

Sesión 5: Algoritmo EM + Consultas SQL en Dbeaver mediante conexión a servidor

Aplicaciones en Computación Estadística

Natalie Julian - www.nataliejulian.com

Estadística UC y Data Scientist en Zippedi Inc.

Extensiones del algoritmo EM: Aproximación estocástica del método EM

SAEM: Stochastic Approximation of EM

Se realizó un seguimiento a vacas, y se posee información sobre la evolución del peso (en kg). El peso de cada vaca se midió de 9 o 10 ocasiones. Se quiere utilizar un modelo exponencial para describir el aumento de peso con el tiempo:

$$y_{ij} = A_i(1 - B_i)e^{-K_it_{ij}} + \epsilon_{ij}$$

Para cada vaca i se tiene:

- El regresor t_{ij} que corresponde al tiempo en días
- Tres variables
 - ▶ Año de nacimiento (entre 1988 y 1998)
 - ▶ Si tiene gemelo (al momento de nacer, nacieron dos o más)
 - ▶ Rango de nacimiento (entre 3 a 7)

Los datos `cow.saemix` se encuentran en la librería `saemix`.

- a) Dada la naturaleza del problema, ¿por qué sería natural tratar estos datos desde la perspectiva estocástica?

SAEM: Stochastic Approximation of EM

- a) Dada la naturaleza del problema, ¿por qué sería natural tratar estos datos desde la perspectiva estocástica?

Se realizó un seguimiento de estas vacas durante un período, es decir, se poseen registros durante varios puntos en el tiempo de las distintas unidades experimentales. Por lo tanto, se esperaría que existan efectos o residuos en los resultados observados de los registros anteriores.

SAEM es un algoritmo iterativo que consiste en construir N cadenas de Markov $(\psi_1^{(k)}), \dots, (\psi_1^{(N)})$ que convergen a la distribución condicional $p(\psi_1|y_1), \dots, p(\psi_N|y_N)$.

SAEM: Stochastic Approximation of EM

- b) Cargue el paquete `saemix` defina los datos en un objeto de tipo `saemixData` como sigue:

```
data(cow.saemix)

saemixcow<-saemixData(cow.saemix, #Datos
                      header=TRUE,
                      name.group=c("cow"), #ID de cada cadena
                      name.predictors=c("time"), #Predictores
                      name.response=c("weight"), #Nombre de la variable respuesta
                      name.covariates=c("birthyear","twin","birthrank"), #Otras variables
                      units=list(x="days",y="kg",covariates=c("yr","-","-"))) #Se definen las unidades
```

- c) Defina la estructura del modelo propuesto, como una función:

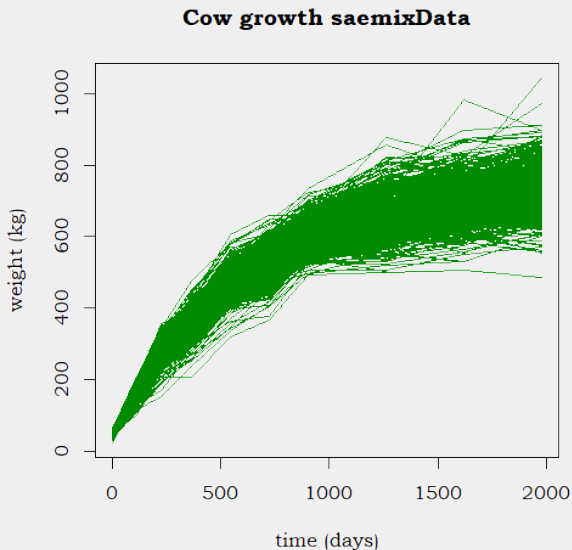
```
growthcow<-function(psi,id,xidep) {

  x<-xidep[,1]
  A<-psi[id,1]
  b<-psi[id,2]
  k<-psi[id,3]

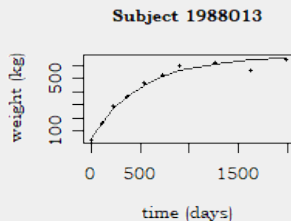
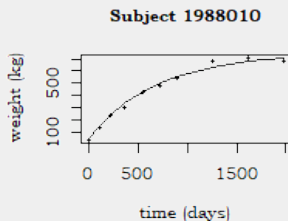
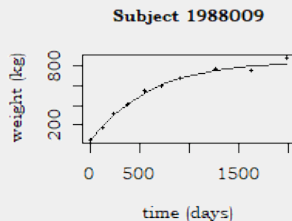
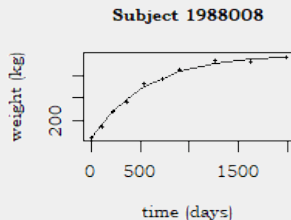
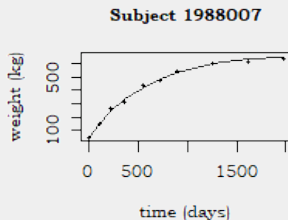
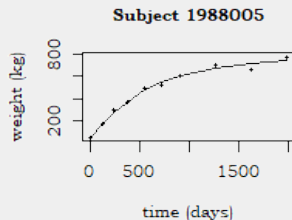
  f<-A*(1-b*exp(-k*x))
  return(f)
}
```

- d) Pruebe el algoritmo con valores iniciales, observe las salidas. ¿El modelo planteado tiene un buen ajuste?

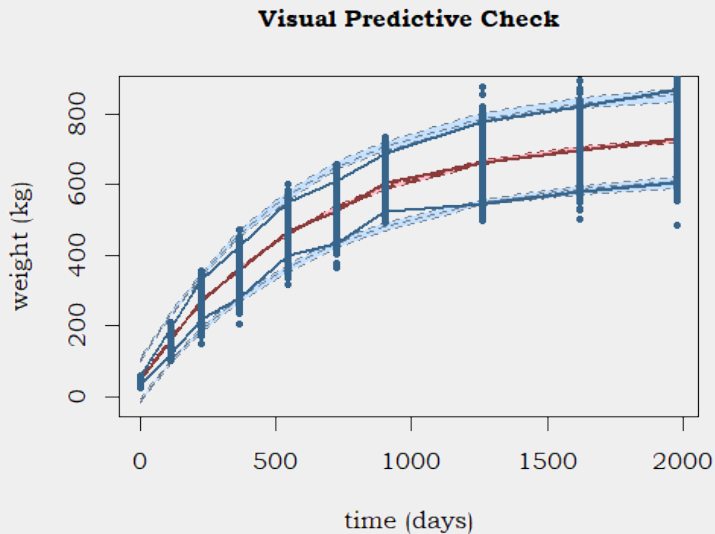
SAEM: Stochastic Approximation of EM



SAEM: Stochastic Approximation of EM



SAEM: Stochastic Approximation of EM



Consultas en DBeaver y R

La base de datos contiene información sobre vuelos de Nueva York en el año 2013. También incluye "metadatos" útiles sobre aerolíneas, aeropuertos, clima y planes.

La base de datos contiene información sobre vuelos de Nueva York en el año 2013. También incluye "metadatos" útiles sobre aerolíneas, aeropuertos, clima y planes.

¿Cómo accederemos a estos datos?

La base de datos contiene información sobre vuelos de Nueva York en el año 2013. También incluye "metadatos" útiles sobre aerolíneas, aeropuertos, clima y planes.

¿Cómo accederemos a estos datos?

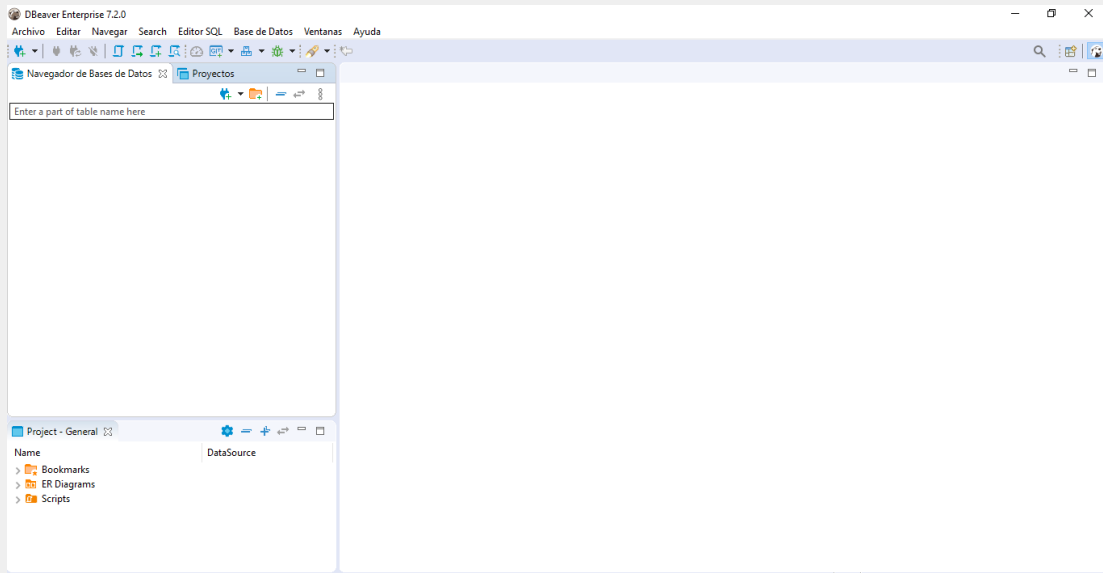
Mediante conexión a un servidor.

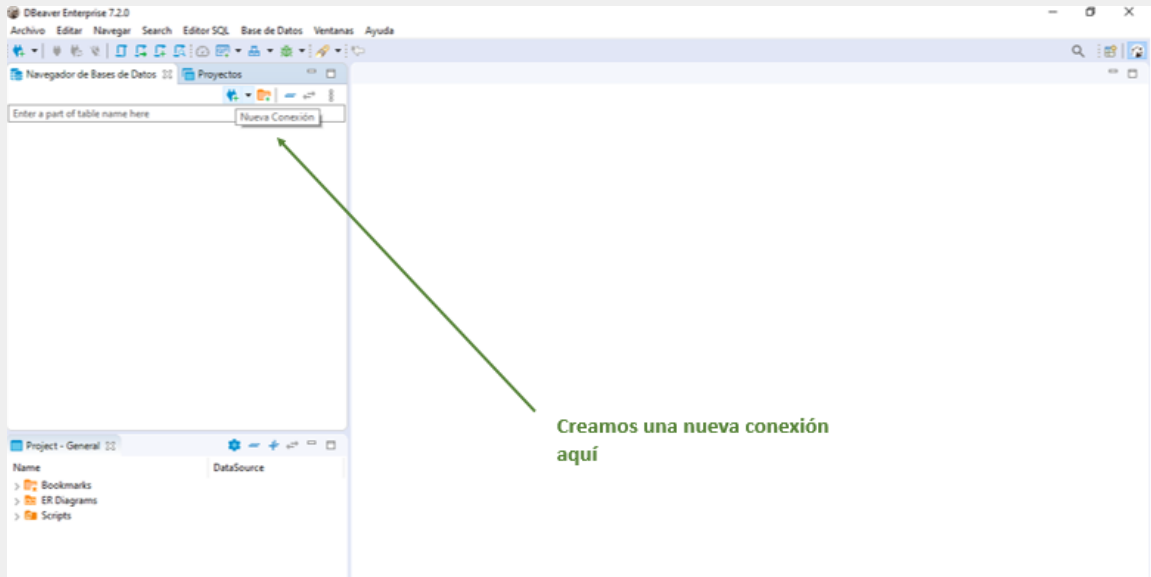
DBeaver es un gestor universal de BBDD multiplataforma, que ofrece soporte a las bases de datos más conocidas del mercado (MySQL, Oracle, DB2, SQL Server, PostgreSQL, etc ..) , así como algunas NoSQL (MongoDB, Cassandra).

Características de Dbeaver:

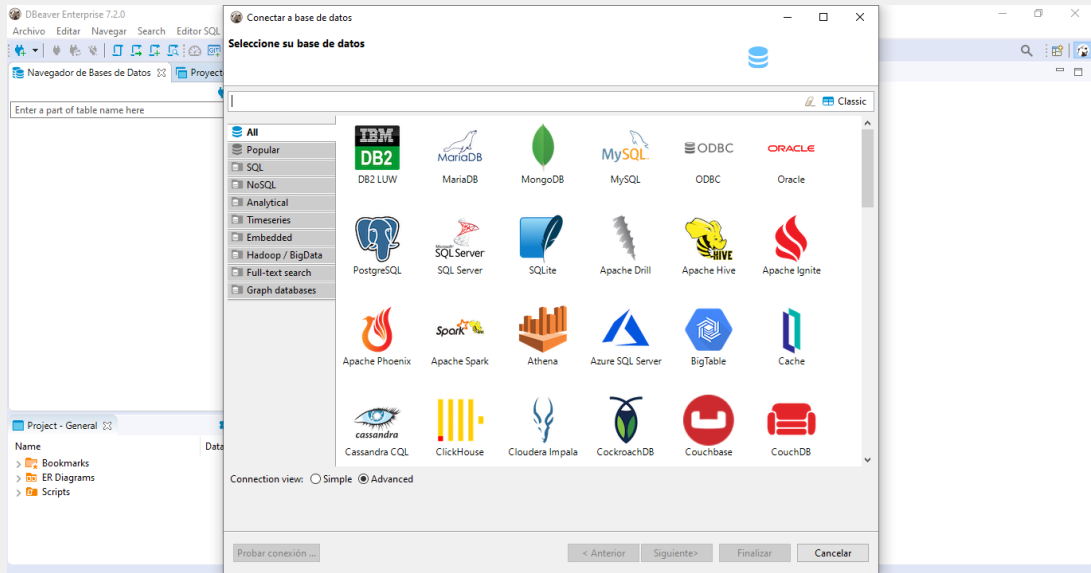
- Open source
- Autocompletado en el editor (símil a RStudio)
- Navegar por la estructura de BBDD (Restricciones, llaves foráneas, etcétera)
- Permite ejecutar scripts SQL

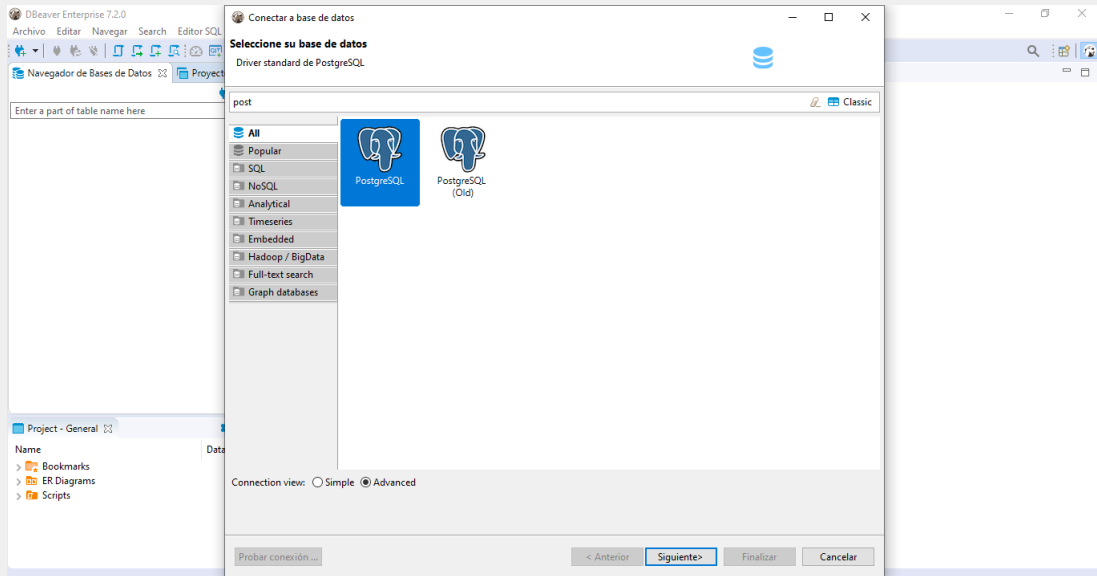
Instalación: <https://dbeaver.io/download/>.

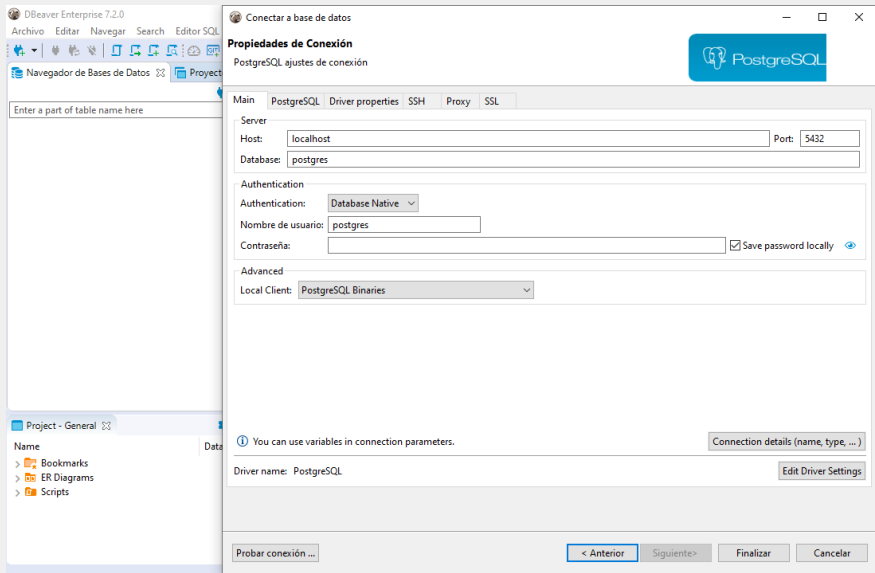




Creamos una nueva conexión
aquí







Datos de la conexión:

Host: db-edu.pacha.dev

Port: 5432

User: student

Pass: tx5mvyRQqD


Database: nycflights13

Especiales agradecimientos a Mauricio Vargas, Scoring Analyst en Banco Ripley y Data Visualization Teacher UC quién provee este servidor y datos.

Conectar a base de datos

Propiedades de Conexión

PostgreSQL ajustes de conexión



MainPostgreSQLDriver propertiesSSHProxySSL

Server

Host:db-edu.pacha.devPort:5432

Database:nycflights13

Authentication

Authentication:Database Native

Nombre de usuario:student

Contraseña:Save password locally

Advanced

Local Client:PostgreSQL Binaries

You can use variables in connection parameters.

Connection details (name, type, ...)

Driver name: PostgreSQL

Edit Driver Settings

Probar conexión ...

< Anterior

Siguiente >

Finalizar

Cancelar

Navegador de Bases de Datos

Proyectos

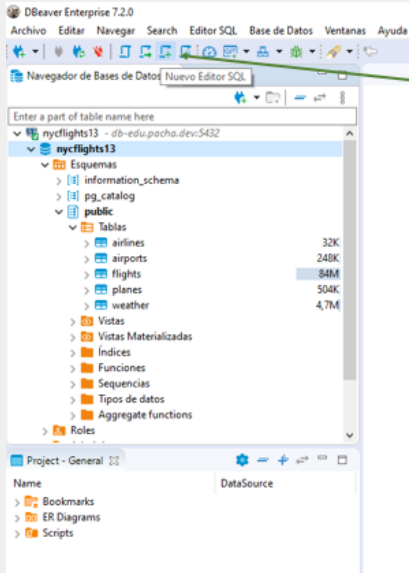
Enter a part of table name here

- ▼ nycflights13 - db-edu.pacha.dev:5432
 - ▼ nycflights13
 - ▼ Esquemas
 - > information_schema
 - > pg_catalog
 - ▼ public
 - ▼ Tablas
 - > airlines 32K
 - > airports 248K
 - > flights 84M
 - > planes 504K
 - > weather 4,7M
 - > Vistas
 - > Vistas Materializadas
 - > Índices
 - > Funciones
 - > Secuencias
 - > Tipos de datos
 - > Aggregate functions
 - > Roles

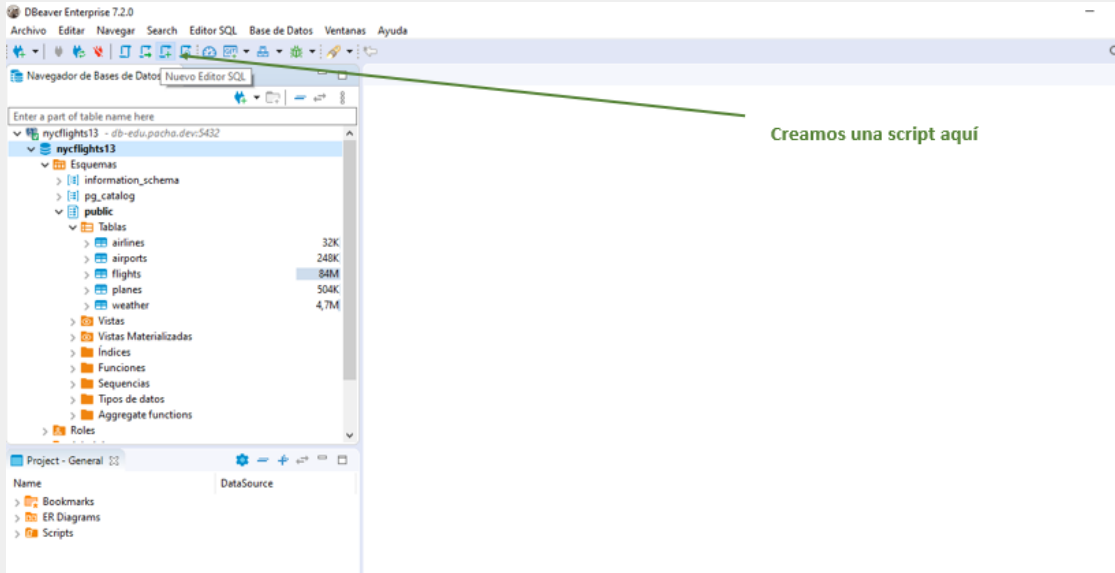
Project - General

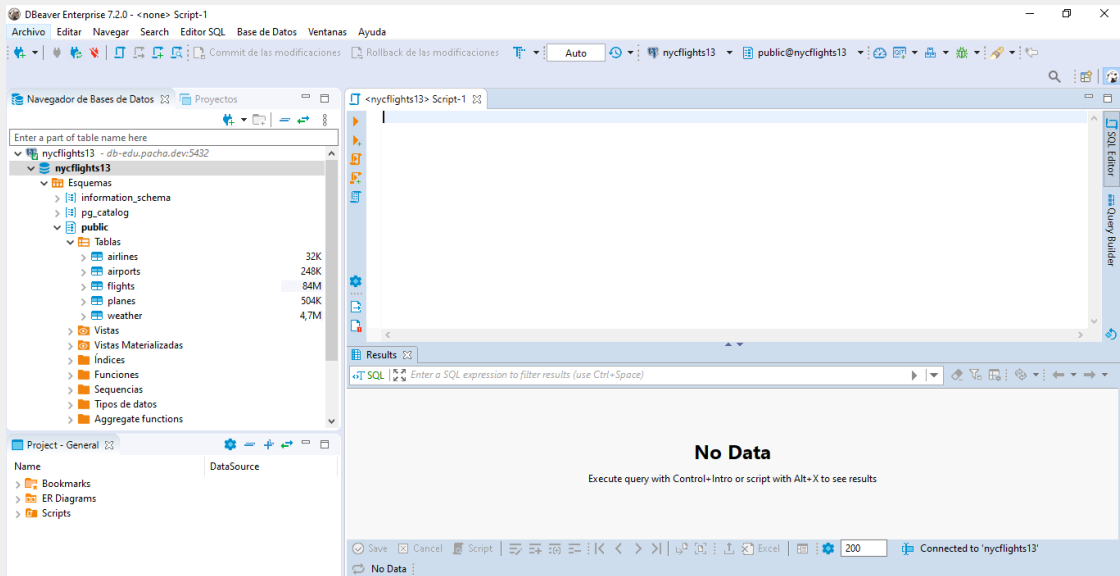
Name DataSource

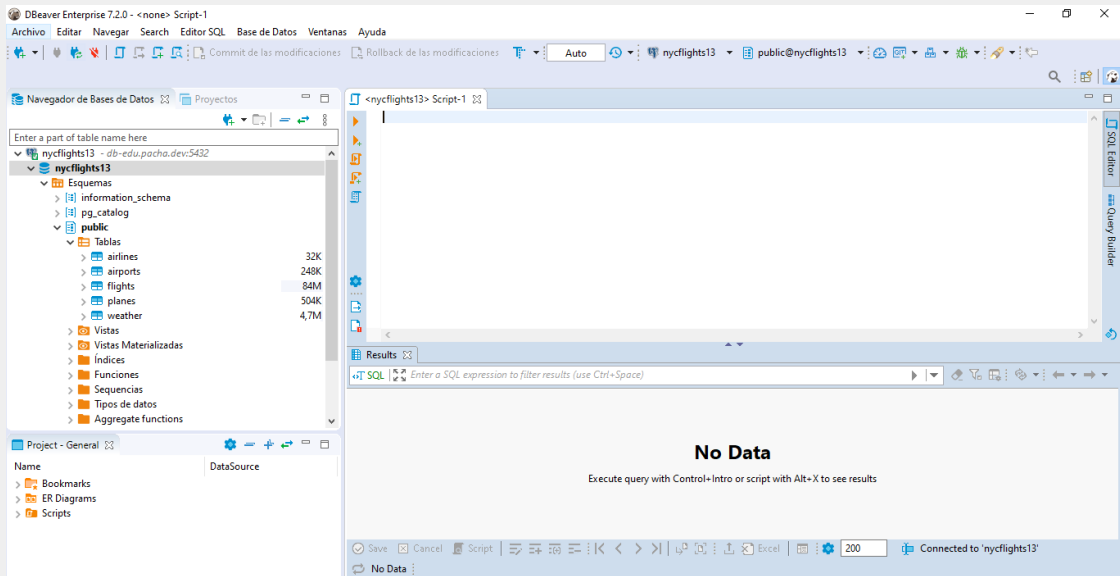
- > Bookmarks
- > ER Diagrams
- > Scripts



Creamos una script aquí







DBeaver Enterprise 7.2.0 - <nycflights13> Script-1

Archivo Editar Navegar Search Editor SQL Base de Datos Ventanas Ayuda

Commit de las modificaciones Rollback de las modificaciones Auto nycflights13 public@nycflights13

Navegador de Bases de Datos Proyectos

Enter a part of table name here

nycflights13 - db-edu.pacha.dev:5432

nycflights13

- Esquemas
 - information_schema
 - pg_catalog
 - public
 - Tablas
 - airlines 32K
 - Columnas
 - carrier (text)
 - name (text)
 - Restricciones
 - Claves foráneas
 - Índices
 - Dependencias
 - Referencias
 - Particiones
 - Disparadores
 - Reglas

Project - General

Name DataSource

- Bookmarks
- ER Diagrams
- Scripts

*<nycflights13> Script-1

```
select name
from airlines |
```

Results

SQL Enter a SQL expression to filter results (use Ctrl+Space)

No Data

Execute query with Control+Intro or script with Alt+X to see results

DBEaver Enterprise 7.2.0 - <nycflights13> Script-1

Archivo Editar Navegar Search Editor SQL Base de Datos Ventanas Ayuda

Commit de las modificaciones Rollback de las modificaciones Auto nycflights13 public@nycflights13

Navegador de Bases de Datos Proyectos

Enter a part of table name here

nycflights13 - db-edu.pacha.dev:5432

nycflights13

- Esquemas
 - information_schema
 - pg_catalog
 - public
 - Tablas
 - airlines 32K
 - Columnas
 - carrier (text)
 - name (text)
 - Restricciones
 - Claves foráneas
 - Índices
 - Dependencias
 - Referencias
 - Particiones
 - Disparadores
 - Reglas

Project - General

Name DataSource

- Bookmarks
- ER Diagrams
- Scripts

<nycflights13> Script-1

```
select name
from airlines
```

Aquí podemos correr una línea

Results

SQL Enter a SQL expression to filter results (use Ctrl+Space)

No Data

Execute query with Control+Intro or script with Alt+X to see results

Commit de las modificaciones Rollback de las modificaciones Auto

Navegador de Bases de Datos Proyectos

Enter a part of table name here

nycflights13 - db-edu.pacha.dev:5432

nycflights13

- Esquemas
 - information_schema
 - pg_catalog
 - public
 - Tablas
 - airlines 32K
 - Columnas
 - carrier (text)
 - name (text)
 - Restricciones
 - Claves foráneas
 - Índices
 - Dependencias
 - Referencias
 - Particiones
 - Disparadores
 - Reglas

Project - General DataSource

Bookmarks ER Diagrams Scripts

<nycflights13> Script-1

```
select name
from airlines |
```

Results

select name from airlines Enter a SQL expression to filter results (use Ctrl+Space)

Load AccessMethodCache - 2.9s

Cancel

Archivo Editar Navegar Search Editor SQL Base de Datos Ventanas Ayuda

Commit de las modificaciones Rollback de las modificaciones Auto nycflights13 public@nycflights13

Navegador de Bases de Datos Proyectos

Enter a part of table name here

nycflights13 - db-edu.pacha.dev:5432

nycflights13

- Esquemas
 - information_schema
 - pg_catalog
 - public
 - Tablas
 - airlines 32K
 - Columnas
 - carrier (text)
 - name (text)
 - Restricciones
 - Claves foráneas
 - Índices
 - Dependencias
 - Referencias
 - Particiones
 - Disparadores
 - Reglas

Project - General

Name DataSource

- Bookmarks
- ER Diagrams
- Scripts

*<nycflights13> Script-1

```
select name
from airlines
```

airlines

select name from airlines Enter a SQL expression to filter results (use Ctrl+Space)

Grid	ABC name	Value
1	Endeavor Air Inc.	Endeavor Air Inc.
2	American Airlines Inc.	
3	Alaska Airlines Inc.	
4	JetBlue Airways	
5	Delta Air Lines Inc.	
6	ExpressJet Airlines Inc.	
7	Frontier Airlines Inc.	
8	AirTran Airways Corporation	

Selecciona de la tabla airlines la columna name

DBBeaver Enterprise 7.2.0 - <nycflights13> Script-2

Archivo Editar Navegar Search Editor SQL Base de Datos Ventanas Ayuda

Commit de las modificaciones Rollback de las modificaciones Auto nycflights13 public@nycflights13

Navegador de Bases de Datos Proyectos

Enter a part of table name here

- information_schema
- pg_catalog
- public
 - Tablas
 - airlines 32K
 - airports 248K
 - flights 84M
 - Columnas
 - year (int4)
 - month (int4)
 - day (int4)
 - dep_time (int4)
 - sched_dep_time (int4)
 - dep_delay (float8)
 - arr_time (int4)
 - sched_arr_time (int4)
 - arr_delay (float8)
 - carrier (text)
 - flight (int4)

Project - General

Name DataSource

Bookmarks ER Diagrams Scripts

*<nycflights13> Script-1

```
select count(*)  
from flights
```

*<nycflights13> Script-2

Result

select count(*) from flights Enter a SQL expression to filter results (use Ctrl+Space)

Grid	1	count
	1	336.776

Value

336776

Cuenta cuántos registros hay en la tabla flights

DBEAVER Enterprise 7.2.0 - <nycflights13> Script-3

Archivo Editar Navegar Search Editor SQL Base de Datos Ventanas Ayuda

Commit de las modificaciones Rollback de las modificaciones Auto nycflights13 public@nycflights13

Navegador de Bases de Datos

Enter a part of table name here

- information_schema
- pg_catalog
- public
 - Tablas
 - airlines 32K
 - airports 248K
 - flights 84M
 - planes 504K
 - weather 4.7M
 - Columnas
 - origin (text)
 - year (int4)
 - month (int4)
 - day (int4)
 - hour (int4)
 - temp (float8)
 - dewp (float8)
 - humid (float8)
 - wind_dir (float8)

Project - General

Name DataSource

Bookmarks ER Diagrams Scripts

SQL Editor

```
select count(*)
from weather
where humid>60
```

Result

Enter a SQL expression to filter results (use Ctrl+Space)

count	Value
13707	13707

1 row(s) fetched - 535ms

Save Cancel Script Excel

200 1 Rows: 1

Cuenta cuántos registros de la tabla weather tenían humid superior a 60

Conexiones en RStudio

Realizamos una conexión para Postgres en RStudio de la siguiente forma:

```
library(RPostgres)
```

```
conexion <- dbConnect(  
  Postgres(),  
  user = "student",  
  password = "tx5mvyRQqD",  
  dbname = "nycflights13",  
  host = "db-edu.pacha.dev"  
)
```

Visualizar las tablas

```
library(dplyr)
```

```
tbl(conexion, "airlines") #Extrae una previsualización de la tabla
```

```
# Source:   table<airlines> [?? x 2]
```

```
# Database: postgres [student@db-edu.pacha.dev:5432/nycflights13]
```

```
  carrier name  
  <chr>    <chr>  
1 9E      Endeavor Air Inc.  
2 AA      American Airlines Inc.  
3 AS      Alaska Airlines Inc.  
4 B6      JetBlue Airways  
5 DL      Delta Air Lines Inc.  
6 EV      ExpressJet Airlines Inc.  
7 F9      Frontier Airlines Inc.  
8 FL      AirTran Airways Corporation  
9 HA      Hawaiian Airlines Inc.  
10 MQ     Envoy Air  
# ... with more rows
```

```
#Si la queremos extraer completa:
```

```
tbl(conexion, "airlines") %>%  
  collect()
```

Querys con dplyr

Nuestra última query en DBeaver fue:

```
select count(*)  
from weather  
where humid>60
```

Equivalente en RStudio a:

```
tbl(conexion, "weather") %>%  
  filter(humid>60) %>%  
  count()
```

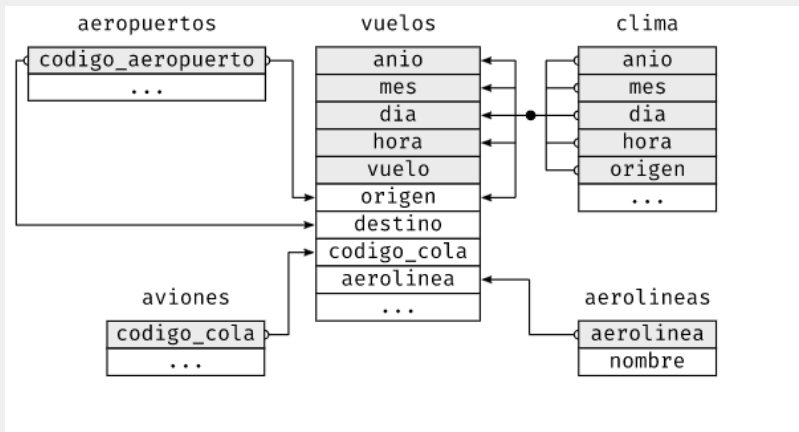
Hack de dplyr para aprender SQL

```
tbl(conexion, "weather") %>%  
  filter(humid>60) %>%  
  count() %>%  
  show_query()
```

```
<SQL>  
SELECT COUNT(*) AS "n"  
FROM "weather"  
WHERE ("humid" > 60.0)
```

Con `show_query()` podemos extraer la consulta en SQL, pegarla en DBeaver y funcionará!

Ejemplo con cruces de tablas



Crucemos las tablas flights y planes:

Ejemplo con cruces de tablas

```
tbl(conexion, "flights") %>%  
  count() #336776 filas  
# Source:   lazy query [?? x 1]  
# Database: postgres [student@db-edu.pacha.dev:5432/nycflights13]  
n  
<int64>  
1 336776
```

```
tbl(conexion, "planes") %>%  
  count() #3322 filas  
# Source:   lazy query [?? x 1]  
# Database: postgres [student@db-edu.pacha.dev:5432/nycflights13]  
n  
<int64>  
1 3322
```

```
intersect(colnames(tbl(conexion, "flights")), colnames(tbl(conexion, "planes")))  
[1] "year"      "tailnum"
```

```
tbl(conexion, "flights") %>%  
  right_join(.,tbl(conexion, "planes"), by="tailnum")%>%  
  show_query()
```

<SQL>

```
SELECT "LHS"."year" AS "year.x", "LHS"."month" AS "month", "LHS"."day" AS "day", "LHS"."dep_time" AS "dep_time", "LHS"."sched_dep_time" AS "sched_dep_time"  
FROM "flights" AS "LHS"  
RIGHT JOIN "planes" AS "RHS"  
ON ("LHS"."tailnum" = "RHS"."tailnum")
```

En DBeaver

DBeaver Enterprise 7.2.0 - <none> Script-5

Archivo Editar Navegar Search Editor SQL Base de Datos Ventanas Ayuda

Commit de las modificaciones Rollback de las modificaciones Auto nycflights13 public@nycflights13

Navegador de Bases de Datos Proyectos

Enter a part of table name here

- information_schema
- pg_catalog
- public
 - Tablas
 - airlines 32K
 - airports 248K
 - flights 84M
 - planes 504K
 - Columnas
 - tailnum (text)
 - year (int4)
 - type (text)
 - manufacturer (text)
 - model (text)
 - engines (int4)
 - seats (int4)
 - speed (int4)
 - engine (text)
 - Restricciones

Project - General

Name

- Bookmarks
- ER Diagrams
- Scripts

DataSource

*nycflights13> Script... *nycflights13> Script... *nycflights13> Script... *nycflights13> Script... *nycflights13> Script...

```
SELECT "LHS"."year" AS "year.x", "LHS"."month" AS "month", "LHS"."day" AS "day", "LHS"."dep_time" AS "dep_time", "LHS"."s" AS "s"
FROM "flights" AS "LHS"
RIGHT JOIN "planes" AS "RHS"
ON ("LHS"."tailnum" = "RHS"."tailnum")
```

flights(+)

Enter a SQL expression to filter results (use Ctrl+Space)

123 year.x 123 month 123 day 123 dep_time 123 sched_dep_time 123 dep_delay 123 arr_time 12

	year.x	month	day	dep_time	sched_dep_time	dep_delay	arr_time
1	2.013	10	1	539	545	-6	801
2	2.013	10	1	539	545	-6	917
3	2.013	10	1	544	550	-6	912
4	2.013	10	1	549	600	-11	653
5	2.013	10	1	550	600	-10	648
6	2.013	10	1	550	600	-10	649
7	2.013	10	1	551	600	-9	727

200 row(s) fetched - 1.300s (+21ms)

200 200+ Rows: 1

CLT es Escribible Inserción inteligente 4:39:10 Set 0 | 0

Finalmente...

Es necesario desconectarnos del servidor o estaremos consumiendo recursos. Basta utilizar:

```
dbDisconnect(conexion).
```