Spatial Database Management
GEP 664 /GEP 380
Class #9: Proximity analysis

Frank Donnelly

Dept of EEGS, Lehman College CUNY

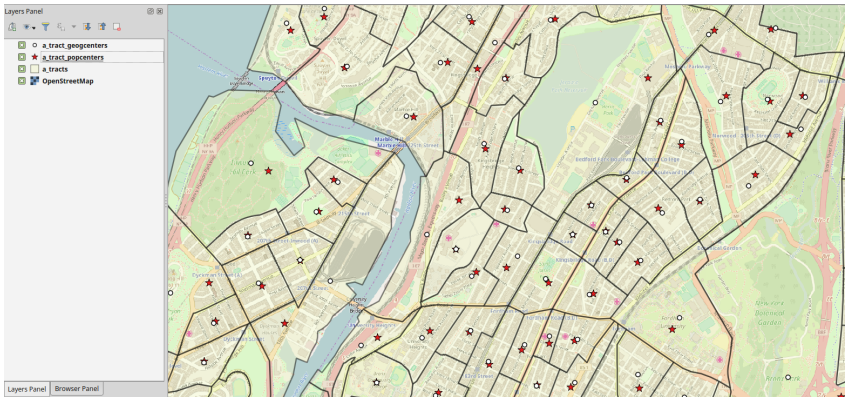
Spring 2019

Today's Topics

Population Centroids or Centers

nyc.tract_popctr

The following examples use census tract population centroids, which represent the center of a population's distribution (red stars) rather than the geographic centroid (white circles).



Create Spatial Index

Remember to create a spatial index for layers to speed up processing. Automatic when loading shapefiles, but must be done manually after building geometry from coordinates or when inserting existing geometry into a new table.

```
CREATE INDEX idx_tract_popctr_geom
ON nyc.tract_popctr
USING gist (geom);
```

Buffers
Buffers

Standard GIS approach of selecting geometry within a distance of another geometry - but a slow process. Selecting all census tract population centroids that fall within 1/2 mile (2,640 ft) of a subway station:

```
SELECT DISTINCT t.geoid, t.tract, t.pop2010
FROM nyc.tract_popctr t, nyc.subway_stations s
WHERE ST_Intersects(t.geom, ST_Buffer(s.geometry, 2640));
```

NOTE - DISTINCT is important if you want to eliminate duplicates (tract centers that intersect more than one buffer).

Creating Buffer Layer I
Buffers

Previous example is on the fly. If you need to create buffers as actual features: this creates overlapping circles where each buffer has the id of the subway station:

```
CREATE TABLE nyc.subway_buffer (
stop_id varchar(3) PRIMARY KEY,
geom geometry (polygon,2263));

INSERT INTO nyc.subway_buffer (stop_id,geom)
SELECT stop_id, ST_BUFFER(geometry,2640)
FROM nyc.subway_stations;
```

Creating Buffer Layer II
Buffers

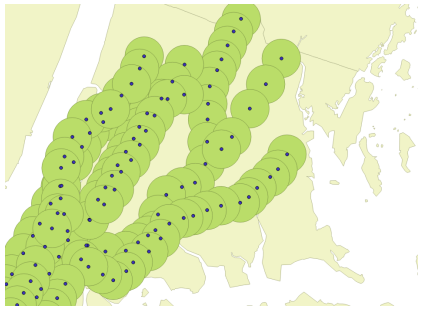
Alternatively - this dissolves overlapping buffers to create one unified buffer feature:

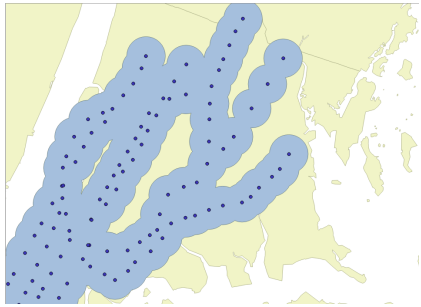
```
CREATE TABLE nyc.subway_buffer2 (
buf_id serial PRIMARY KEY,
geom geometry (multipolygon,2263));

INSERT INTO nyc.subway_buffer2 (geom)
SELECT ST_Union(ST_BUFFER(geometry,2640))
FROM nyc.subway_stations;
```

(You could also create a view instead of a table)

Two Buffer Types
Buffers





Measure Distance Within

Basic Distances

Do NOT use buffers unless you need to. ST_DWithin is much more efficient. All census tract population centers within a distance of 1/2 mile (2,640 ft) from subway stations:

```
SELECT DISTINCT t.geoid, t.tract, t.pop2010
FROM nyc.tract_popctr t, nyc.subway_stations s
WHERE ST_DWithin(t.geom, s.geometry, 2640);
```

Total population within 1/2 mile of a subway station:

```
SELECT SUM(pop2010) AS subwaypop
FROM nyc.tract_popctr
WHERE geoid IN (
SELECT DISTINCT t.geoid
FROM nyc.tract_popctr t, nyc.subway_stations s
WHERE ST_DWithin(t.geom, s.geometry, 2640));
```

Measuring Distance

Basic Distances

Use ST_Distance to calculate straight-line distance from one feature to another.

```
SELECT s.stop_id, s.stop_name, s.trains, z.zcta,
ROUND(ST_Distance(s.geometry, z.geom)) AS dist
FROM nyc.subway_stations s, nyc.zctas z
WHERE s.stop_id='405'
ORDER BY dist;
```

	stop_id character varying(3)	stop_name character varying(38)	trains character varying(13)	zcta character varying(254)	dist double precision
1	405	Bedford Park Bl	4	10468	0
2	405	Bedford Park Bl	4	10458	765
3	405	Bedford Park Bl	4	10463	1358
4	405	Bedford Park Bl	4	10467	1853
5	405	Bedford Park Bl	4	10470	4587
6	405	Bedford Park Bl	4	10471	4680
7	405	Bedford Park Bl	4	10469	5071
8	405	Bedford Park Bl	4	10034	5687

ST_DWithin vs ST_Distance

Basic Distances

ALWAYS do this to find *all* features within a certain distance:

```
SELECT DISTINCT t.geoid, t.tract, t.pop2010
FROM nyc.tract_popctr t, nyc.subway_stations s
WHERE ST_DWithin(t.geom, s.geometry, 2640);
```

NEVER do this:

```
SELECT DISTINCT t.geoid, t.tract, t.pop2010
FROM nyc.tract_popctr t, nyc.subway_stations s
WHERE ST_Distance(t.geom, s.geometry) <= 2640;
```

The former can take advantage of the spatial index, while the latter cannot. Use ST_Distance just for calculating distances, not for making selections.

Find Closest N Features From Edge

Basic Distances

Combination of Distance, DWithin, ORDER BY, and LIMIT to find the closest features to a feature. Subways within 1/2 mile from ZCTA 10468. For polygons, distance is measured from the edge...

```
SELECT s.stop_id, s.stop_name, s.trains,
ST_Distance(s.geometry, z.geom) AS dist
FROM nyc.subway_stations s, nyc.zctas z
WHERE ST_DWithin(s.geometry, z.geom, 2640)
AND z.zcta='10468'
ORDER BY dist
LIMIT 10;
```

	stop_id character varying(3)	stop_name character varying(38)	trains character varying(13)	dist double precision
1	405	Bedford Park Blvd -	4	0
2	407	Fordham Rd	4	0
3	406	Kingsbridge Rd	4	0
4	408	183 St	4	1.041613165
5	D05	Fordham Rd	B D	2.191219257
6	D03	Bedford Park Blvd	B D	2.417439484
7	D04	Kingsbridge Rd	B D	41.76483155
8	402	Mosholu Pkwy	4	278.4867870
9	D06	182-183 Sts	B D	530.8009488

Find Closest N Features From Center

Basic Distances

Calculate centroids to measure from the center of a polygon instead.

```
SELECT s.stop_id, s.stop_name, s.trains,
ST_Distance(s.geometry, ST_Centroid(z.geom)) AS dist
FROM nyc.subway_stations s, nyc.zctas z
WHERE ST_DWithin(s.geometry, ST_Centroid(z.geom), 2640)
AND z.zcta='10468'
ORDER BY dist';
```

	stop_id character varying(3)	stop_name character varying(38)	trains character varying(13)	dist double precision
1	406	Kingsbridge Rd	4	762.478833905
2	407	Fordham Rd	4	1974.54271352
3	D04	Kingsbridge Rd	B D	1813.81398471
4	D05	Fordham Rd	B D	2566.46794888

For Every Feature Find the Closest Other Feature

Advanced Distance Methods

Use special DISTINCT ON clause to generate a list of the closest feature to every feature. The DISTINCT ON identifier and the distance must appear in the ORDER BY clause.

For every census tract pop center, return closest subway station:

```
SELECT DISTINCT ON (t.geoid)
t.geoid, s.stop_id, s.stop_name,
ST_Distance(t.geom, s.geometry) AS dist
FROM nyc.tract_popctr t, nyc.subway_stations s
WHERE ST_DWithin(t.geom, s.geometry, 50000)
ORDER BY t.geoid, dist;
```

Nearest Neighbors KNN - Single Feature

Advanced Distance Methods

Use K-Nearest Neighbor operators to quickly retrieve closest neighbors. Pretty easy for N closest to a single feature. Closest 10 subways to Bronx tract 040701.

```
SELECT s.stop_id, s.stop_name, s.trains
FROM nyc.subway_stations s
ORDER BY s.geometry <->
(SELECT geom FROM nyc.tract_popctr t WHERE t.geoid='
    36005040701')
LIMIT 10;
```

	stop_id character varying(3)	stop_name character varying(38)	trains character varying(13)
1	405	Bedford Park Blvd - Lehman	4
2	D03	Bedford Park Blvd	B D
3	D04	Kingsbridge Rd	B D
4	406	Kingsbridge Rd	4
5	D01	Norwood - 205 St	D
6	402	Mosholu Pkwy	4
7	D05	Fordham Rd	B D
8	407	Fordham Rd	4
9	104	231 St	1
10	106	Marble Hill - 225 St	1

Nearest Neighbors KNN - Multiple Features

Advanced Distance Methods

K-Nearest Neighbor can also be used to quickly find the closest feature for every feature, and the N closest features to all features, but the statements are more complex. See sections 10.1.5 to 10.1.7 in PostGIS in Action.

Network Distance

Advanced Distance Methods

- Instead of straight-line distance, you many need to measure distance along or through a network.
- Roads, highways, railroads, etc.
- Can account for constraints like speed, direction, traffic.
- PostGIS has the pgRouting add-on for network distances (Chapter 16)

Applications for Distance

Geotagging

A point / polygon operation. Tag features based on what region they are in; for example assign airports a timezone or a country code based on where it's located. Achieved with spatial joins, intersections, or distance.

Linear Referencing

A point / line operation. Associate point with the closest line string, for example, snap or identify buildings to the closest street.

Today's Topics

Geography Type

- Models the spherical earth, as opposed to geometry which is 2D
- Uses only GCS systems. WGS 84 is the default
- Has fewer functions, some the same as geometry, some slightly different
- All tables are registered in the geography_columns view in the public schema
- Best use case is for geodetic measurements over long distances
- All measurements returned from the geography data type **are in meters**

Geodetic vs Cartesian Distance

Geography handles great circle distances. Geometry falls flat.

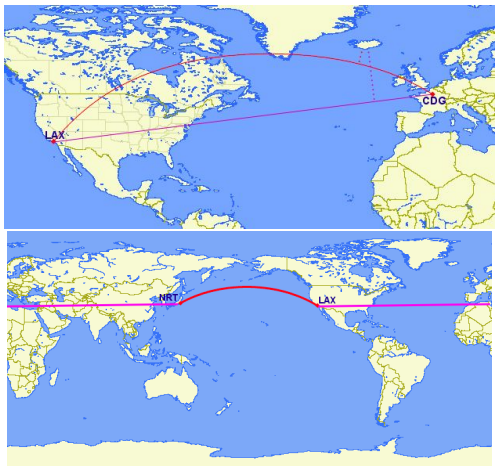


Image source: http://workshops.boundlessgeo.com/postgis-intro/geography.html

Creating Geography

```
CREATE TABLE airport_geog (
iata varchar(3) PRIMARY KEY,
state varchar(2),
lon numeric(9,6),
lat numeric(9,6),
geog geography(point,4326));
```

```
INSERT INTO airport_geog
VALUES ('JFK','NY',−73.7789,40.6397),
('IAD','VA',−77.4558, 38.9444),
('LAX','CA',−118.4081,33.9425),
('ANC','AK',−149.9983,61.1742),
('HNL','HI',−157.9225,21.3186);
```

```
UPDATE airport_geog
SET geog=ST_SetSRID(ST_Point(lon,lat),4326);
```

Geodetic Distance Between Airports

Measure distances between features within the same layer by adding the same table twice to the FROM statement, with a different alias for each. Omit measurements from a feature to itself by saying id of the feature should not equal id of the neighbor.

```
SELECT a.iata, n.iata,
ST_Distance(a.geog, n.geog)/1000 AS geog_dist
FROM airport_geog a, airport_geog n
WHERE a.iata='JFK' AND a.iata != n.iata;
```

	iata character varying(3)	iata character varying(3)	geog_dist double precision
1	JFK	IAD	366.8493479
2	JFK	LAX	3982.949776
3	JFK	ANC	5448.503489
4	JFK	HNL	8019.798634

Comparing Distance Results

The geometry distances could be improved by using an equidistant projection, but could only preserve distances from 1 to 1, or 1 to many locations. Cannot do many to many.

Geography using WGS 84 (EPSG 4326)

	iata character varying(3)	iata character varying(3)	geog_dist double precision
1	JFK	IAD	366.8493479
2	JFK	LAX	3982.949776
3	JFK	ANC	5448.503489
4	JFK	HNL	8019.798634

Geometry using WGS 84 / Pseudo-Mercator (EPSG 3857)

	iata character varying(3)	iata character varying(3)	geom_dist double precision
1	JFK	IAD	477.3572886
2	JFK	LAX	5055.921680
3	JFK	ANC	9258.997128
4	JFK	HNL	9702.339826

Proximity Summary

ST_Distance Measure distance between features

ST_DWithin Select all features within a specified distance of other features

ORDER BY & LIMIT Use with ST_DWithin to select the N closest or most distant features

DISTINCT ON Quickly generate a list with the closest feature for every feature

KNN Quickly retrieve N closest features to a single feature. Can also do closest and N closest to multiple features with more complex statements

Geography Type For geodetic distances between many distant points

Today's Topics

Measuring Distances

The Geography Type

Next Class

Due Next Class

The following are due at the beginning of our next class:

Assignment #9

Posted on the course website

Readings for Class #10

Listed in the syllabus, in *PostGIS In Action* and *Practical SQL*

Readings

Practical SQL - READ Chapter 18

PostGIS in Action - READ Chapter 14

But in this chapter you can skim or skip the following:

- 14.4.3 Using triggers