NYC Taxicab : Data Operations

Patrick Frenett

7/8/2016

Formatting 2009-2014 data.

- For the years 2009-2014, PostgreSQL has a hard time reading it in. This is due to two reasons :
  - The files contain 'invisible' characters that don't read into SQL.
  - Line 2 is blank.
- To clean up unwanted characters
  - To be run on Windows PERL client OR Mac OS Terminal :
  - `perl -pi.bak -e 's/[\000-\007\013-\037\177-\377]//g;' document.csv`
- To skip the 2nd row
  - Create a file named skipper.pl with contents: `#!/usr/bin/perl -w use strict; my $line_to_skip = 2; my $i = 0; while(<>) {print if ++$i != $line_to_skip;}`
  - To be run on Windows PERL client OR Mac OS Terminal :
  - `perl skipper.pl data.csv > other.csv`

Reverse geocoding NYC `long/lat` to zip codes.

- Import geojson file to QGIS
  - For this to be done, the file must first be loaded to pgAdmin which will read the json file and then create a geometry column so it can be added/manipulated by PostGIS. A connection between the two pieces of software will have to be made.
- Import point data to pgAdmin
  - Create a new table and specify the columns of the point data (including `long/lat`)
  - `COPY table FROM 'file\path.csv' CSV HEADER;`
- Create `location` and `zip` columns
  - `location` should be saved as public.geometry and `zip` should be a character string of length 5
- Populate `location` so that it will work with PostGIS
  - `UPDATE data SET location = ST_SetSRID(ST_MakePoint(longitude, latitude),4326)`
  - 4326 is the code for WGS 84, the global reference system used in GPS navigation
- Populate `zip` using PostGIS
  - `UPDATE data SET zip = zip_codes.postalcode FROM zip_codes WHERE ST_Within(data.location, zip_codes.geom)`
- Export data to .csv file from pgAdmin
  - `COPY data TO 'file_location' DELIMITER ',' CSV HEADER`

SQL query for everything in PostgreSQL.

```
CREATE TABLE public.sample
(
  "VendorID" integer,
  tpep_pickup_datetime timestamp without time zone,
  tpep_dropoff_datetime timestamp without time zone,
  passenger_count integer,
  trip_distance real,
  pickup_longitude real,
  pickup_latitude real,
  "RatecodeID" integer,
  store_and_fwd_flag character(1),
  dropoff_longitude real,
  dropoff_latitude real,
  payment_type integer,
  fare_amount real,
  extra real,
  mta_tax real,
  tip_amount real,
  tolls_amount real,
  improvement_surcharge real,
  total_amount real
)
WITH (
  OIDS=FALSE
);
ALTER TABLE public.sample
  OWNER TO owner;

COPY sample FROM 'path/sample.csv' CSV HEADER;

ALTER TABLE sample ADD pickup_location geometry;
ALTER TABLE sample ADD dropoff_location geometry;
ALTER TABLE sample ADD pickup_zip CHAR(5);
ALTER TABLE sample ADD dropoff_zip CHAR(5);

UPDATE sample SET pickup_location = ST_SetSRID(ST_MakePoint(pickup_longitude, pickup_latitude),4326);
UPDATE sample SET dropoff_location = ST_SetSRID(ST_MakePoint(dropoff_longitude, dropoff_latitude),4326)

UPDATE sample SET pickup_zip = zip_codes.postalcode FROM zip_codes WHERE ST_Within(sample.pickup_locati
UPDATE sample SET dropoff_zip = zip_codes.postalcode FROM zip_codes WHERE ST_Within(sample.dropoff_loca

COPY (SELECT "tpep_pickup_datetime", "pickup_longitude", "dropoff_latitude" FROM sample ORDER BY tpep_p

DROP TABLE sample;
```