where CONTAINS_SUBSTR(glucose,'na');
21
22 UPDATE `test-diabete.population.population`
23 set insuline = REPLACE(insuline,'na','0')
24 where CONTAINS_SUBSTR(insuline,'na');
25
26 UPDATE `test-diabete.population.population`
27 set age = REPLACE(age,'na','0')
28 where CONTAINS_SUBSTR(age,'na');
29
30 -- cast int
31
32 UPDATE `test-diabete.population.population`
33 set glucose_num=cast(glucose as int)
34 where glucose_num is null;
35
36 UPDATE `test-diabete.population.population`
37 set insuline_num=cast(insuline as int)
38 where insuline_num is null;
39
40 UPDATE `test-diabete.population.population`
41 set age_num=cast(age as int)
42 where age_num is null;
43
44 -- replace 0
45
46 UPDATE `test-diabete.population.population`
47 set glucose_num=105
48 where glucose_num=0;
49
50 (select PERCENTILE_CONT(glucose_num, 0.5) over()
51 from `test-diabete.population.population`
52 limit 1)
53
54 UPDATE `test-diabete.population.population`
55 set insuline_num=128
56 where insuline_num=0;
57
58 (select PERCENTILE_CONT(insuline_num, 0.5) over()
59 from `test-diabete.population.population`
60 limit 1)
61
62 UPDATE `test-diabete.population.population`
63 set age_num=24
64 where age_num=0;
65
66 (select PERCENTILE_CONT(age_num, 0.5) over()
67 from `test-diabete.population.population`
68 limit 1)
69
70 -- drop columns
71
72 alter table `test-diabete.population.population`
73 drop column glucose;
74
75 alter table `test-diabete.population.population`
76 drop column insuline;
77
78 alter table `test-diabete.population.population`
79 drop column age;
80
```

EDA / Exploratory Data Analysis

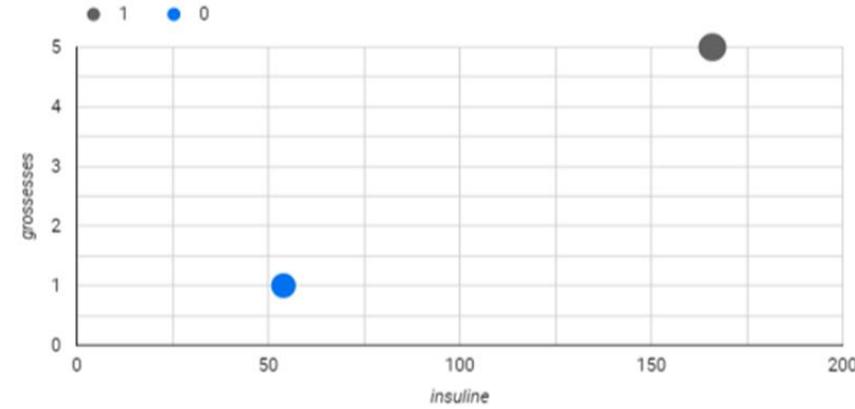
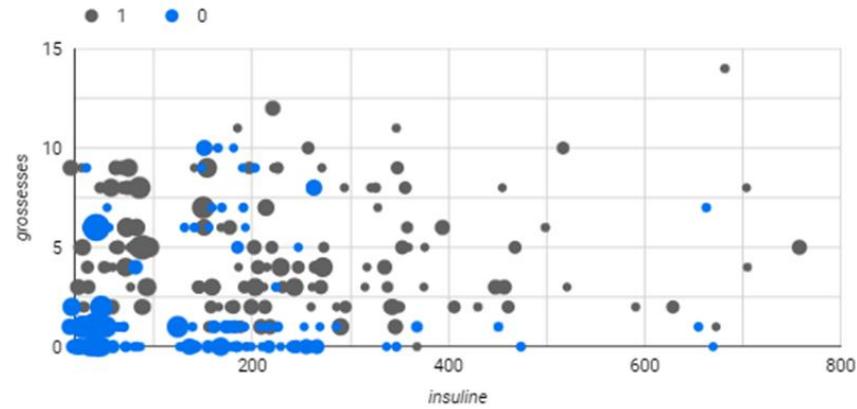
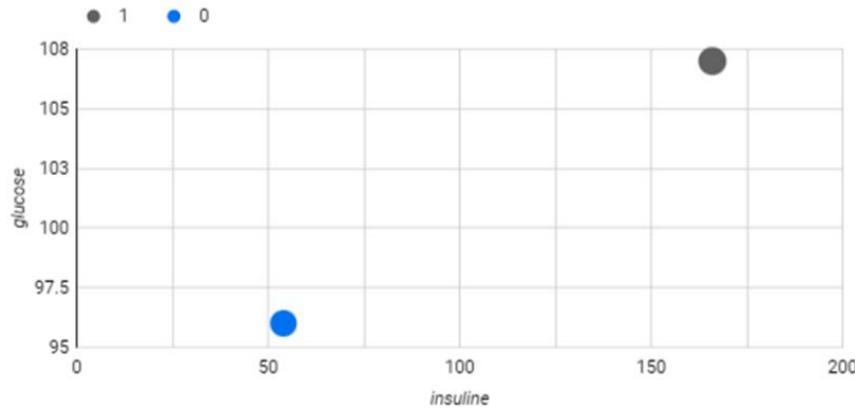
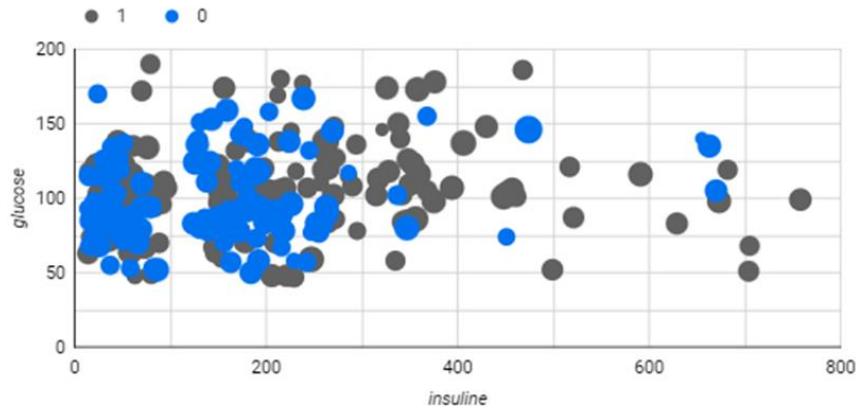
<https://lookerstudio.google.com/s/hutQa9uTOE4>

population



<https://lookerstudio.google.com/s/hutQa9uTOE4>

population



Logistic regression model

Classification Problem

Supervised Learning

Summary metrics (1)

Type to search

Viewing workspace resources.

SHOW STARRED ONLY

- test-diabete
 - Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
 - External connections
 - patients
 - patients
 - population
 - Models (2)
 - LR_model
 - RandForest_model
 - population
 - population2

LR_model

EVALUATION

Aggregate metrics

Threshold	0.5000
Precision	0.7868
Recall	0.7750
Accuracy	0.7825
F1 score	0.7809
Log loss	0.4588
ROC AUC	0.8721

Score threshold

Positive class threshold	0.0187
Positive class	true
Negative class	false
Precision	0.5000
Recall	1.0000
Accuracy	0.5000
F1 score	0.6667

Precision-recall by threshold

Precision-recall curve

ROC curve

Area under curve: 0.843

Area under curve: 0.872

Summary metrics (2)

Explorer + ADD K

Type to search

Viewing workspace resources.
SHOW STARRED ONLY

- test-diabete
 - Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
 - External connections
 - patients
 - patients
 - population
 - Models (5)
 - Boosted_Trees_mo...
 - DNN_model
 - LR_HPT_model
 - LR_model
 - RandForest_model
 - population

Modelling LR_model

LR_model

DETAILS TRAINING EVALUATION INTERPRETABILITY SCHEMA

QUERY MODEL DELETE MODEL EXPORT MODEL

Precision-recall by threshold

Precision-recall curve

ROC curve

Confusion matrix

This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in grey).

True label	Predicted label	
	true	false
true	78%	23%
false	21%	79%

Summary metrics for Test data

Explorer + ADD K

Type to search

Viewing workspace resources.
SHOW STARRED ONLY

- test-diabete
 - Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
 - External connections
 - patients
 - patients
 - population
 - Models (2)
 - LR_model
 - RandForest_model
 - SHOW MORE
 - population
 - population2

Modelling RUN SAVE SHARE SCHEDULE MORE Query completed.

```
31 FROM
32 ML_EVALUATE (MODEL `test-diabete.population.LR_model`);
33
34 -- evaluate the model on test data
35
36 SELECT
37 ...*
38 FROM
39 ML_EVALUATE (MODEL `test-diabete.population.LR_model`,
40 ...(
41 ...  SELECT
42 ...    grossesses,pressure,imc,K,glucose,insuline,age,label
43 ...  FROM
44 ...    `test-diabete.patients.patients`
45 ...
46 ...);
47
48 -- globally explain the model
49
50 SELECT
```

Press Alt+F1 for accessibility options.

Query results SAVE RESULTS EXPLORE DATA

Row	precision	recall	accuracy	f1_score	log_loss	roc_auc
1	0.711538461538...	0.74	0.72	0.725490196078...	0.561732922173...	0.803197802197...

Features Importance

The screenshot shows a data modeling interface with the following details:

- Explorer View:** On the left, under the workspace "test-diabete", there are sections for "Saved queries (2)", "External connections", "patients", "population", and "Models (2)". The "LR_model" is selected, highlighted with a blue border.
- Model Overview:** The main area shows the model "LR_model" with tabs for DETAILS, TRAINING, EVALUATION, INTERPRETABILITY (which is active), and SCHEMA. There are also buttons for QUERY MODEL, DELETE MODEL, EXPORT MODEL, and REFRESH.
- Explainable AI Section:** This section is titled "Explainable AI" and contains the following text: "Explainable AI provides a way of explaining the model and the predictions it produces. The following contains the features with the largest importance scores for your model overall. For explanationable prediction on a per example basis, please run ML.EXPLAIN_PREDICT or ML.EXPLAIN_FORECAST." Below this, a table lists the top features and their attribution scores:

Feature name	Attribution
grossesses	0,986
insuline	0,56
age	0,544
imc	0,475
K	0,44
pression	0,204
glucose	0,091

SQL code

The screenshot shows a data modeling interface with the following components:

- Explorer Sidebar:** On the left, it displays workspace resources under the project "test-diabete". The "Modelling" folder is selected. Other visible items include "Saved queries (2)", "Project queries", "data_cleaning", "External connections", "patients", "population", and "Models (2)" which contain "LR_model" and "RandForest_model".
- Central Query Editor:** The main area shows an SQL script titled "Modelling". The script is a logistic regression model (LR_model) for predicting diabetes based on patient data. It includes steps for creating the model, evaluating it on training data, evaluating it on test data, and globally explaining the model.
- Toolbar:** At the top, there are buttons for RUN, SAVE, SHARE, SCHEDULE, and MORE. A status message indicates the query will process 211,61 KB.
- Bottom Navigation:** Includes links for PERSONAL HISTORY and PROJECT HISTORY, a REFRESH button, and a message to press Alt+F1 for accessibility options.

```
10
11 -- LOGISTIC MODEL
12
13 CREATE OR REPLACE MODEL
14 | `test-diabete.population.LR_model`
15 OPTIONS
16 | ( model_type='LOGISTIC_REG',
17 | auto_class_weights=TRUE,
18 | enable_global_explain=TRUE,
19 | data_split_method='NO_SPLIT',
20 | input_label_cols=['label'],
21 | max_iterations=15) AS
22 SELECT
23 | grossesses,pressure,imc,K,glucose,insuline,age,label
24 FROM
25 | `test-diabete.population.population`;
26
27 -- evaluate the model on train data
28
29 SELECT
30 | *
31 FROM
32 | ML.EVALUATE (MODEL `test-diabete.population.LR_model`);
33
34 -- evaluate the model on test data
35
36 SELECT
37 | *
38 FROM
39 | ML.EVALUATE (MODEL `test-diabete.population.LR_model`,
40 | (
41 | SELECT
42 | grossesses,pressure,imc,K,glucose,insuline,age,label
43 | FROM
44 | `test-diabete.patients.patients`
45 | )
46 | );
47
48 -- globally explain the model
49
50 SELECT
51 | *
52 FROM
53 | ML.GLOBAL_EXPLAIN(MODEL `test-diabete.population.LR_model`);
```

Random Forest Classifier

Classification Problem

Supervised Learning

Summary metrics (1)

Type to search

RandForest_model

Viewing workspace resources. SHOW STARRED ONLY

- test-diabete
 - Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
 - External connections
 - patients
 - patients
 - population
 - Models (2)
 - LR_model
 - RandForest_model
- population
- population2

DETAILS TRAINING EVALUATION INTERPRETABILITY SCHEMA

Aggregate metrics

Threshold	0.5000
Precision	0.9701
Recall	0.9750
Accuracy	0.9725
F1 score	0.9726
Log loss	0.2167
ROC AUC	0.9972

Score threshold

Positive class threshold	0.1326
Positive class	true
Negative class	false
Precision	0.5000
Recall	1.0000
Accuracy	0.5000
F1 score	0.6667

Precision-recall by threshold

A line graph showing precision on the y-axis (0.0 to 1.0) against confidence threshold on the x-axis (0.0 to 1.0). Two curves are shown: a blue curve starting at (0, 1) and a red curve starting at approximately (0.1, 0.1). The blue curve drops sharply to 0 at a threshold of about 0.7.

Area under curve: 0.996

Precision-recall curve

A line graph showing precision on the y-axis (0% to 100%) against recall on the x-axis (0% to 100%). A single blue curve starts at (0%, 100%) and drops to (100%, 0%).

Area under curve: 0.996

ROC curve

A line graph showing true positive rate on the y-axis (0% to 100%) against false positive rate on the x-axis (0% to 100%). A single blue curve starts at (0%, 0%) and goes to (100%, 100%).

Area under curve: 0.996

Summary metrics (2)

Type to search ?

Viewing workspace resources. [SHOW STARRED ONLY](#)

- test-diabete
- Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
- External connections
- patients
- population
- Models (5)
 - Boosted_Trees_mo...
 - DNN_model
 - LR_HPT_model
 - LR_model
 - RandForest_model
- population

RandForest_model

QUERY MODEL DELETE MODEL EXPORT MODEL CREFR

DETAILS TRAINING EVALUATION INTERPRETABILITY SCHEMA

Precision-recall by threshold ?

Precision-recall curve ?

ROC curve ?

Confusion matrix

This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in grey).

True label	Predicted label	
	true	false
true	98%	3%
false	3%	97%

Summary metrics for Test data

The screenshot shows a data modeling workspace interface. The left sidebar displays workspace resources under categories like test-diabete, patients, population, and Models. The central area is a query editor titled "Modelling" with the following SQL code:

```
74 FROM
75 ML.EVALUATE (MODEL `test-diabete.population.RandForest_model`);
76
77 -- evaluate the model on test data
78
79 SELECT
80 ...
81 FROM
82 ML.EVALUATE (MODEL `test-diabete.population.RandForest_model`,
83 ...
84 (
85 ...
86 ...
87 'test-diabete.patients.patients'
88 ...
89 );
90
91 -- globally explain the model
92
93 SELECT
94 *
```

The status bar indicates "Query complete". Below the query editor is a "Query results" section with tabs for JOB INFORMATION, RESULTS, JSON, EXECUTION DETAILS, and EXECUTION GRAPH. The RESULTS tab is selected, showing a table with columns: Row, precision, recall, accuracy, f1_score, log_loss, and roc_auc. A single row of data is present:

Row	precision	recall	accuracy	f1_score	log_loss	roc_auc
1	0.933333333333...	0.84	0.89	0.884210526315	0.322185200369...	0.959001998001...

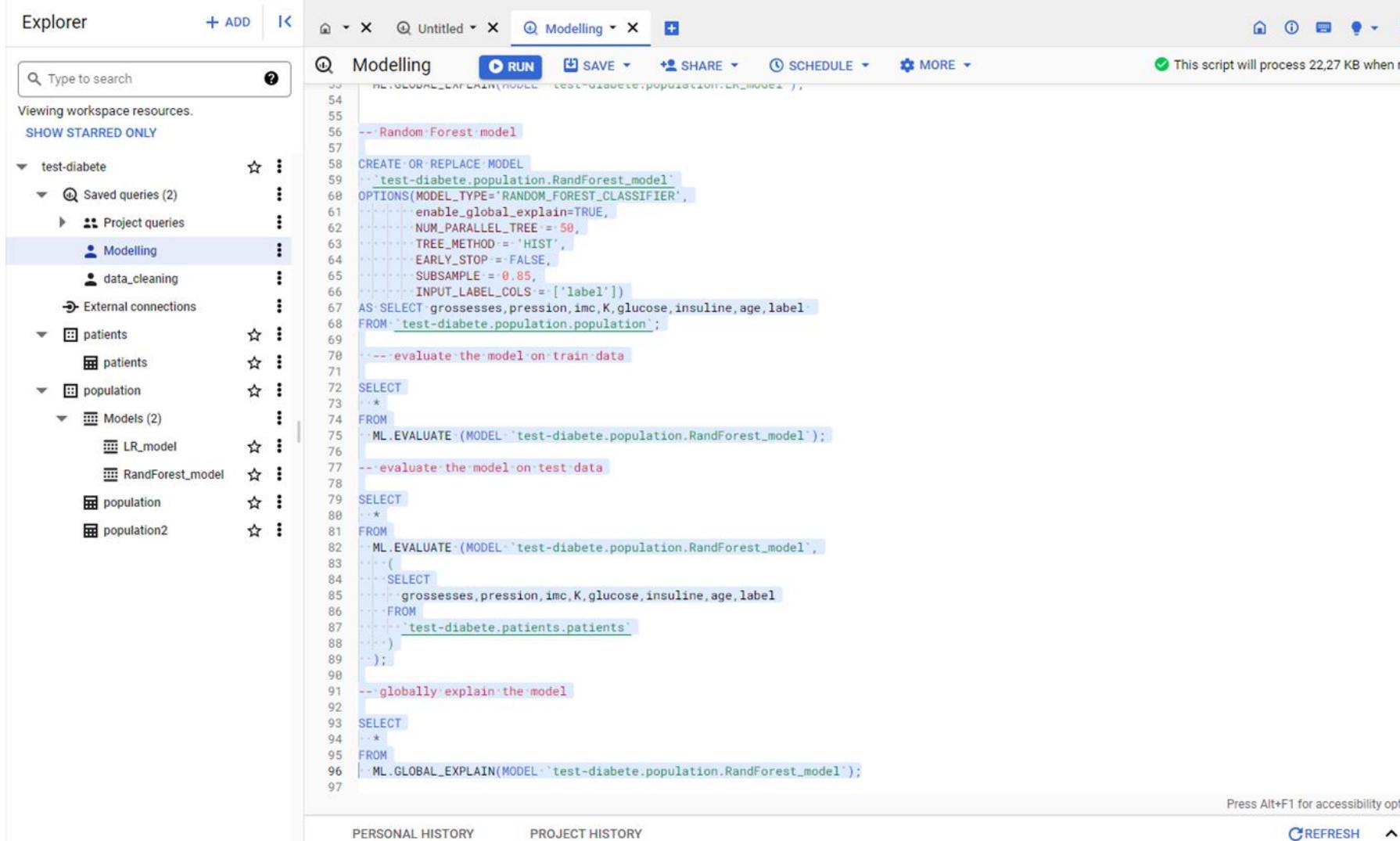
Features Importance

The screenshot shows a data workspace interface with the following details:

- Explorer:** On the left, a sidebar lists workspace resources under "test-diabete".
 - Saved queries (2):** Project queries, Modelling, data_cleaning.
 - External connections:** (empty)
 - patients:** patients (empty)
 - population:** population (empty)
 - Models (2):** LR_model, RandForest_model (selected).
- RandForest_model Details:** The main panel shows the "RandForest_model" tab selected. It includes tabs for DETAILS, TRAINING, EVALUATION, INTERPRETABILITY (which is active), and SCHEMA.
- Interpretability:** The "Explainable AI" section provides a list of features with their attribution scores:

Feature name	Attribution
grossesses	0,533
imc	0,183
insuline	0,139
age	0,13
glucose	0,044
pression	0,021
K	0,019

SQL code



The screenshot shows a SQL development environment with the following interface elements:

- Explorer** pane on the left, displaying workspace resources under "test-diabete".
- Modelling** tab selected in the top bar.
- Code Editor** pane containing the following SQL script:

```
RE.GEVERSE_EARLYSTOP(MODEL `test-diabete.population.RandForest_model`);  
-- Random Forest model  
CREATE OR REPLACE MODEL `test-diabete.population.RandForest_model`  
OPTIONS(MODEL_TYPE='RANDOM_FOREST_CLASSIFIER',  
       enable_global_explain=TRUE,  
       NUM_PARALLEL_TREE = 50,  
       TREE_METHOD = 'HIST',  
       EARLY_STOP = FALSE,  
       SUBSAMPLE = 0.85,  
       INPUT_LABEL_COLS = ['label'])  
AS SELECT grossesses,pressure,imc,K,glucose,insuline,age,label  
FROM `test-diabete.population.population`;  
-- evaluate the model on train data  
SELECT *  
FROM  
ML.EVALUATE(MODEL `test-diabete.population.RandForest_model`);  
-- evaluate the model on test data  
SELECT *  
FROM  
ML.EVALUATE(MODEL `test-diabete.population.RandForest_model`,  
           (  
           SELECT  
           grossesses,pressure,imc,K,glucose,insuline,age,label  
           FROM  
           `test-diabete.patients.patients`  
           ));  
-- globally explain the model  
SELECT *  
FROM  
ML.GLOBAL_EXPLAIN(MODEL `test-diabete.population.RandForest_model`);  
97
```

The code implements a Random Forest classifier on the "population" dataset from the "test-diabete" project. It includes global explanation and evaluation on both training and testing datasets.

Deep Neural Networks

Classification Problem

Supervised Learning

Summary metrics (1)

Type to search

Viewing workspace resources. SHOW STARRED ONLY

- test-diabete
 - Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
 - External connections
- patients
 - patients
- population
 - Models (4)
 - Boosted_Trees_mo...
 - DNN_model
 - LR_model
 - RandForest_model
 - population
 - population2

DNN_model

QUERY MODEL DELETE MODEL EXPORT MODEL REFRESH

EVALUATION

Aggregate metrics

Threshold	0.5000
Precision	0.8916
Recall	0.9050
Accuracy	0.8975
F1 score	0.8983
Log loss	0.2339
ROC AUC	0.9678

Score threshold

Positive class threshold	0.0003
Positive class	true
Negative class	false
Precision	0.5000
Recall	1.0000
Accuracy	0.5000
F1 score	0.6667

Precision-recall by threshold

Precision

Confidence threshold

Area under curve: 0.965

Precision-recall curve

Precision

Recall

ROC curve

True positive rate

False positive rate

Area under curve: 0.968

Summary metrics (2)

Type to search

DNN_model

Viewing workspace resources.
SHOW STARRED ONLY

- test-diabete
- Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
- External connections
- patients
- population
- Models (5)
 - Boosted_Trees_mo...
 - DNN_model
 - LR_HPT_model
 - LR_model
 - RandForest_model
- population

QUERY MODEL DELETE MODEL EXPORT MODEL REFRES

EVALUATION DETAILS TRAINING SCHEMA

Precision-recall by threshold

Precision-recall curve

ROC curve

Confusion matrix

This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in grey).

True label	Predicted label	
	true	false
true	90%	10%
false	11%	90%

Item counts

Summary metrics for Test data

The screenshot shows a data workspace interface with the following details:

- Explorer:** On the left, under "test-diabete" (selected), there are two saved queries: "Modelling" (selected) and "data_cleaning". Other sections include "patients", "population", and "Models (4)" which contains "Boosted_Trees_mo...", "DNN_model" (selected), "LR_model", and "RandForest_model".
- Modelling Tab:** The main tab shows a query for evaluating a DNN model on test data. The code is as follows:

```
117 SELECT
118 | *
119 FROM
120 ML.EVALUATE (MODEL `test-diabete.population.DNN_model`);
121
-- evaluate the model on test data
123
124 SELECT
125 | *
126 FROM
127 | ML.EVALUATE (MODEL `test-diabete.population.DNN_model`,
128 | (
129 | | SELECT
130 | | | grossesses,pressure,imc,K,glucose,insuline,age,label
131 | | | FROM
132 | | | | `test-diabete.patients.patients`
133 | | )
134 | );
135
-- globally explain the model
137
138 SELECT
139 | *
140 FROM
141 | ML.GLOBAL_EXPLAIN(MODEL `test-diabete.population.DNN_model`);
142
-- Boosted Trees model
144
145 CREATE MODEL `test-diabete.population.Boosted_Trees_model`
```

- Query results:** Below the code, it says "Press Alt+F1 for accessibility options". The results table has columns: Row, precision, recall, accuracy, f1_score, log_loss, and roc_auc. A single row is shown with values: 0.775510204081..., 0.76, 0.77, 0.767676767676..., 0.557753980810..., 0.846214785214..., and an empty cell.

SQL code

Type to search ?

Modelling RUN SAVE SHARE SCHEDULE MORE This script will process 54 KB when run

Viewing workspace resources.

SHOW STARRED ONLY

test-diabete

- Saved queries (2)
 - Project queries
 - Modelling
- External connections
- patients
- population
- Models (4)
 - Boosted_Trees_mo...
 - DNN_model
 - LR_model
 - RandForest_model
- population
- population2

```
97 -- Deep Neural Network (DNN) model
98
99
100 CREATE OR REPLACE MODEL `test-diabete.population.DNN_model`
101 OPTIONS(MODEL_TYPE='DNN_CLASSIFIER',
102         enable_global_explain=TRUE,
103         ACTIVATION_FN = 'RELU',
104         BATCH_SIZE = 16,
105         DROPOUT = 0.1,
106         EARLY_STOP = TRUE,
107         HIDDEN_UNITS = [128, 128, 128],
108         INPUT_LABEL_COLS = ['label'],
109         LEARN_RATE=0.001,
110         MAX_ITERATIONS = 25,
111         OPTIMIZER = 'ADAM')
112 AS SELECT grossesses.pression,imc,K,glucose,insuline,age,label
113 FROM `test-diabete.population.population`;
114
115 -- evaluate the model on train data
116 SELECT
117   *
118 FROM
119   ML.EVALUATE (MODEL `test-diabete.population.DNN_model`);
120
121 -- evaluate the model on test data
122 SELECT
123   *
124 FROM
125   ML.EVALUATE (MODEL `test-diabete.population.DNN_model`,
126   (
127     (
128       SELECT
129       | grossesses,pression,imc,K,glucose,insuline,age,label
130       FROM
131       | `test-diabete.patients.patients`
132     )
133   );
134
135 -- globally explain the model
136 SELECT
137   *
138 FROM
139   ML.GLOBAL_EXPLAIN(MODEL `test-diabete.population.DNN_model`);
```

Boosted Trees Classifier

Classification Problem

Supervised Learning

Summary metrics (1)

Type to search

Viewing workspace resources.
SHOW STARRED ONLY

Boosted_Trees_model

DETAILS TRAINING EVALUATION INTERPRETABILITY SCHEMA

Aggregate metrics

Threshold	0.5000
Precision	1.0000
Recall	1.0000
Accuracy	1.0000
F1 score	1.0000
Log loss	0.0117
ROC AUC	1.0000

Score threshold

Positive class threshold	0.0001
Positive class	true
Negative class	false
Precision	0.5000
Recall	1.0000
Accuracy	0.5000
F1 score	0.6667

Precision-recall by threshold

Precision-recall curve

ROC curve

Area under curve: 0.997

Area under curve: 1

Summary metrics (2)

Explorer + ADD 🔍

Type to search

Viewing workspace resources. SHOW STARRED ONLY

- test-diabete
 - Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
 - External connections
 - patients
 - patients
 - population
 - Models (4)
 - Boosted_Trees_mo... (selected)
 - DNN_model
 - LR_model
 - RandForest_model
 - population
 - population2

Boosted_Trees_model

EVALUATION

Precision-recall by threshold

Precision

Confidence threshold

Area under curve: 0.997

Precision-recall curve

ROC curve

True positive rate

False positive rate

Area under curve: 1

Confusion matrix

This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in grey).

True label	Predicted label	
	true	false
true	100%	0%
false	100%	0%

PERSONAL HISTORY PROJECT HISTORY REFRESH

Summary metrics for Test data

The screenshot shows a data modeling interface with the following details:

- Header:** Search bar with placeholder "Search (/) for resources, docs, products and more" and a "Search" button.
- Toolbar:** Includes icons for Home, Help, and More.
- Explorer:** Shows workspace resources under "test-diabete".
 - Saved queries (2):** Project queries and Modelling (selected).
 - patients:** patients and patients.
 - population:** population and Models (4).
 - Models: Boosted_Trees_mo..., DNN_model, LR_model, RandForest_model.
 - population and population2.
- Modelling Tab:** Active tab. It contains the following SQL code:

```
159
160 SELECT
161 *
162 FROM
163 ML.EVALUATE (MODEL `test-diabete.population.Boosted_Trees_model`);
164
165 -- evaluate the model on test data
166
167 SELECT
168 ...
169 FROM
170 ML.EVALUATE (MODEL `test-diabete.population.Boosted_Trees_model`,
171 ...
172 ...
173 ...
174 ...
175 ...
176 ...
177 ...
178 ...
179 -- globally explain the model
```
- Query results:** A table showing summary metrics for the test data.

Row	precision	recall	accuracy	f1_score	log_loss	roc_auc
1	0.96	0.96	0.96	0.96	0.119129015752...	0.992408591408...

Features Importance

Type to search ?

Viewing workspace resources. SHOW STARRED ONLY

- test-diabete
 - Saved queries (2)
 - Project queries
 - Modelling
 - data_cleaning
 - External connections
- patients
 - patients
- population
 - Models (4)
 - Boosted_Trees_mo... ☆ ::
 - DNN_model ☆ ::
 - LR_model ☆ ::
 - RandForest_model ☆ ::
 - population ☆ ::
 - population2 ☆ ::

Boosted_Trees_model

QUERY MODEL DELETE MODEL EXPORT MODEL REFRESH

DETAILS TRAINING EVALUATION **INTERPRETABILITY** SCHEMA

Explainable AI

Explainable AI provides a way of explaining the model and the predictions it produces. The following contains the features with the largest importance scores for your model overall. For explanationable prediction on a per example basis, please run `ML.EXPLAIN_PREDICT` or `ML.EXPLAIN_FORECAST`.

Feature name	Attribution
grossesses	1,231
imc	0,611
age	0,602
insuline	0,493
glucose	0,418
pression	0,296
K	0,233

SQL code

Modelling RUN SAVE SHARE SCHEDULE MORE This script will process 54 KB when run.

```
142 -- Boosted Trees model
143
144 CREATE MODEL `test-diabete.population.Boosted_Trees_model`
145 OPTIONS(MODEL_TYPE='BOOSTED_TREE_CLASSIFIER',
146           enable_global_explain=TRUE,
147           BOOSTER_TYPE = 'GBTREE',
148           NUM_PARALLEL_TREE = 1,
149           MAX_ITERATIONS = 50,
150           TREE_METHOD = 'HIST',
151           EARLY_STOP = FALSE,
152           SUBSAMPLE = 0.85,
153           INPUT_LABEL_COLS = ['label'])
154 AS SELECT grossesses,pressure,imc,K,glucose,insuline,age,label
155 FROM `test-diabete.population.population`;
156
157 -- evaluate the model on train data
158
159 SELECT
160   *
161 FROM
162   ML.EVALUATE (MODEL `test-diabete.population.Boosted_Trees_model`);
163
164 -- evaluate the model on test data
165
166 SELECT
167   *
168 FROM
169   ML.EVALUATE (MODEL `test-diabete.population.Boosted_Trees_model`,
170   (
171     SELECT
172       grossesses,pressure,imc,K,glucose,insuline,age,label
173     FROM
174       `test-diabete.patients.patients`
175   )
176 );
177
178 -- globally explain the model
179 SELECT
180   *
181 FROM
182   ML.GLOBAL_EXPLAIN(MODEL `test-diabete.population.Boosted_Trees_model`);
```

Press Alt+F1 for accessibility options.

Query results SAVE RESULTS EXPLORE DATA

Prediction for test / SQL code

Explorer + ADD [Modelling](#) *Modelling population

Type to search

Viewing workspace resources.

[SHOW STARRED ONLY](#)

- test-diabete
 - Saved queries (2)
 - Project queries
 - [Modelling](#)
 - [data_cleaning](#)
 - External connections
- patients
 - patients
- population
 - Models (4)
 - Boosted_Trees_mo...
 - DNN_model
 - LR_model
 - RandForest_model
 - population

Modelling Query completed

```
181 -- the model prediction for test data
182 SELECT
183 |
184 |
185 FROM
186 ML.PREDICT (MODEL `test-diabete.population.Boosted_Trees_model`,
187 (
188   SELECT
189   | grossesses,pressure,imc,K,glucose,insuline,age,label
190   FROM
191   | `test-diabete.patients.patients`
192   )
193 )
194
195 -- explain the model prediction for test data
196 SELECT
197 *
198 FROM
199 ML.EXPLAIN_PREDICT(MODEL `test-diabete.population.Boosted_Trees_model`,
200 (
201   SELECT
202   | grossesses,pressure,imc,K,glucose,insuline,age,label
203   FROM
204   | `test-diabete.patients.patients`
205   ),
206   STRUCT(3 as top_k_features))
```

Press Alt+F1 for accessibility options

Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		EXECUTION GRAPH			
Row	predicted_	probability	top_feature_attribu...	feature	t... attribution	baseline_prediction	prediction_value	approximation_error	grossesses
1	false	0.989577753469...		grossesses	2.851242065429...	0.545232415199...	3.034668743610...	0.0	0
				imc	-0.21151876449...				
				K	0.211415126919...				
2	false	0.999275551002...		grossesses	2.057404756546...	0.545232415199...	4.263222649693...	0.0	0
				age	1.032763004302...				

Results per page: 50 ▾ 1 – 50 of 100

Hyperparameter Tuning

Example for a Logistic Regression model

Evaluation Summary

Type to search ?

Viewing workspace resources. SHOW STARRED ONLY

- test-diabete ☆ ⋮
 - Saved queries (2) ⋮
 - Project queries ⋮
 - Modelling ⋮
 - data_cleaning ⋮
 - External connections ⋮
- patients ☆ ⋮
 - patients ☆ ⋮
- population ☆ ⋮
 - Models (5) ⋮
 - Boosted_Trees_mo... ☆ ⋮
 - DNN_model ☆ ⋮
 - LR_HPT_model ☆ ⋮
 - LR_model ☆ ⋮
 - RandForest_model ☆ ⋮
 - population ☆ ⋮

LR_HPT_model

🔍 QUERY MODEL trash DELETE MODEL export EXPORT MODEL

DETAILS	TRAINING	EVALUATION	INTERPRETABILITY	SCHEMA			
Trial ID ↑	L1 regularisation	L2 regularisation	Precision	Recall	Accuracy	F1 score	Log loss
1	0.0000	0.0000	0.7868	0.7750	0.7825	0.7809	0.4588
2	0.0000	0.0000	0.7868	0.7750	0.7825	0.7809	0.4588
3	0.0000	0.0000	0.7868	0.7750	0.7825	0.7809	0.4588
4	0.0000	0.0006	0.7868	0.7750	0.7825	0.7809	0.4588
5	0.0000	7.6178	0.7868	0.7750	0.7825	0.7809	0.4625
6	0.0073	0.0002	0.7868	0.7750	0.7825	0.7809	0.4588
7	10.0000	0.0000	0.7755	0.7600	0.7700	0.7677	0.4746
8	10.0000	0.0000	0.7755	0.7600	0.7700	0.7677	0.4746
9 (Optimal)	0.0000	10.0000	0.7979	0.7700	0.7875	0.7837	0.4635
10	0.0002	0.0000	0.7868	0.7750	0.7825	0.7809	0.4588
11 (Optimal)	0.0000	10.0000	0.7979	0.7700	0.7875	0.7837	0.4635
12	0.0000	0.1571	0.7868	0.7750	0.7825	0.7809	0.4588
13 (Optimal)	0.0000	10.0000	0.7979	0.7700	0.7875	0.7837	0.4635
14	0.0000	9.5086	0.7979	0.7700	0.7875	0.7837	0.4632
15	10.0000	0.0209	0.7755	0.7600	0.7700	0.7677	0.4746
16	0.0000	0.0000	0.7868	0.7750	0.7825	0.7809	0.4588
17	0.0000	0.0130	0.7868	0.7750	0.7825	0.7809	0.4588
18	10.0000	0.0000	0.7755	0.7600	0.7700	0.7677	0.4746
19	0.0000	0.0000	0.7868	0.7750	0.7825	0.7809	0.4588
20	10.0000	0.0000	0.7755	0.7600	0.7700	0.7677	0.4746

SQL code

Explorer + ADD 🔍 *Modelling X LR_HPT_model X +

Type to search ?

Viewing workspace resources.

SHOW STARRED ONLY

test-diabete ★ :

- Saved queries (2) ::
- Project queries ::
- Modelling ::
- data_cleaning ::
- External connections ::

patients ★ :

- patients ★ ::

population ★ :

- Models (5) ::
- Boosted_Trees_mo... ★ ::
- DNN_model ★ ::
- LR_HPT_model ★ ::
- LR_model ★ ::
- RandForest_model ★ ::
- population ★ ::

Modelling RUN SAVE SHARE SCHEDULE MORE ✓ Query completed

```
43   FROM
44   | `test-diabete.patients.patients`
45   )
46 );
47
48 -- globally explain the model
49
50 SELECT
51   *
52 FROM
53   ML.GLOBAL_EXPLAIN(MODEL `test-diabete.population.LR_model`);
54
55 -- Hyper Parameter Tuning
56
57 CREATE OR REPLACE MODEL
58   | `test-diabete.population.LR_HPT_model`
59   OPTIONS
60   | ('model_type='LOGISTIC_REG',
61   | 'enable_global_explain=TRUE,
62   | 'input_label_cols=['label'],
63   | 'num_trials=20,
64   | 'max_parallel_trials=2') AS
65   SELECT
66     | 'grossesses,pressure,imc,K,glucose,insuline,age,label'
67   FROM
68   | `test-diabete.population.population`;
```

Press Alt+F1 for accessibility options

Query results SAVE RESULTS EXPLORE DATA

JOB INFORMATION	RESULTS	EXECUTION DETAILS	EXECUTION GRAPH										
Elapsed time 10 min 6 sec	Slot time consumed ? 55 min 29 sec	Stages ? <table border="1"><tr><td>Preprocess</td><td>0</td></tr><tr><td>trial_1:Train</td><td>0</td></tr><tr><td>trial_2:Train</td><td>0</td></tr><tr><td>trial_1:Evaluate</td><td>0</td></tr><tr><td>trial_2:Evaluate</td><td>0</td></tr></table>	Preprocess	0	trial_1:Train	0	trial_2:Train	0	trial_1:Evaluate	0	trial_2:Evaluate	0	Training iterations Completed: 0 Planned: 0
Preprocess	0												
trial_1:Train	0												
trial_2:Train	0												
trial_1:Evaluate	0												
trial_2:Evaluate	0												

Model deployment

Example for the Boosted Trees model

Export the model to the Bucket

Cloud Storage [Bucket details](#) [REFRESH](#) [HELP ASSISTANT](#) [LEARN](#)

Buckets **230712-test-diabete**

Location	Storage class	Public access	Protection
us (multiple regions in United States)	Standard	Not public	None

[OBJECTS](#) [CONFIGURATION](#) [PERMISSIONS](#) [PROTECTION](#) [LIFECYCLE](#) [OBSERVABILITY](#) [INVENTORY REPORTS](#)

Buckets > 230712-test-diabete > BT_model

[UPLOAD FILES](#) [UPLOAD FOLDER](#) [CREATE FOLDER](#) [TRANSFER DATA](#) ▾ [MANAGE HOLDS](#) [DOWNLOAD](#) [DELETE](#)

Filter by name prefix only ▾ [Filter](#) Filter objects and folders Show deleted data [☰](#)

Name	Size	Type	Created	Storage class	Last modified	Public	⋮
assets/	–	Folder	–	–	–	–	⋮
explanation_metadata.json	216 B	application/octet-stream	Jul 18, 2023, 10:24:03 AM	Standard	Jul 18, 2023, 10:24:03 AM	Not p	⬇️ ⋮
main.py	883 B	application/octet-stream	Jul 18, 2023, 10:24:03 AM	Standard	Jul 18, 2023, 10:24:03 AM	Not p	⬇️ ⋮
model.bst	94 KB	application/octet-stream	Jul 18, 2023, 10:24:02 AM	Standard	Jul 18, 2023, 10:24:02 AM	Not p	⬇️ ⋮
xgboost_predictor-0.1.tar.gz	4,2 KB	application/octet-stream	Jul 18, 2023, 10:24:03 AM	Standard	Jul 18, 2023, 10:24:03 AM	Not p	⬇️ ⋮

Register and Deploy the model

Vertex AI

BT_model EDIT DETAILS

Model description: diabete

Region: us-west2 (Los Angeles)

Model labels: -

Versions

Filter: Enter a property name

Version ID	Alias	Status	Description	Endpoints	Created	Labels
1	default	Deployed on Vertex AI	-	BT_ep1	Jul 18, 2023, 12:32:21	

⋮

Edit alias
Edit description
Edit labels
Copy to another region
Delete model version
Deploy to endpoint
Resume training

Vertex AI

BT_ep1 EDIT SETTINGS SAMPLE REQUEST

Region: us-west2

Logs: View Logs

Model Monitoring: Disabled

Model	Status	Deployment resource pool	Most recent alerts	Monitoring	Traffic split	Compute nodes	Type	Created
BT_model (Version 1)	Ready	-	-	Disabled	100 %	Auto (1 minimum, 1 maximum)	Custom trained	Jul 18, 2023, 2:21:53 PM

Workbench request

The screenshot shows the Google Cloud Workbench interface with a Jupyter notebook titled "diabete_model.ipynb". The notebook contains Python code for interacting with Google's AI Platform API to make predictions on diabetes dataset instances.

```
[16]: from typing import Dict, List, Union
       from google.cloud import aiplatform
       from google.protobuf import json_format
       from google.protobuf.struct_pb2 import Value

[18]: project="689275957566"
       endpoint_id="4892536472533467136"
       location="us-west2"

[19]: endpoint_name=f"projects/{project}/locations/{location}/endpoints/{endpoint_id}"

[29]: # projects/689275957566/locations/us-west2/endpoints/4892536472533467136

[20]: endpoint=aiplatform.Endpoint(endpoint_name=endpoint_name)

[25]: instances0=[[0,80,37.22895209,0.28373613,138,71,22]]
       instances1=[[1,63,33.15925172,0.608796752,98,62,57]]

[30]: endpoint.predict(instances=instances0).predictions, endpoint.predict(instances=instances1).predictions

[30]: ([[0.01042224653065205, 0.9895777702331543]],
      [[0.7208105325698853, 0.2791894674301147]])
```

Summary / Key Learnings

Summary of models performance

Model	Accuracy / Train	Accuracy / Test
Logistic Regression / HPT	0,78 / 0,80	0,72 / 0,71
Random Forrest Trees	0,97	0,89
Deep Neural Network	0,90	0,77
Boosted Trees	1,00	0,96

Major points / technical

1. Data cleaning:
 1. Manual cleaning for modelling is not necessary
 2. BigQuery (BQ) ML automatically preprocesses data cleaning (imputation of missing values & data transformation (standardisation, encoding))
 3. But the type of the missing value is to be NULL
2. BQ Looker gives a lot of possibilities to visualize data & make data stories
3. ML models make data transformation automatically, though it may be done manually
4. There is a possibility for Hyper Parameters Tuning to improve the models' performance