Research question:
The objective of this research is to develop creative ways of generating spatial SQL queries. We implement generation of two varieties of spatial queries for two different applications namely next query prediction and cardinality estimation.

1. Next query prediction:
An Interactive Data Exploration session primarily consists of a user interacting with a dataset to draw interesting insights/patterns by issuing a sequence of queries.

Human intent is where the queries become progressively more complex and converge towards a specific insight.

For example, a session of spatial queries issued by the user may look like this:

```
QUERY 1:
```sql
CREATE OR REPLACE VIEW trips AS
SELECT ST_PolygonFromEnvelope(pickup_longitude, pickup_latitude, dropoff_longitude, dropoff_latitude) as shape, 'I am a polygon'
FROM NYCTaxiTrips

QUERY 2:
```sql
SELECT ST POINT(dropoff_longitude, dropoff_latitude), passenger_count FROM NYCTaxiTrips
WHERE passenger_count > 5
```

2. Cardinality estimation:
Cardinality estimation has been an active topic of interest in recent years and among the useful and important types of metadata has been the number of distinct values in a column. We shall generate several independent spatial queries with each query having a different purpose.

There are three different types of queries we synthesize:

1. Proximity query: The number of points that lie within a certain X miles from a specific latitude, longitude point. Spatial predicates such as KNN JOIN are used.

2. Point-polygon Spatial overlap query: The number of points that overlap with a given polygon. Spatial predicates such as ST_Contains, ST_Overlaps are used.

3. Polygon-polygon Spatial overlap query: The number of polygons that overlap with a given polygon. Spatial predicates such as ST_Contains, ST_Overlaps are used.

References:


- ST_Contains
- ST_Intersects
- ST_Overlaps
- ST_Within
- ST_Equals
- ST_Crosses
- ST_Touches

Spatial predicate operators: