DSC 202 - Data Management for Data Science

Crime Analytics: Neo4J & PSQL Integration

Final Project Presentation

By :-
Kavya Sridhar
Shreyash Reddy
Susmit Singh

Instructor :-
Dr. Amarnath Gupta

Introduction: Context

Background and Motivation

Crime is a critical issue that affects public safety, economy, and urban development.

Data-driven crime analysis helps identify patterns, trends, and anomalies.

Modern technologies like graph databases enhance the ability to analyze complex relationships in crime data.

Why Crime Analysis Matters?

Helps in predictive policing by identifying high-risk areas.

Assists in resource allocation for law enforcement agencies.

Provides transparency and insights for policymakers and the public.

Main Goals of the Project:

Extract valuable insights from crime data using structured analysis.

Identify crime trends, correlations, and hidden patterns.

Develop an interactive system that enables efficient crime-related queries.

Introduction: Applications

Purpose

- Provide structured crime analysis through data-driven approaches.
- Help law enforcement and policymakers make informed decisions.
- Utilize graph-based representation for better data connectivity and insights.

Expected Outcomes

- Identification of crime hotspots and trends.
- Data-driven insights for pro-active crime prevention.
- Improved understanding of crime patterns for better law enforcement strategies.

Data Sources (1/2)

- The dataset used in this project comes from the City of London crime data repository at data.police.uk, covering crime records from ***2022 to 2024*** with over ***6,000 incidents***. *The dataset includes street-level crime data and outcome records.*
- The ***street-level crime*** files contains fields such as:
  - Crime ID
  - Month (of occurrence)
  - Reported by
  - Falls within (jurisdiction)
  - Longitude
  - Latitude
  - Location (descriptive)
  - LSOA code
  - LSAO name
  - Crime type
  - Last Outcome category

	Crime ID	Month	Reported by	Falls within	Longitude	Latitude	Location	LSOA code	LSOA name	Crime type	Last outcome category	Context
1	8c7661d1b68d476454c5b68a58daee0d91781a94f6001555df3c7f0d6498d821	2025-01	City of London Police	City of London Police	-0.107682	51.517786	On or near B521	E01000917	Camden 027C	Other theft	Under investigation	
2	1b7fbc8deac5182e6b1e580ab4eb4ed520df688c3576bc28ea25ec8561979a1c	2025-01	City of London Police	City of London Police	-0.111596	51.518281	On or near Chancery Lane	E01000914	Camden 028B	Theft from the person	Under investigation	
3	8476f32b188fae2d0d14b7db79e872fd7688f064e8ced34368d266803c19ea75	2025-01	City of London Police	City of London Police	-0.097078	51.519045	On or near A1	E01000001	City of London 001A	Drugs	Under investigation	
4	023587ed2c674a28bd52d6e03d505c3b0dba6d25e8b95b0966f826ca5837d645	2025-01	City of London Police	City of London Police	-0.098519	51.517332	On or near Little Britain	E01000001	City of London 001A	Other theft	Investigation complete; no suspect identified	
5	92b8de6e45c3b711e802fb9d99e2a030c3f56b1c589e551e18f2070d0396c13a	2025-01	City of London Police	City of London Police	-0.09729	51.521575	On or near Fann Street	E01000001	City of London 001A	Other theft	Under investigation	

Data Sources (2/2)

The ***outcome record*** files contains fields such as:

- Crime ID
- Month (of occurrence)
- Reported by
- Falls within (jurisdiction)
- Longitude
- Latitude
- Location (descriptive)
- LSOA code
- LSAO name
- Outcome type

		Crime ID	Month	Reported by	Falls within	Longitude	Latitude	Location	LSOA code	LSOA name	Outcome type
1	8	7904a04727092e0fcf87049e45c9248676a4613402b8debaa392ea98ab2b3c10	2025-01	City of London Police	City of London Police	-0.104276	51.513712	On or near A201	E01032739	City of London 001F	Investigation complete; no suspect identified
2	12	ab5ed191a35842b8b54ee96699b40f1a3e0d788fbbec9befcf5037dbbcbb4ecf	2025-01	City of London Police	City of London Police	-0.084836	51.511939	On or near Shopping Area	E01032739	City of London 001F	Investigation complete; no suspect identified
3	15	c96523ad1238060f449f2c12e1150e213a7f7b334f666e23855f2cd163929356	2025-01	City of London Police	City of London Police	-0.100201	51.513286	On or near Dean's Court	E01032739	City of London 001F	Investigation complete; no suspect identified
4	16	39d4a451c35c36453765b345d3dd239b401c833a59b618c9d543c0cd173c0699	2025-01	City of London Police	City of London Police	-0.101292	51.51477	On or near Amen Court	E01032739	City of London 001F	Investigation complete; no suspect identified
5	17	f12f1b073c09882ee9e1ad2ab8ceb0c3f7f0694574ca6304c41f863b41ea8bbc	2025-01	City of London Police	City of London Police	-0.08491	51.512588	On or near Bell Inn Yard	E01032739	City of London 001F	Investigation complete; no suspect identified

Methodology

1. Data cleaning and Preprocessing
2. Setting constraints and Inserting data into Neo4J
3. Uploading data to Postgres

Step 1 - Data cleaning and Preprocessing

- ***Handling Missing or Null Values***
  - Ensure that only valid (non-null) values are processed.
  - Crime ID, Location, Category, Date, Latitude, and Longitude must not be NaN.
  - Outcome Type is checked and assigned only if it exists.
- ***String Cleaning and Trimming***
  - Crime ID, Location, and Category fields are stripped of leading and trailing spaces.
  - Outcome Type is also cleaned before mapping to ensure consistency.
- ***Mapping Crime Outcomes to Crime IDs***
  - A dictionary is built to map Crime IDs to their respective Outcome Types.
  - This prevents duplicate data processing and allows for efficient outcome lookups.
- ***Ensuring Complete Records Before Appending***
  - A crime record is only added to database if all key attributes are present, ensuring a clean and complete dataset.
  - Required fields before adding a record are Crime ID, Location, Crime Category, Outcome Type, Latitude & Longitude, Date
  - Any incomplete records are discarded.

	id	category	date	latitude	longitude	location	outcome
0	4ab5961fc41ed02a96ab90181b5ebfe8ac96cbce225a87...	Other theft	2022-02	51.518207	-0.106453	On or near Charterhouse Street	Investigation complete; no suspect identified
1	4f0df2df47af1714478f7a3bac24dbb88a05ec61d5ca10...	Criminal damage and arson	2022-02	51.520206	-0.097736	On or near Conference/Exhibition Centre	Investigation complete; no suspect identified
2	7e9cca9b7eb9f34e8582c6bcce2f5a86a50cfa5c522dfb...	Other theft	2022-02	51.521567	-0.097334	On or near Fann Street	Investigation complete; no suspect identified
3	45599c46a93ca27608968f41fc2e489cc688843d37972e...	Theft from the person	2022-02	51.517577	-0.098062	On or near Montague Street	Investigation complete; no suspect identified
4	3b5c8d705f1df925a5d1581eecaf5ecfad3347df7810ce...	Theft from the person	2022-02	51.515472	-0.096348	On or near Foster Lane	Investigation complete; no suspect identified
...	...	...	...	...	...	...	...
6262	c422bbdcb55ca6ed82cf815dae5f4397be78fb3c3a0860...	Theft from the person	2025-01	51.520455	-0.082648	On or near Earl Street	Investigation complete; no suspect identified
6263	7c32efcfee98552fa6f0f2ede8914b506575f2e8fcfa29...	Drugs	2025-01	51.506419	-0.088195	On or near Borough High Street	Suspect charged
6264	573205f721777c08d2ec52fd17176d52807ae48769ebf9...	Shoplifting	2025-01	51.514762	-0.062335	On or near Hessel Street	Unable to prosecute suspect
6265	a0bb4cb12396b7fb0bb922ef978e7230bed32eaa97982a...	Drugs	2025-01	51.509785	-0.068095	On or near Dock Street	Local resolution
6266	644043a1b1c21342bdd47bceada0f2ac09a9885f7c0a76...	Public order	2025-01	51.508588	-0.074039	On or near Tower Bridge Approach	Investigation complete; no suspect identified

Dataset post cleaning

Step 2: Setting constraints and Inserting data into Neo4J

- ***Constraint*** are set to maintain data integrity, prevent duplication, and improve query performance in Neo4j.
  - Unique Places by Latitude and Longitude
  - Unique Crime Categories
  - Unique Year Entries
  - Unique Month Entries
  - Unique Crime Outcomes
  - Unique Crime Cases
- The crime data is ***structured*** into a graph format for relationship-based analysis.
  - Column names are standardized
  - Year and month is extracted from dates
  - Crime records are mapped to places, categories, outcomes, and time entities while ensuring uniqueness
- The data is processed in batches of 1000 records for efficiency, with error handling to skip problematic entries.
- ***Relationships*** are created such as
  - Crimes occur at places
  - Crimes are linked to years and months
  - Crimes are classified by category
  - Crimes have outcomes

Step 3: Uploading data to Postgres

- **Connect to PostgreSQL**
  - Psycopg2 was used to connect to the database creating a cursor to execute queries.

- **Create a '*crime_data*' table**
  - A table "crime_data" is created with the columns matching our preprocessed data:
    - id (Primary Key, auto-incremented)
    - location (Text, required)
    - date (Date, required)
    - category (Text, required)
    - outcome (Text, required)

- **Insert Data into PostgreSQL**
  - SQLAlchemy was used to insert data into the ***crime_data*** table.

Use case queries – Using Neo4J

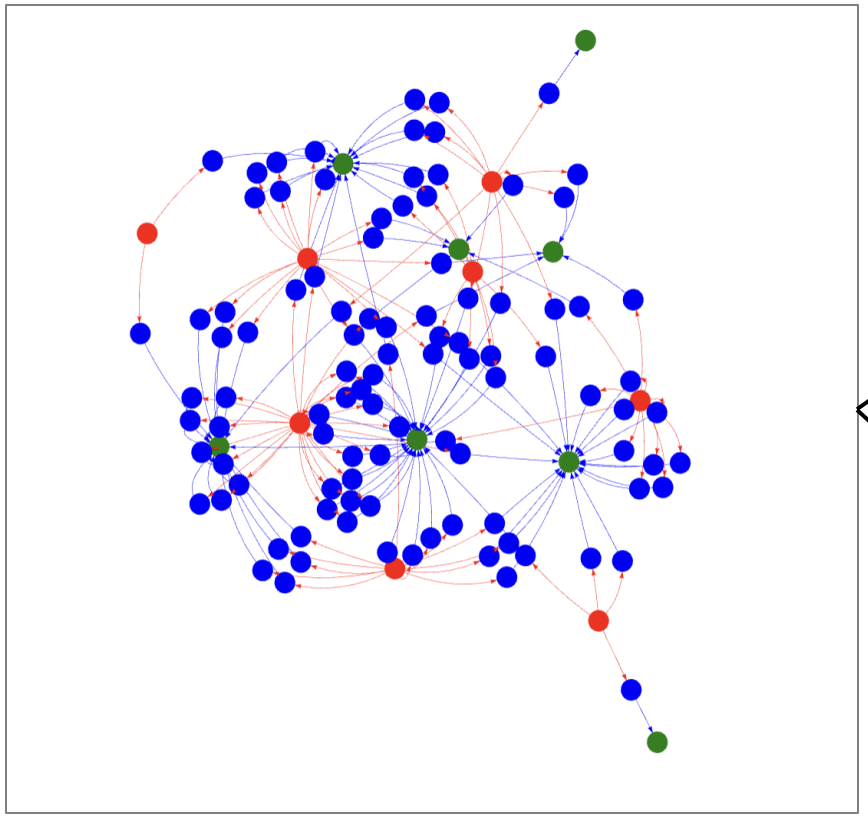
Query 1: Top 10 crime hotspots based on frequency

- The query identifies crime hotspots by selecting places where crimes have occurred in more than one distinct month and have more than five total crimes.

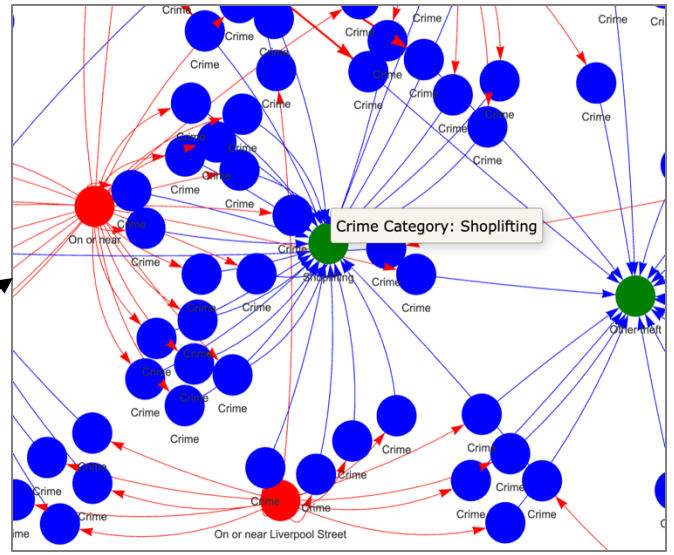
- It retrieves the top 10 such places and fetches details of the latest 100 crimes, including their category, month, and year.

- The MATCH clause in Cypher directly models real-world relationships like "Crime occurred at Place" or "Crime belongs to a Category". PostgreSQL would require complex joins and indexing strategies to achieve similar efficiency.

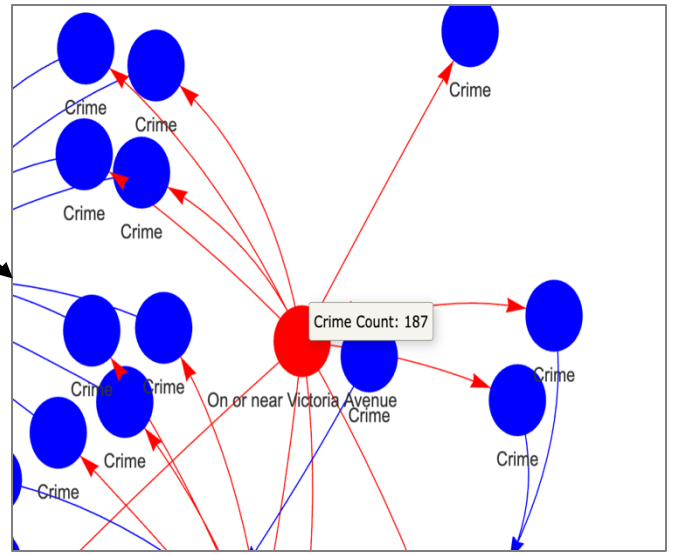
- In the output the nodes and edges represent the following:
  - ***Nodes***:
    - Red Nodes → Places (Crime Hotspots)
    - Blue Nodes → Crimes
    - Green Nodes → Crime Categories (e.g., Theft, Assault, Burglary). Edges:
  - ***Edges***
    - From Red (Place) → Blue (Crime)
    - The edge is labeled as "Occurred At" and indicates that a crime happened at the specific place.
    - From Blue (Crime) → Green (Crime Category)
    - The edge is labeled as "Crime Type and connects a crime to its respective category, showing what type of crime was committed.



Output Graph



Crime ID -> *Crime Category*



Place -> *Crime ID*

Query 2: Using PageRank algorithm to rank places by their importance based on crime connections

- The query projects a graph in Neo4j where places are connected by crimes using the "OCCURRED_AT" relationship.
- The PageRank algorithm is used to rank the top 100 most influential crime hotspots based on their connectivity.
- Neo4j outperforms PostgreSQL which would require recursive CTEs and multiple joins for this task. It efficiently traverses relationships without expensive joins or recursive queries. This makes it faster and more scalable for analyzing crime centrality and hotspot influence.
- The query outputs a ranked list of **top 100 crime hotspots**, showing each **location's name** and its **PageRank score**, which indicates its influence based on crime connectivity. Higher scores represent places with **frequent and highly connected crimes**, making them key areas of interest for crime analysis.

	location	score
0	On or near Montague Street	0.15
1	On or near Foster Lane	0.15
2	On or near Beech Street	0.15
3	On or near Conference/Exhibition Centre	0.15
4	On or near Park/Open Space	0.15
...	...	...
95	On or near America Square	0.15
96	On or near Conference/Exhibition Centre	0.15
97	On or near Bus/Coach Station	0.15
98	On or near White Kennett Street	0.15
99	On or near Fann Street	0.15

Query 3: Using Louvain algorithm to identify clusters of interconnected crimes, ranking the top 5 largest crime networks

- The query applies the Louvain algorithm to detect crime communities by grouping closely connected crimes into clusters.
- It returns the top 5 largest crime communities, listing their community ID and the associated crime case IDs.
- This helps in identifying patterns of criminal activity, such as recurring crime networks or hotspots with frequent offenses.

	communityId	crime_cases
0	386	[5fe737fc8c3a5c12b4dd9af09476c644d3f8ed764149f...
1	771	[31b462c2f9f2d6855c28e5c175099e39b309ddd5a40fe...
2	1486	[48dd14dc102a4592f7591888762839ede6e513bff602c...
3	1349	[a86f33f326d3d9a1f8c3972520e117177660290 31efb9...
4	1669	[f352a40ae9df837bf0817f0b50cf7db7f9f797388f0b8...

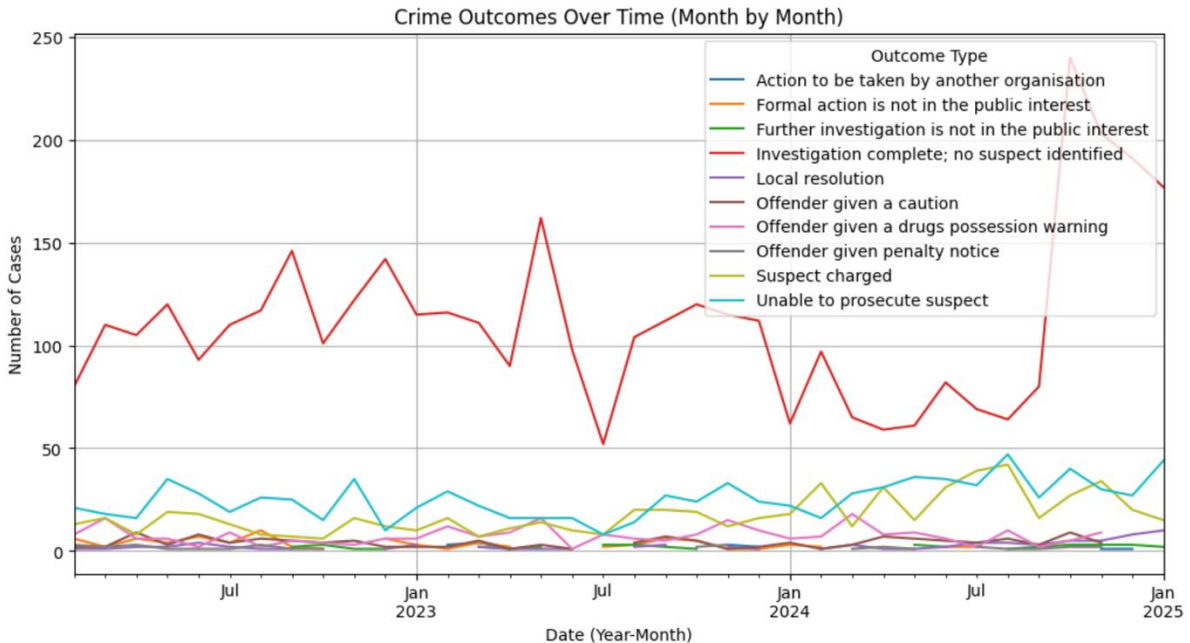
Query 4: Top 50 Co-Occurring Crime Categories: Patterns & Associations

- The query identifies crime categories that frequently co-occur at the same places by finding pairs of crimes linked to the same location. It ranks the top 50 category pairs based on the number of places where both crimes have occurred together.
- This helps in understanding crime patterns, such as which crime types tend to happen together, aiding in crime prevention strategies.

- In the output the nodes and edges represent the following:
  - ***Nodes*** → Crime Categories
    - Each node represents a crime category (e.g., Theft, Assault, Burglary).
  - ***Edges*** → Connections Between Categories)
    - An edge between two categories means they co-occurred at the same place.
  - ***Edge Thickness***
    - The thicker the edge, the higher the number of places where the two crime categories co-occurred, highlighting stronger crime associations.

Query 5: Crime case outcomes over time

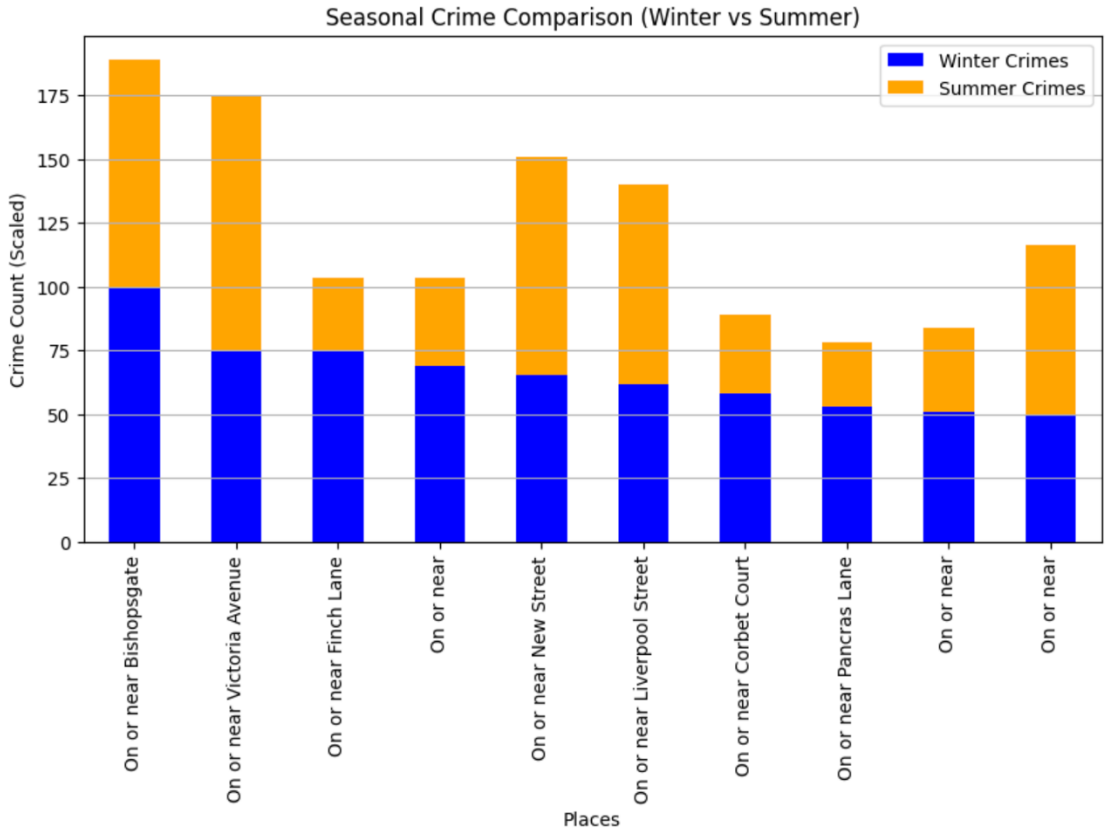
- The query retrieves crime outcomes over time, counting the number of cases that resulted in each outcome type per month and year.
- It organizes the data chronologically and visualizes trends using a line graph, where each line represents a different outcome type.
- This helps in understanding how crime resolutions change over time, such as fluctuations in convictions, dismissals, or ongoing investigations.
- The month-by-month breakdown provides insights for law enforcement and policymakers to track justice system efficiency.



	Year	Month	OutcomeType	cases	Date
0	2022	02	Formal action is not in the public interest	6	2022-02-01
1	2022	02	Further investigation is not in the public int...	1	2022-02-01
2	2022	02	Investigation complete; no suspect identified	80	2022-02-01
3	2022	02	Local resolution	1	2022-02-01
4	2022	02	Offender given a caution	2	2022-02-01
...	...	...	...	...	...
284	2025	01	Investigation complete; no suspect identified	177	2025-01-01
285	2025	01	Local resolution	10	2025-01-01
286	2025	01	Offender given a caution	1	2025-01-01
287	2025	01	Suspect charged	15	2025-01-01
288	2025	01	Unable to prosecute suspect	44	2025-01-01

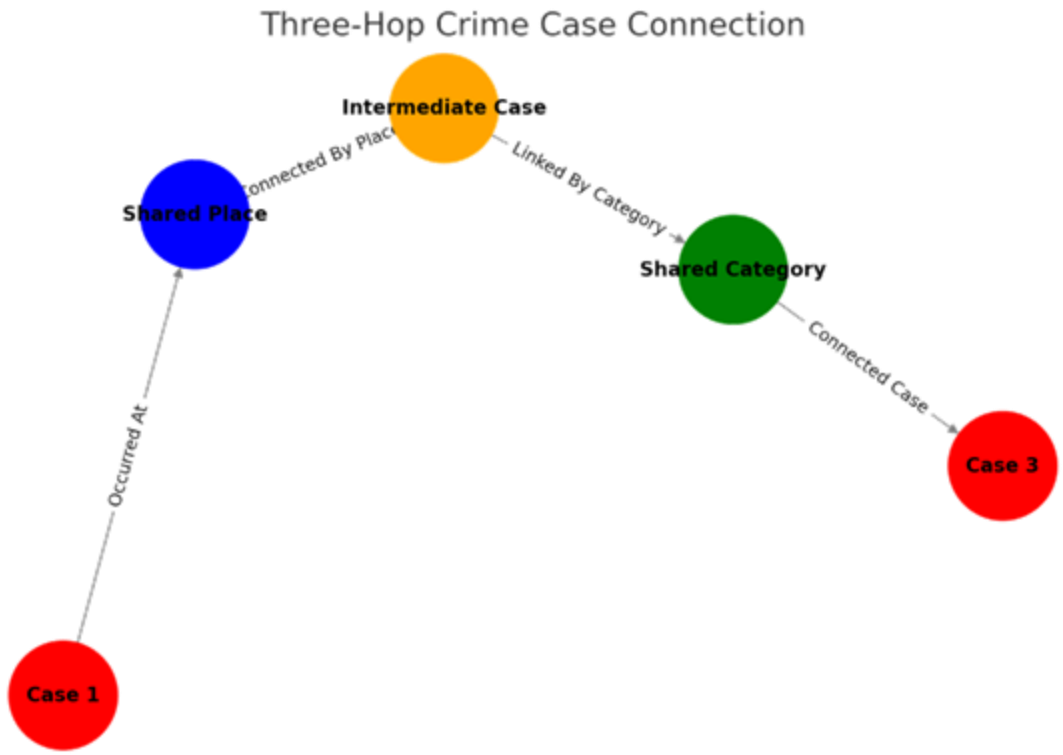
Query 6: Seasonal crime trends

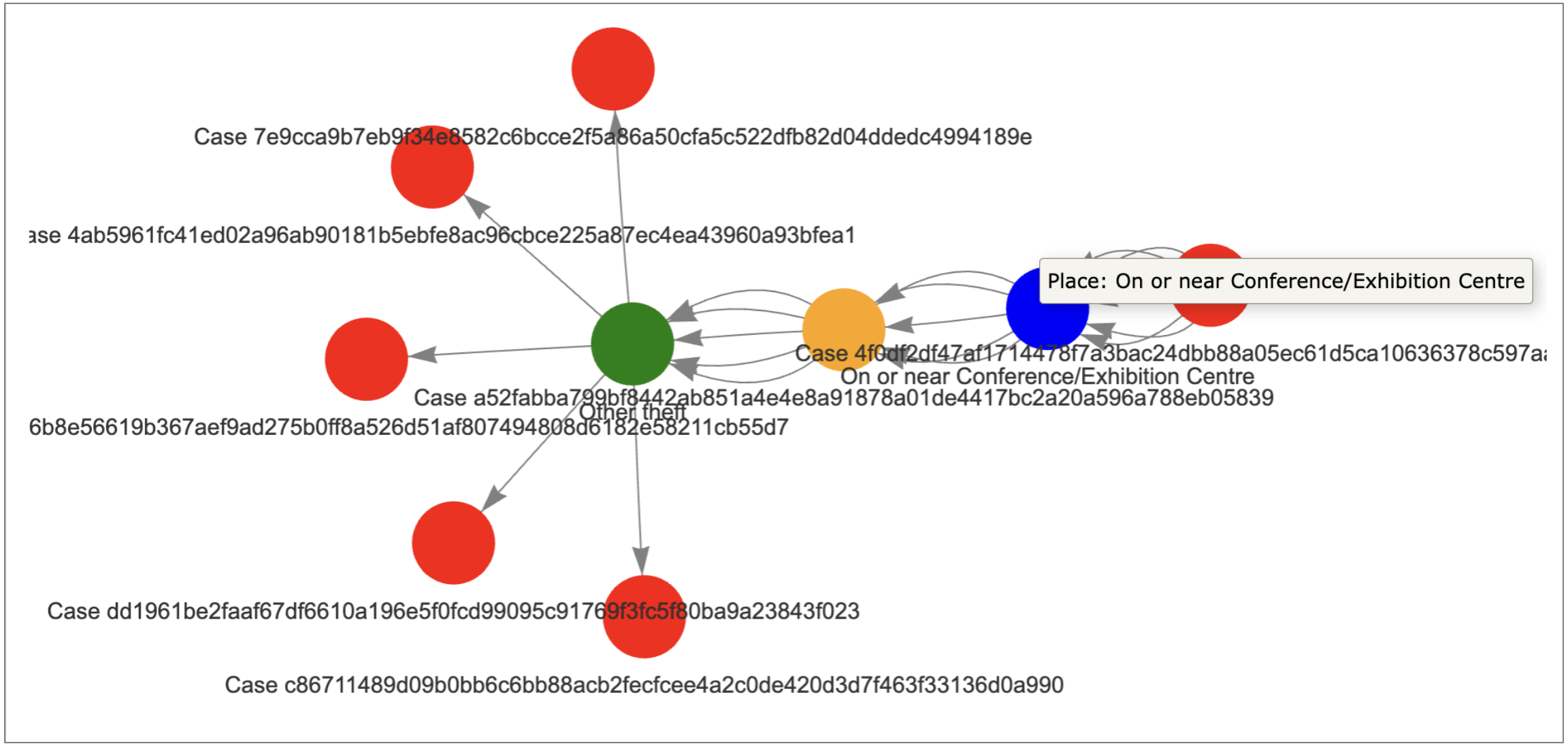
- The query analyzes seasonal crime trends by counting the number of crimes that occurred during winter (Dec-Feb) and summer (Jun-Aug) for different places.
- It identifies the top 10 locations with significant seasonal crime variations and compares their winter and summer crime counts.
- The results are visualized using a stacked bar chart, highlighting places where crime rates fluctuate between seasons. This helps in understanding seasonal crime patterns, which can assist law enforcement in resource allocation and crime prevention strategies.
- Blue bars represent winter crimes, while orange bars represent summer crimes, showing seasonal crime variations.



Query 7: Identifying interconnected crime cases

- The query identifies ***multi-hop relationships*** between crime cases, linking them through shared locations and categories.
- It finds cases that occurred at the same place, categorizes them by their crime type, and establishes a three-hop connection between cases based on intermediate relationships.
- The results are visualized in a network graph, where cases, shared places, and crime categories are interconnected to reveal patterns of linked criminal activities.
- In the output the nodes and edges represent the following:
  - ***Nodes***:
    - Red → Crime ID
    - Orange → Intermediate case.
    - Blue → Shared place.
    - Green → Shared crime category.
  - ***Edges***:
    - "Occurred At" → Connects a crime case to its location.
    - "Connected By Place" → Links cases occurring at the same place.
    - "Linked By Category" → Shows category-based crime connections.
    - "Connected Case" → Final connection between cases in the multi-hop relationship.





Output Graph

Use case queries – Using PostgreSQL

Query 1: Crime Categories with the Highest Resolution Rate

- The query retrieves crime data, grouping cases by category and calculating the total number of cases, the number of solved cases, and the resolution rate (percentage of solved cases).
- It excludes unresolved cases from the solved count and sorts the results by resolution rate in descending order.

Crime Categories with the Highest Resolution Rate:

Crime Category	Total Cases	Solved Cases	Resolution Rate
Drugs	520	517	99.42%
Possession of weapons	72	64	88.89%
Other crime	44	36	81.82%
Violence and sexual offences	805	638	79.25%
Public order	363	265	73.00%
Shoplifting	826	494	59.81%
Criminal damage and arson	217	63	29.03%
Robbery	44	11	25.00%
Burglary	89	22	24.72%
Other theft	1941	101	5.20%
Bicycle theft	162	8	4.94%
Vehicle crime	123	5	4.07%
Theft from the person	1061	29	2.73%

Query 2: Predicting Crime Trends Using Rolling Averages

- The query calculates the total number of crimes per month and computes a 6-month moving average of crime trends over time.
- It first aggregates crime counts by month using a common table expression (CTE) and then applies a moving average window function to smooth fluctuations.

Crime Trends Over Time:

Crime Month	Total Crimes	6-Month Moving Avg
2022-02-01 00:00:00-08:00	136	100.00%
2022-03-01 00:00:00-08:00	168	90.48%
2022-04-01 00:00:00-07:00	155	98.71%
2022-05-01 00:00:00-07:00	190	85.39%
2022-06-01 00:00:00-07:00	164	99.15%
2022-07-01 00:00:00-07:00	163	99.80%
2022-08-01 00:00:00-07:00	173	97.59%
2022-09-01 00:00:00-07:00	196	88.52%
2022-10-01 00:00:00-07:00	136	125.25%
2022-11-01 00:00:00-07:00	182	92.86%
2022-12-01 00:00:00-08:00	182	94.51%
2023-01-01 00:00:00-08:00	161	106.63%
2023-02-01 00:00:00-08:00	179	96.46%
2023-03-01 00:00:00-08:00	165	101.52%
2023-04-01 00:00:00-07:00	136	123.16%

Query 3: Identifying High Crime Time Series Anomalies

- The query analyzes monthly crime data to detect anomalies using Z-scores.
- It first calculates the total crimes per month, then computes the average and standard deviation of crime counts.
- Each month's crime count is standardized using the Z-score formula, and months with an absolute Z-score greater than 2 (indicating significant crime spikes or drops) are identified and sorted in descending order of severity.

Crime Anomalies (Months with Unusual Crime Spikes or Drops):

Crime Month	Total Crimes	Z-Score
2024-10-01 00:00:00-07:00	332	3.3066879053919957
2024-11-01 00:00:00-07:00	291	2.4481705178179261

Query 4: Crime Evolution Over the Years

- The query analyzes yearly crime trends by category, calculating the total number of crimes per year.
- It then computes the difference in crime counts from the previous year using a window function.
- The results are sorted in descending order based on the yearly difference, highlighting the top 25 categories with the most significant increases in crime incidents.

Crime Year	Category	Total Crimes	Yearly Difference
2022	Theft from the person	338	NaN
2022	Criminal damage and arson	77	NaN
2022	Bicycle theft	41	NaN
2022	Other crime	5	NaN
2022	Public order	91	NaN
2022	Shoplifting	183	NaN
2022	Burglary	25	NaN
2022	Vehicle crime	61	NaN
2022	Robbery	16	NaN
2022	Violence and sexual offences	252	NaN
2022	Other theft	589	NaN
2022	Drugs	152	NaN
2022	Possession of weapons	15	NaN
2024	Shoplifting	409	222.0
2023	Other theft	745	156.0
2024	Theft from the person	385	102.0
2024	Violence and sexual offences	299	77.0
2023	Public order	119	28.0
2023	Drugs	175	23.0
2024	Public order	137	18.0
2023	Other crime	23	18.0
2024	Burglary	39	16.0
2023	Bicycle theft	56	15.0
2024	Criminal damage and arson	73	12.0
2023	Possession of weapons	26	11.0

Use case queries – Using PostgreSQL and Neo4J in combination

- The query aims to identify the month with the highest crime rate and analyze the most common crime categories during that period.
- It first queries a PostgreSQL database to determine the peak crime month by counting incidents per month.
- Then, it retrieves the top 5 crime categories for that month using a Neo4j graph database, grouping cases by category.
- The results are displayed, and a pie chart visualization is created to show the distribution of crime types, helping to understand crime trends during the peak period.

```
Peak Crime Month: 10/2024

Top Crime Types in Peak Month:
Theft from the person: 89 incidents
Other theft: 72 incidents
Shoplifting: 66 incidents
Violence and sexual offences: 37 incidents
Public order: 19 incidents
```