МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

КУРСОВИЙ ПРОЕКТ

з дисципліни “Бази даних”

спеціальність 121 – Програмна інженерія

на тему:

Моніторингова система збройних нападів

Студентка

групи КП-73 Берещенко Анастасія Сергіївна _____

(підпис)

Викладач

к.т.н, доцент кафедри

СПіСКС **Петрашенко А.В.** _____

(підпис)

Захищено з оцінкою _____

Київ – 2020

Анотація

Метою розробки даного курсового проекту є набуття практичних навичок розробки сучасного програмного забезпечення, що взаємодіє з постреляційними базами даних, а також здобуття навичок оформлення відповідного текстового, програмного та ілюстративного матеріалу у формі проектної документації. У результаті виконання курсового проекту було опановано навик розробляти програмне забезпечення для постреляційних баз даних, володіння основами використання СУБД, а також інструментальними засобами аналізу великих обсягів даних.

Темою даного курсового проекту є створення ПЗ для моніторингової системи для виявлення закономірності у характеристиках збройних нападів в США. У документі викладена актуальність та проблематика аналізу великого обсягу даних, аналіз використаного інструментарію (опис мови програмування, використаних бібліотек та СКБД), описана структура бази даних, опис розробленого програмного забезпечення (загальний, опис модулів та основних алгоритмів роботи), аналіз функціонування засобів масштабування, та опис результатів проведеного аналізу.

Результатами даного проекту стали діаграми та графіки, що зображають статистику кількості інцидентів в усіх штатах за весь час, кількості інцидентів по рокам у кожному штаті, кількості поранених і убитих по рокам, гендерний та віковий розподіл між жертвами на підозрюваними. З даними діаграмами та графіками можна ознайомитися в додатку А.

Зміст

Анотація **1**

Зміст **2**

Вступ **3**

Аналіз інструментарію для виконання курсового проект **5**

Аналіз СКБД 5

Обґрунтування вибору мови програмування 9

Обґрунтування вибору бібліотек і фреймворків 10

Структура бази даних **12**

Опис результатів аналізу предметної галузі **21**

Висновки **23**

Література **25**

Додаток А **27**

Вступ

Станом та поточний рік все більше і більше компаній приходять до того, що дані, якими вони оперують та збирають в процесі своєї роботи потребують агрегації та аналізу. Використання класичних методів аналізу, що базуються на звичайних алгоритмах математичної статистики та ймовірності не проносять очікуваної користі. З плином часу все більшу і більшу актуальність набирають програмні засоби аналізу даних, які інколи базуються на алгоритмах машинного навчання, що дозволяє зробити аналіз даних інтелектульним (*en. data minning*) Цю науку, яка поєднує в собі програмування, мат. статистику, теорію ймовірності та машине навчання називаються наукою про дані (*en. Data Science*).

Сучасний Data Science спеціаліст (дослідник даних) має оперувати великими обсягами даних (від одного гігабайта до декількох терабайт) та вміти виокремити з них приховані залежності, на основі яких зробити прогноз про те, як будуть надалі відбуватися ті чи інші явища. Дослідники даних використовують свої дані та аналітичні здібності для пошуку та інтерпретації великих джерел даних; керують великими обсягами даних безвідносно до апаратного та програмного забезпечення і обмежень пропускної здатності; об'єднують джерела даних; забезпечують цілісність наборів даних; створюють візуалізації для кращого розуміння даних; з використанням даних будують математичні моделі; надають тлумачення даних та висновки.

Класичним набором інструментів, яким користується Data Science спеціаліст є мова програмування Python 3 і набір математичних бібліотек до неї (pandas, scikit-learn, numpy, mathplotlib), нереляційні бази даних (на кшталт MongoDB, DynamoDB, Cassandra) та закони математичного аналізу, теорії ймовірності та мат. статистики).

Метою створення даного проекту був аналіз даних про закономірності в збройних нападах в США, з метою дослідження з метою встановлення соціальних залежностей. Критерієм актуальності було обрано той факт, що в США зараз проходять протести #BlackLivesMatter, тому цілком доцільно чи вплине характер поведінки простестуючих на загальну картину озброєних нападів.

Дані про озброєні напади в США було запозичено з відкритого реєстру даних для machine learning ентузіастів https://www.kaggle.com/jameslko/gun-violence-data.

Аналіз інструментарію для виконання курсового проект

Аналіз СКБД

В процесі виконання цього курсового проекту перед нами встала потреба кешувати та зберігати дані про збройні напади між запусками аналізатора. Кожного разу читати дані із CSV-файлів є дуже дорогою операцією, тож було прийняте рішення використати СКБД. В якості СКБД були розглянуті варіанти: PostgreSQL, MongoDB, CassandraDB. З порівняльною характеристикою цих СКБД можна ознайомитися в таблиці 1.

таблиця 1. Порівняльна характеристика СКБД

Критерій порівняння	Назва СКБД		
	MongoDB	PostgreSQL	CassandraDB
Має відкритий вихідний код	так	так	так
Схема даних	динамічна	статична і динамічна	статична і динамічна
Підтримка ієрархічних даних	так	так (з 2012)	ні
Реляційні дані	ні	так	так
Транзакції	ні	так	так
Атомарністі операцій	всередині документа	по всій БД	всередині партіції

Мова запитів	JSON	SQL	CQL
Найлегший спосіб масштабування	горизонтальний	вертикальний	горизонтальний
Підтримка шардингів	так	так (важка конфігурація)	так (може зберігати партіції на різних машинах)
Приклад використання	Великі дані (мільярди записів) з великою кількістю паралельних оновлень, де цілісність і узгодженість даних не потрібно.	Транзакційні і операційні програми, вигода яких в нормалізованому формі, об'єднаннях, обмеження даних і підтримки транзакцій.	Багато запитів на запис/читання у одиницю часу, до даних можна задіяти партіціюваннння за ключем, дані мають лише первинні індекси
Наявність бібліотек для мови програмування Python 3	так	так	так
Підтримка реплікації	так, автоматичне	За принципом master-slave	так, через партіціювання

	переобрання головного процесу		
Засіб збреження та відновлення даних	mongodump	pg_dump	не має окремого доданка, виконується засобами CQL
Форма збереження даних	документи BSON	таблиця	таблиця

За результатами порівняння цих СКБД було прийнято рішення зупинитися на NoSQL рішеннях. Оскільки вони чудово поєднують в собі переваги неструктурованих баз даних та простоту використання горизонтального масштабування. Крім цього класичним прикладом використання NoSQL СКБД є системи збору та аналізу даних, до яких можна застосувати індексування за первинними та вторинними ключами.

Ця база даних є об’єктно орієнтованою та дозволяє зберігати великі масиви неструктурованих даних. На відміну від SQL баз даних ми можемо зберігати дані у “сирому” об’єктному вигляді, який використовується програмою та є більш близьким за структурою до моделі даних, яку буде використовувати ПЗ написане з використанням мови програмування Python. Це пришвидшить збір, збереження та отримання даних програмним забезпеченням. Оскільки MongoDB є представником NoSQL баз даних, вона не потребує жорсткої схеми даних, що дозволяє пришвидшити процес розробки та зробити його більш гнучким. Окрім цього дана СУБД підтримує горизонтальне масштабування за допомогою шардингу з метою зменшення

навантаження на кожен окремий вузол шляхом розподілення навантаження між ними всіма.

Нижче наведено перелік основних переваг:

- Підтримка ієрархічних даних
- Динамічна схема
- Швидкість запису у колекцію
- Швидкість читання із колекції
- Простота масштабування та відновлення даних

Обґрунтування вибору мови програмування

Мовою програмування для ПЗ було обрано Python 3.8. Оскільки це строго динамічно типізована мова програмування. Це дозволяє пришвидшити швидкість розробки ПЗ за рахунок уникнення постійного оголошення типів даних, якими оперує програма, та водночас не дає зробити критичних помилок, прихованих за автоматичним приведенням типів, на відміну від мови JavaScript.

Ключовою особливістю Python є широкий інструментарій засобів розробки систем збору та аналізу даних. Де-факто ця мова є стандартом у світі математичних розрахунків та обробки даних у реальному часі. Тож під час виконання курсового проекту було відносно легко знайти необхідну документацію та приклади роботи із цією мовою.

Обґрунтування вибору бібліотек і фреймворків

Використані бібліотеки:

- pandas — бібліотека для обробки та аналізу даних. Pandas є open-source бібліотекою, ліцензована BSD (Berkeley Source Distribution), забезпечує швидкі, гнучкі та експресивні структури даних, призначені для того, щоб працювати з «реляційними» або «помітними» даними як простими, так і інтуїтивно зрозумілими. Він прагне бути основним будівельним блоком високого рівня для здійснення практичного, реального аналізу даних у Python. Можливості бібліотеки: інструменти для обміну даними між структурами в пам'яті і файлами різних форматів, засоби поєднання даних і способи обробки відсутньої інформації, Переформатування наборів даних, в тому числі створення зведених таблиць, зріз даних за значеннями індексу, розширені можливості індексування, вибірка з великих наборів даних, вставка і видалення стовпців даних, можливості угруповання дозволяють виконувати трьохетапні операції типу «поділ, зміна, об'єднання», злиття і об'єднання наборів даних;
- matplotlib — це бібліотека Python 2D, яка представляє числові дані у різноманітних форматах та інтерактивних середовищах на різних платформах. Також ця бібліотека - це математичне розширення NumPy. Він надає об'єктно-орієнтований API для вбудови ділянок у додатки, що використовують набір інструментів для загального графічного інтерфейсу, таких як Tkinter, wxPython, Qt або GTK+. Matplotlib може використовуватися в скриптах Python, оболонках Python та IPython, серверах веб-додатків та чотирьох графічних наборів інструментів для користувацького інтерфейсу. Бібліотека sciPy використовує matplotlib.
- numpy — математична бібліотека мови Python, що додає підтримку великих багатовимірних масивів і матриць, разом з великою

бібліотекою високорівневих математичних функцій для операцій з цими масивами. А також інструменти для інтеграції C/C++ і Fortran коду, корисну лінійну алгебру, перетворення Фур'є і можливості випадкових чисел. Крім очевидних наукових застосувань, NumPy також може бути використаний як ефективний багатовимірний контейнер загальних даних. Можна визначити довільні типи даних. Це дає змогу без проблем і швидкої інтеграції NumPy з різноманітними базами даних. NumPy має ліцензію за ліцензією BSD, що дозволяє повторно використовувати кілька обмежень.

Структура бази даних

База даних складається з однієї колекції gun_cases, яка зберігає інформацію про збройні напади в США за 2013-2018 рр. Стурктура колекції наведена в таблиці 2.

таблиця 2. Опис властивостей документа у колекції gun_cases

Назва властивості	Тип	Опис
_id	ObjectId	Ідентифікатор запису
date	Date	Дата інциденту
state	Str	Назва штату
city	Str	Назва міста
n_killed	Int	Кількість загиблих
n_injured	Int	Кількість поранених
victim_age	Str	Перелік через кому вікових груп жертв
victim_gender	Str	Перелік через кому гендерів жертв
suspect_age	Str	Перелік через кому вікових груп підозрюваних
suspect_gender	Str	Перелік через кому гендерів підозрюваних

Опис результатів аналізу предметної галузі

В результаті виконання курсового проекту було проаналізовано закономірності в збройних нападах в США.

1. Згідно з рисунком А1

   Найнебезпечнішими штатами є Техас, Флорида та Каліфорнія. Найспокійнішими штатами є Гавайї, Південна Дакота та Вайомінг.

2. Згідно із рисунком А2

   Припустивши, що дані за крайні роки (2013 та 2018) можуть бути неповними, кількість інцидентів залишається незмінною з року в рік.

3. Згідно із рисунком А3

   В штаті Огайо щорічно збільшується кількість загиблих та поранених.

4. Згідно із рисунком А4

   Чоловіків виявилося більше і серед жертв, і серед підозрюваних. Подібну картину можна пояснити тим, що участь в збройних нападах здебільшого беруть участь чоловіки.

5. Згідно із рисунком А5

   Участь у збройних нападах переважною більшістю беруть повнолітні особи.

Висновки

В процесі виконання даного курсового проекту було отримано практичні навички обробки великих масивів даних за допомогою мови програмування Python 3 та СКБД MongoDB.

Було проаналізовано сучасні методи та інструменти для роботи із великими даними, знайдено гарно працюючу комбінацію із мови програмування Python 3 та бібліотек до нього: Pandas, , Mathplotlib, Numpy. Було складено порівняльну таблицю із трьох баз даних, які найчастіше використовують для роботи із часовими рядами. На основі цих даних було обрано в якості СКБД NoSQL рішення MongoDB.

На основі зібраних даних було проаналізовано статистику збройних нападів в США. Були знайдені закономірності кількості інцидентів в усіх штатах за весь час, кількості інцидентів по рокам у кожному штаті, кількості поранених і убитих по рокам, гендерний та віковий розподіл між жертвами на підозрюваними. Дані аналізу приведені у Додатку А. Текстовий опис надано в відповідному розділу цього курсового проекту.

В ході виконання даного курсового проекту було досягнуто поставленої мети: було набуто практичних навичок розробки сучасного програмного забезпечення, що взаємодіє з NoSQL базами даних, а також були здобуті навички оформлення відповідного текстового, програмного та ілюстративного матеріалу у формі проектної документації. У результаті виконання курсового проекту я навчилася писати програмне забезпечення для NoSQL баз даних, володіти основами використання СУБД, а також інструментальними засобами аналізу великих обсягів даних, а саме відкритими бібліотеками мови Python.

Література

1. Hashed sharding [Електронний ресурс]. Режим доступу до ресурсу: https://docs.mongodb.com/manual/core/hashed-sharding/
2. Aggregation [Електронний ресурс]. Режим доступу до ресурсу: https://docs.mongodb.com/manual/aggregation/
3. 5 причин, почему машинное обучение станет технологией 2018 [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: https://blogs.oracle.com/russia/machine-learning-2018;
4. Сложности накопления данных для интеллектуального анализа [Електронний ресурс]. – 2012. – Режим доступу до ресурсу: https://habr.com/post/154787/;
5. Почему Python так хорош в научных вычислениях [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: https://habr.com/post/349482/;
6. Python [Електронный ресурс]. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Python;
7. PostgreSQL лучше чем MongoDB[Електронний ресурс]. – 2015. – Режим доступу до ресурсу: https://habr.com/post/272735/;
8. 27 шпаргалок п машинному обучению [Електронний ресурс]. – 2017. – Режим доступу до ресурсу: https://proglib.io/p/ds-cheatsheets/;
9. Pandas [Електронний ресурс]. – Режим доступу до ресурсу: https://ru.wikipedia.org/wiki/Pandas;
10. Scikit-learn [Електронний ресурс]. – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Scikit-learn;
11. Matplotlib [Електронний ресурс]. – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Matplotlib;
12. NumPy [Електронний ресурс]. – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/NumPy;

Додаток А

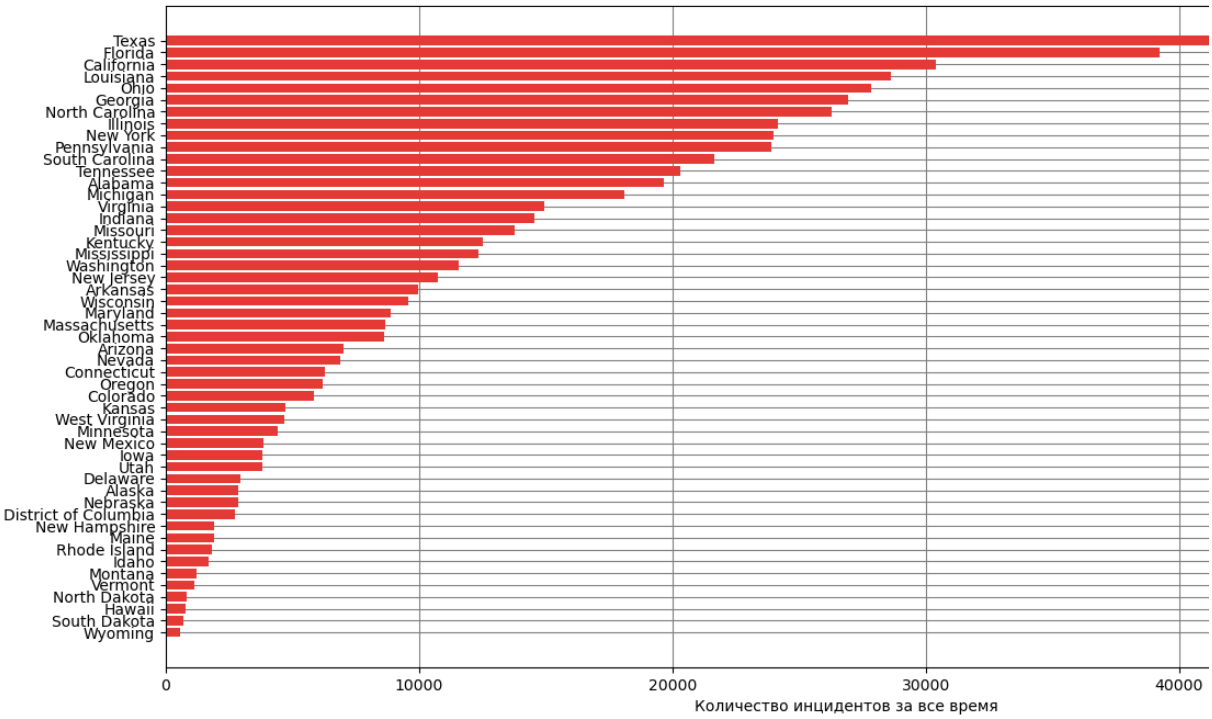

Рис 1. Розподіл інцидентів між штатами

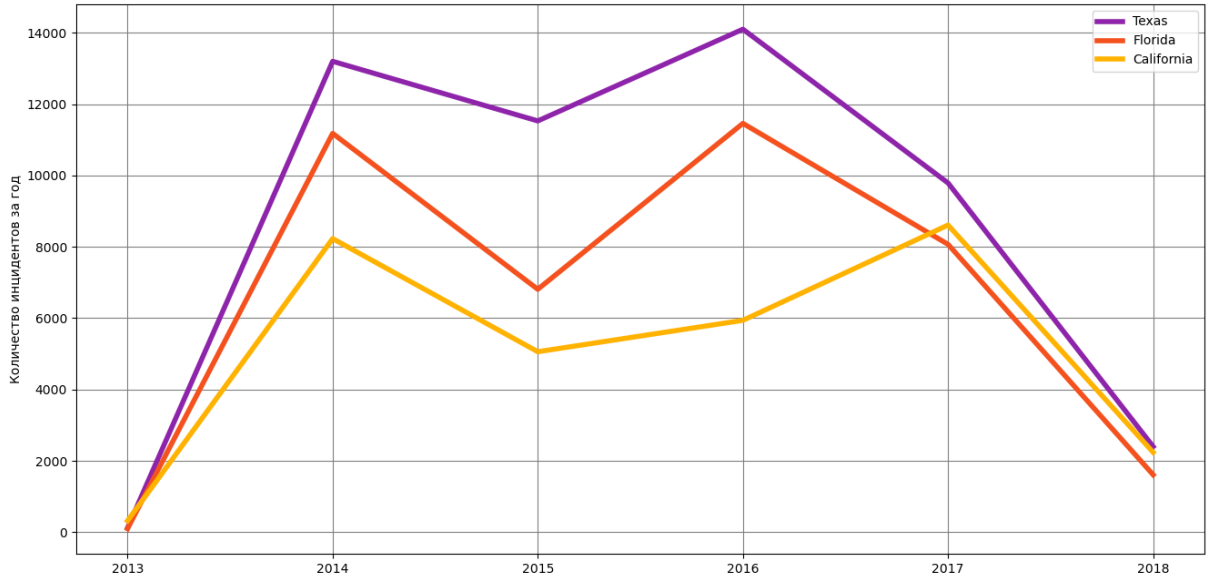

Рис 2. Розподіл кількості інцидентів по роках

(штати Техас, Флорида, Каліфорнія)

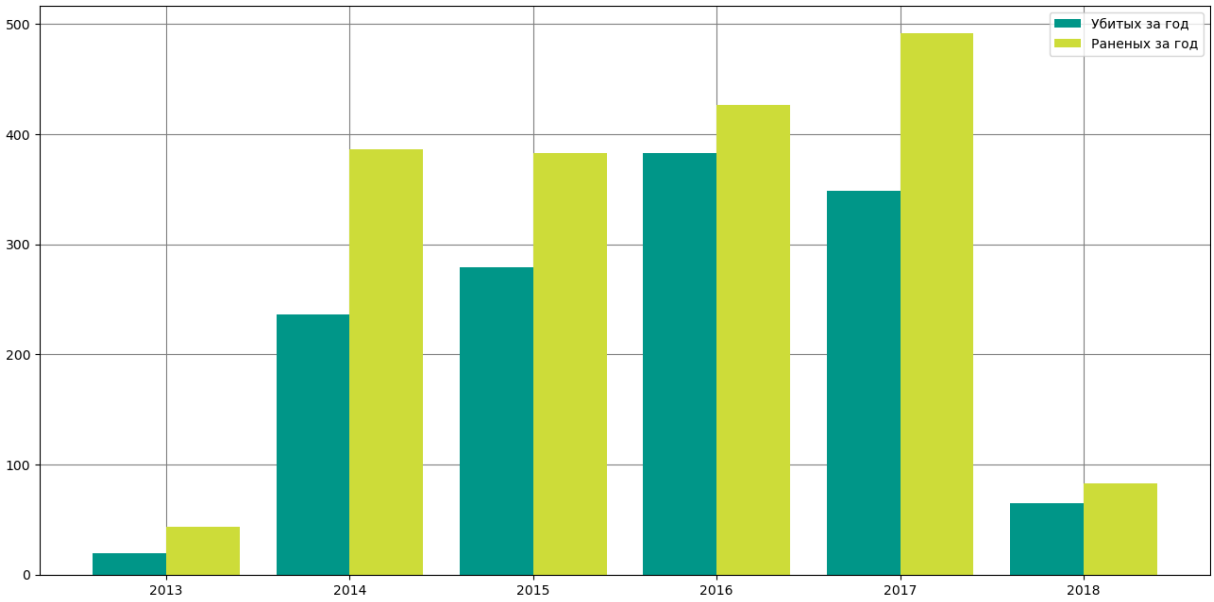

Рис 3. Розподіл загиблих та поранених по рокам в штаті Огайо

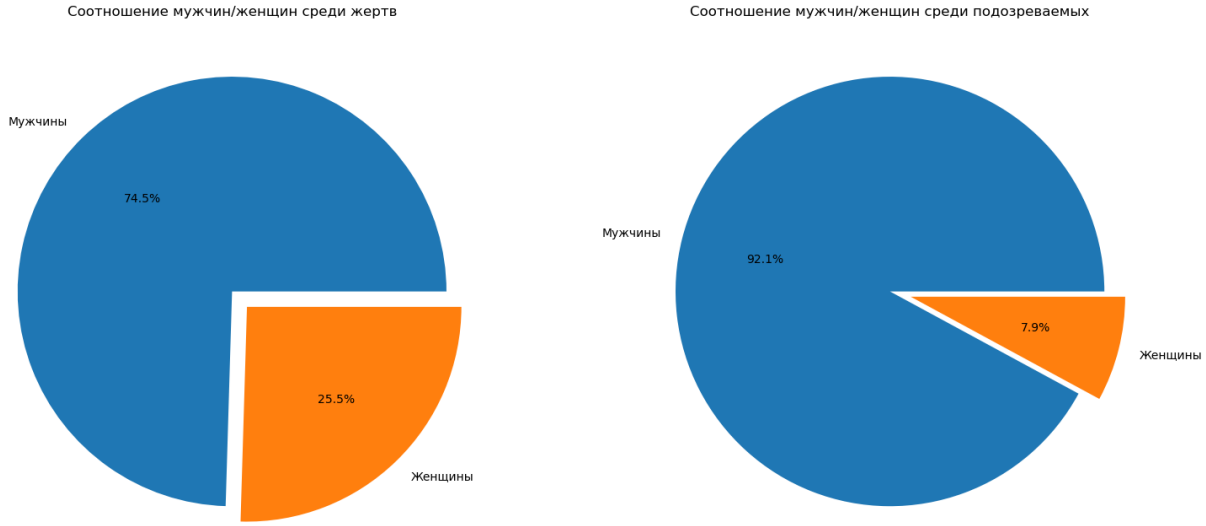

Рис 4. Гендерний розподіл між жертвами та підозрюваними

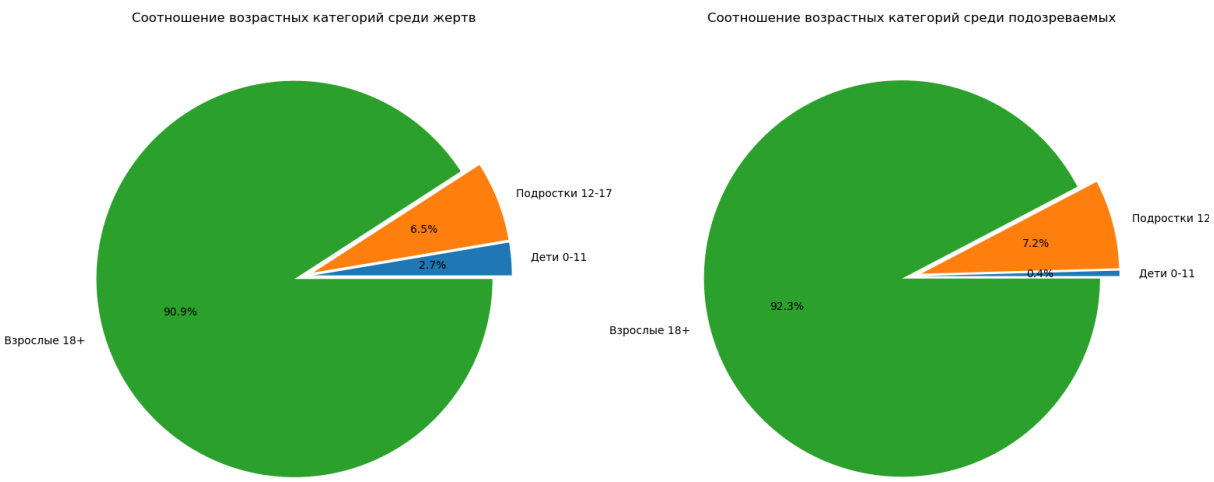

Рис 5. Віковий розподіл між жертвами та підозрюваними