Finding nearest neighbors between two tables with point locations in SpatiaLite?

asked 6 years, 4 months ago
viewed 9,917 times
active 6 months ago

8

I have started playing with SpatiaLite today and already stumbled upon a problem.

For each point location stored in tableOne I would like to select one, nearest (linear distance) point from tableTwo.

So far I came up with a clumsy solution that utilizes VIEW:

7

```
CREATE VIEW testview AS
SELECT
A.id ,
B.myValue,
Distance(A.Geometry, B.Geometry) AS distance
FROM tableOne AS A, tableTwo AS B
WHERE distance < 10000
ORDER BY A.Id, distance;
```

And then:

```
SELECT * FROM testview
WHERE distance = (SELECT MIN(distance) FROM testview AS t WHERE t.id = testview.id)
```

seems to do the job.

Two questions:

Is there a way to perform such query without creating a VIEW?

Is there any other way to optimize this query for better performance? In a real world scenario tableOne will have hundreds-couple thousands records, and tableTwo - 1.3 million.

sql spatialite nearest-neighbor

share improve this question

edited Aug 1 '17 at 10:10
PolyGeo ♦
48.4k 16 72 208

asked Oct 11 '11 at 20:20
radek
7,755 8 69 110

I can give you an approach that is several orders of magnitude faster, but it would require you to use postgresql 9 knngist index instead of spatialite... – Ragi Yaser Burhum Oct 13 '11 at 2:20

actually faster than GRASS, ArcGIS, QGIS, SQLServer and pretty much any other spatial db/Desktop GIS (have not tried Oracle nearest neighbour functionality though).Just let me know if it is an option. – Ragi Yaser Burhum Oct 13 '11 at 2:23

@Ragi: I'm aware that PostGIS would be much more efficient way to work with such problem. However the ultimate goal of this exercise would be to make small portable app and in this case SpatiaLite is a winner. – radek Oct 13 '11 at 10:42

What's your development platform for your portable app? – Allan Adair Oct 13 '11 at 12:27

@Allan: Working on both: Windows Server 2008 & Ubuntu at the moment. – radek Oct 13 '11 at 14:07

show 4 more comments

Linked

3 Using nearest neighbor query in SpatiaLite?

1 SQLite db with FDO geometries: how to assign ID of point in layer A to closest point in Layer B with SpatiaLite?

Related

15 Optimize nearest neighbor query on 70 million point cloud on SQL Server 2008

1 Finding nearest point in SpatiaLite without GEOS

6 Finding all neighbors for all polygons in Spatialite

12 Find Nearest Line Segments to Point

2 Create Buffer in meters Spatialite query

3 Using nearest neighbor query in SpatiaLite?

1

4 Answers

active oldest **votes**

4

I have just tested this SQL and it works:

```
SELECT g1.OGC_FID As id1, g2.OGC_FID As id2, MIN(ST_Distance(g1.GEOMETRY,g2.GEOMETRY
FROM table_01 As g1, table_02 As g2
WHERE g1.OGC_FID <> g2.OGC_FID
AND ST_Contains(ST_Expand(g1.geometry,50),g2.geometry)
GROUP BY id1
ORDER BY id1
```

As you can read here "The naive way to carry out a nearest neighbour query is to order the candidate table by distance from the query geometry, and then take the record with the smallest distance".

Best regards,

Andrea

share improve this answer

answered Oct 31 '13 at 17:53

aborruso

613 4 16

I am trying to use this query but I am getting unexpected results - I am getting a resulting table but with IDs for lines I can see are not the nearest neighbor. I am trying to find the closest line in a multiline string layer to each point in another layer. I am new with spatiaLite. Any suggestions? Also, I ultimately want to run this on 1 million+ points – kflaw Jul 19 '15 at 17:00

I'm also not sure I understand the purpose of this statement: WHERE g1.OGC_FID <> g2.OGC_FID – kflaw Jul 19 '15 at 17:36

Also, in my result I am getting null distance. I have played around with this line: AND ST_Contains(ST_Expand(g1.geometry,50),g2.geometry) as well as removed it and still get no distance values, even though I am getting an ID – kflaw Jul 19 '15 at 18:44

add a comment

6

If you don't want to compute the distances between all the point combinations, you can use a spatial index on one of the table :

```
SELECT
  A.id ,
  B.myValue,
  MIN(Distance(A.Geometry, B.Geometry)) AS distance
FROM tableOne AS A, tableTwo AS B
WHERE A.ROWID IN (
  SELECT ROWID
  FROM SpatialIndex WHERE
    f_table_name = 'A'
    AND search_frame = BuildCircleMbr(ST_X(B.Geometry), ST_Y(B.Geometry), 10000))
GROUP BY A.id, B.myValue
```

share improve this answer

answered Oct 3 '13 at 10:20

Samuel

61 1 1

I've tried to use the solution you posted as I need to use a spatial index, but it returns no values? for the line `f_table_name = 'A'`, do I need to replace 'A' with the actual table name (table one)? I've tried either way and it still doesn't return anything, why might this be – kflaw Jul 29 '15 at 0:06

What is the quickest way to find in which plots a set of points lie?

3

Nearest neighbor analysis, write multiple columns to table

2

Nearest neighbors

3

ogr2ogr sql - returning points from one sqlite dataset that intersect with another

Hot Network Questions

- Was there ever any reason to wait 30 seconds to restart a c.1995 PC?
- Why are punch card readers no longer in use?
- How to talk to someone about being chronically late - not to fix it, just to get more accurate ETAs?
- Explicit default constructor
- What is wrong with magic strings?

more hot questions

You are right `f_table_name = 'A'` should be `f_table_name = 'tableOne'` . Note that this request assumes spatialite > 4.x (`SpatialIndex` virtual table is used). Did you try to adjust the `search_frame` for your use case ? In the example above, points are assumed to be at a maximum distance of 10000 meters. – Samuel Jul 30 '15 at 8:08

I did play around with the search frame value, I assume that means within 10000 meters which should work for me. I don't actually know which version of spatialite, I created the database through qgis and am using the gui in qgis. Let me see if i can figure that out – kflaw Jul 30 '15 at 14:28

It is version 4.1.1 with sqlite version 3.7.17, so it should work then? I wonder what is wrong I will test it some more – kflaw Jul 30 '15 at 15:30

add a comment

2

You can simplify your query like this.

```
SELECT
   A.id ,
   B.myValue,
   MIN(Distance(A.Geometry, B.Geometry)) AS distance
FROM tableOne AS A, tableTwo AS B
GROUP BY A.id, B.myValue
```

For a more generic solution, it might be worth trying to convert this PostGIS Nearest Neighbor function: http://blog.mackerron.com/2011/03/postgis-nearest-neighbour/

share improve this answer

edited Oct 13 '11 at 10:50

answered Oct 12 '11 at 20:30

underdark ♦

63.1k 11 160 306

unfortunately the code results in: `SQL error: "misuse of aggregate: MIN()"` – radek Oct 13 '11 at 10:38

As of PostGIS there are also some examples on BostonGIS website, but so far I wasn't successful in translating them into SpatiaLite :/ – radek Oct 13 '11 at 10:57

add a comment

2

Since version 4.4.0 SpatiaLite supports a KNN virtual table index for nearest neighbor problems. Here is a query the finds the nearest line in a linestring table to each point in a point table.

```
SELECT k.* FROM knn k, points p
WHERE f_table_name = 'linestrings'
AND ref_geometry = p.geometry
AND max_items = 1;
```

share improve this answer

answered Feb 7 '17 at 14:59

Greg Krakow

89 1

add a comment

Your Answer

Sign up or log in

Sign up using Google

Sign up using Facebook

Sign up using Email and Password

Post as a guest

Name

Email

required, but never shown

Post Your Answer

By posting your answer, you agree to the privacy policy and terms of service.

Not the answer you're looking for? Browse other questions tagged sql spatialite nearest-neighbor or ask your own question.

question feed