Connexió d'R a BigQuery

Examen 10 de gener de 2022

2022-06-30

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat de Matemàtiques i Estadística

UNIVERSITAT DE BARCELONA

Contents

Connectem R a BigQuery

Per a realitzar les nostres consultes a la base de dades sense haver d'importar les dades a R es pot fer servir la llibreria `bigrquery`.

```
#install.packages("bigrquery")
library(bigrquery)
library(kableExtra)
projecte <- "level-oxygen-353005"
library(DBI)
library(dplyr)
library(tidyverse)

dades <- dbConnect(
  bigrquery::bigquery(),
  project = projecte,
  dataset = "examen_final",
  billing = projecte
)
```

En primer lloc, obtenim les llibreries corresponents i ens autentifiquem amb la nostra clau API creada mitjançant el correu electrònic vinculat al nostre compre de *Google Cloud*.

Un cop fet això, ja podrem començar a consultar les nostres dades de BigQuery, així com els conjunts de dades públics.

Sobre el paquet `bigrquery`

El paquet `bigrquery` facilita el treball amb les dades emmagatzemades en Google BigQuery, ja que permet consultar les taules de BigQuery i recuperar metadades sobre els projectes, conjunts de dades, taules i treballs. Aquest paquet proporciona els següents 3 nivells d'abstracció:

- La API[1] de baix nivell proporciona fins sobre la API REST[2] subjacent. Totes les funcions de baix nivell comencen amb **bq_**, i en la seva majoria tenen la forma **bq_nom_verb()**. Aquest nivell d'abstracció és el més apropiat si estàs familiaritzat amb la API REST i vols fer alguna cosa que no està suportat en les APIs d'alt nivell.
- La interfície **DBI** embolica la API de baix nivell i fa que treballar amb BigQuery sigui com treballar amb qualsevol altre sistema de base de dades. Aquesta és la capa més convenient si vols executar consultes SQL en BigQuery o carregar quantitats més petites (és a dir, inferiors a 100 MB) de dades.
- La interfície **dplyr** li permet tractar les taules de BigQuery com si fossin marcs de dades en memòria. Aquesta és la capa més convenient si no vols escriure SQL, sinó que vols que `dplyr` ho escrigui per tu.

[1] El terme API és una abreviatura de *Application Programming Interfaces*, que en català significa interfície de programació d'aplicacions. Es tracta d'un conjunt de definicions i protocols que s'utilitzen per a desenvolupar i integrar el programari de les aplicacions, permetent la comunicació entre dues aplicacions a través d'un conjunt de regles.

[2] Una API de REST és una interfície de programació d'aplicacions (API o API web) que s'ajusta als límits de l'arquitectura REST i permet la interacció amb els serveis web de RESTful.

Descripció de la base de dades

La base de dades està formada per les següents tres taules:

Accident és un llistat d'accidents de trànsit ocorreguts al desembre de 2015 als Estats Units:

- ST_CASE: codi de l'accident (PK)
- DAY: dia de l'accident (de l'1 al 31)
- HOUR: hora de l'accident (99 = desconeguda)
- MINUTE: minut de l'accident (99 = desconegut)
- RUR_URB: informació sobre la localització (1 = Rural, 2 = Urbà, 6 = Via no classificada, 8 = No registrat, 9 = Desconegut)
- DAY_WEEK: dia de la setmana (1 = Diumenge, 2 = Dilluns, ..., 7 = Dissabte)
- FATALS: nombre de ferits a l'accident
- DRUNK_DR: nombre de conductors beguts involucrats a l'accident

Person és un llistat de totes les persones (conductors, passatgers o vianants) involucrades als accidents

- ST_CASE: codi de l'accident al qual està involucrada la persona (PK)
- PER_NO: nombre de persona dins de cada accident (PK)
- AGE: edat de la persona (998 = No registrada, 999 = Desconeguda)
- SEX: sexe de la persona (1 = home, 2 = dona, 8 = No registrat, 9 = Desconegut)
- PER_TYP: tipus de persona (1 = conductor, 2 = ocupant, resta de codis = altres)
- DOA: tipus de víctima (0 = sobreviu, 7 = mort a l'accident, 8 = mort al trasllat, 9 = Desconegut)

Vehicle és un llistat de tots els vehicles involucrats als accidents

- ST_CASE: codi de l'accident al qual està involucrat el vehicle (PK)
- VEH_NO: nombre de vehicle dins de cada accident (PK)
- HIT_RUN: identificador de vehicle fugit (0 = No, 1 = Sí, 9 = Desconegut)
- TRAV_SP: velocitat estimada (mph)[3] del vehicle quan va tenir l'accident (997,998 i 999 = Desconegut)
- PREV_SP: indicador d'existència de límit de velocitat permesa just abans de l'accident (997,998 i 999 = Desconegut)

[3]mph vol dir milles per hora. 100 mph=161 Km/h

Consultes amb `bigrquery`

Interfície DBI

1. Trobeu un llistat (només ST_CASE) de tots els accidents ocorreguts en cap de setmana a les 21:00.

```
query1 <- "SELECT ST_CASE
           FROM `level-oxygen-353005.examen_final.accident`
           WHERE (DAY = 1 or DAY = 7) and HOUR = 21 and MINUTE = 0"

kable(dbGetQuery(dades, query1, n = 10), format = "simple")
```

ST_CASE
62353

2. Elaboreu un llistat d'accidents (ST_CASE) posant al costat el nombre de persones involucrades a l'accident (que anomenareu NUMPERS).

```
query2 <- "SELECT ST_CASE, COUNT(*) AS NUMPERS
           FROM `level-oxygen-353005.examen_final.person`
           GROUP BY ST_CASE"

kable(dbGetQuery(dades, query2, n = 10), format = "simple")
```

ST_CASE	NUMPERS
40761	14
40813	4
50479	4
62283	3
62549	12
62693	4
62883	4
122484	4
180773	4
240404	5

3. Dieu quants accidents tenen més de tres vehicles involucrats.

```
query3 <- "SELECT COUNT(*) AS N_ACCIDENTS FROM (SELECT COUNT(*)
                                FROM `level-oxygen-353005.examen_final.vehicle`
                                GROUP BY ST_CASE
                                HAVING COUNT(*) >3)"

kable(dbGetQuery(dades, query3, n = 10), format = "simple")
```

N_ACCIDENTS
44

4. Llisteu els identificadors (només ST_CASE) dels accidents que tenen lloc a una via no classificada i tenen involucrat un vehicle fugit.

```
query4 <- "SELECT A1.ST_CASE
           FROM `level-oxygen-353005.examen_final.accident` AS A1 LEFT JOIN
                `level-oxygen-353005.examen_final.vehicle` AS A2
           ON A1.ST_CASE = A2.ST_CASE
           WHERE RUR_URB = 6 and HIT_RUN = 1
           GROUP BY A1.ST_CASE"

kable(dbGetQuery(dades, query4, n = 10), format = "simple")
```

ST_CASE
240433

5. Calculeu la mitjana de velocitat dels vehicles involucrats als accidents segons el dia de la setmana (DAY_WEEK) d'ocurrència de l'accident. Abans de res elimineu els casos en què la velocitat és desconeguda.

```
query5 <- "SELECT DAY_WEEK, AVG(TRAV_SP) AS MITJANA
           FROM `level-oxygen-353005.examen_final.vehicle` AS A1 LEFT JOIN
                `level-oxygen-353005.examen_final.accident` AS A2
           ON A1.ST_CASE = A2.ST_CASE
           WHERE TRAV_SP<900
           GROUP BY DAY_WEEK"

kable(dbGetQuery(dades, query5, n = 10), format = "simple")
```

DAY_WEEK	MITJANA
1	50.63590
3	41.01230
6	45.98113
5	44.16461
4	44.10714
7	46.33597
2	41.47126

6. Feu un llistat dels accidents que tenen més de 15 persones involucrades, posant dia, hora i minut d'ocurrència.

```
query6 <- "SELECT DAY, HOUR, MINUTE, ST_CASE
           FROM `level-oxygen-353005.examen_final.accident`
           WHERE ST_CASE in (SELECT ST_CASE
                             FROM `level-oxygen-353005.examen_final.person`
                             GROUP BY ST_CASE
```

```
                            HAVING count(*) > 15)"

kable(dbGetQuery(dades, query6, n = 10), format = "simple")
```

DAY	HOUR	MINUTE	ST_CASE
20	5	27	483017
23	9	27	420707
25	14	35	230148

7. Calculeu el total d'homes conductors (SEX=1, PER_TYP=1) i dones conductores (SEX=2, PER_TYP=1) que han estat involucrats en accidents que tenien al menys 1 conductor begut (DRUNK_DR>0). Podeu fer dues consultes.

```
query7 <- "SELECT SEX, COUNT(*) AS N_ACCIDENTS
           FROM `level-oxygen-353005.examen_final.accident` AS a left join
                `level-oxygen-353005.examen_final.person` AS p
           ON a.ST_CASE = p.ST_CASE
           WHERE DRUNK_DR > 0 AND PER_TYP = 1 AND SEX < 8
           GROUP BY SEX"

kable(dbGetQuery(dades, query7, n = 10), format = "simple")
```

SEX	N_ACCIDENTS
1	765
2	179

8. Calculeu un llistat per edats (en ordre ascendent i només dels que tenen entre 60 i 65 anys ambdós incloses) amb el nombre total de persones de cada edat que han estat involucrades en accidents i quants d'aquestes han estat involucrades en accidents en dilluns (DAY_WEEK=2).

```
query8 <- "SELECT T1.AGE, NUMACC, NUMDILLUNS
           FROM (SELECT AGE, COUNT(*) AS NUMACC
                 FROM `level-oxygen-353005.examen_final.person`
                 WHERE AGE BETWEEN 60 and 65
                 GROUP BY AGE
                 ORDER BY AGE) AS T1 LEFT JOIN
                (SELECT AGE, COUNT(*) AS NUMDILLUNS
                 FROM `level-oxygen-353005.examen_final.person` AS p LEFT JOIN
                      `level-oxygen-353005.examen_final.accident` AS a
                 ON p.ST_CASE = a.ST_CASE
                 WHERE AGE BETWEEN 60 and 65 AND DAY_WEEK = 2
                 GROUP BY AGE
                 ORDER BY AGE) as T2
           ON T1.AGE=T2.AGE "

kable(dbGetQuery(dades, query8, n = 10), format = "simple")
```

AGE	NUMACC	NUMDILLUNS
60	78	15
61	66	5
62	71	8
63	73	9
64	59	5
65	72	10

9. Feu un llistat dels accidents de zona urbana (RUR_URB=2), durant un diumenge (DAY_WEEK=1) de 0:00 a 2:59 (ambdós inclosos) amb al menys dos ferits (FATALS>1), posant només ST_CASE .

```
query9 <- "SELECT ST_CASE
           FROM `level-oxygen-353005.examen_final.accident`
           WHERE RUR_URB = 2 and DAY_WEEK = 1 and HOUR >= 0
                 and HOUR < 3 and FATALS > 1"

kable(dbGetQuery(dades, query9, n = 10), format = "simple")
```

ST_CASE
482927
90244
180779
420767
62259
280568

10. Llisteu els accidents (sense repetició) on almenys hi hagi un vehicle involucrat amb una velocitat observada (TRAV_SP) superior a 110.

```
query10 <- "SELECT ST_CASE
            FROM `level-oxygen-353005.examen_final.vehicle`
            WHERE TRAV_SP > 110 AND TRAV_SP < 900 and PREV_SP < 900
            GROUP BY ST_CASE"

kable(dbGetQuery(dades, query10, n = 10), format = "simple")
```

ST_CASE
371227
330093
290807
122474
210641

11. Creeu una sèrie temporal de 31 observacions, on per cada dia digui el nombre d'accidents ocorreguts amb víctimes.

```
query11 <- "SELECT DAY, COUNT(*) AS FREQ
            FROM `level-oxygen-353005.examen_final.accident`
            GROUP BY DAY"

kable(dbGetQuery(dades, query11, n = 10), format = "simple")
```

DAY	FREQ
1	72
2	84
3	108
4	96
5	117
6	112
7	78
8	81
9	88
10	85

12. Busqueu si hi ha casos en aquestes dades on al mateix dia a la mateixa hora i minut hi ha hagut tres accidents o més.

```
query12 <- "SELECT DAY, HOUR, MINUTE
            FROM `level-oxygen-353005.examen_final.accident`
            GROUP BY DAY, HOUR, MINUTE
            HAVING COUNT(*) > 2"

kable(dbGetQuery(dades, query12, n = 10), format = "simple")
```

DAY	HOUR	MINUTE
1	17	0
6	99	99
12	99	99
22	99	99
23	15	50

13. ¿Quants accidents tenen l'identificador de la localització desconegut?

```
query13 <- "SELECT ST_CASE
            FROM `level-oxygen-353005.examen_final.accident`
            WHERE RUR_URB = 9"

kable(dbGetQuery(dades, query13, n = 10), format = "simple")
```

ST_CASE
420599
390987
390999

ST_CASE
40826
490259
240440
240443

14. Quants accidents involucren almenys dues persones de més de 90 anys? Compte: No incloure les persones d'edat no registrada o desconeguda.

```
query14 <- "SELECT ST_CASE, COUNT(*)
            FROM `level-oxygen-353005.examen_final.person`
            WHERE AGE BETWEEN 90 AND 900
            GROUP BY ST_CASE
            HAVING COUNT(*) > 1"

kable(dbGetQuery(dades, query14, n = 10), format = "simple")
```

ST_CASE	f0_
421087	3
550512	2