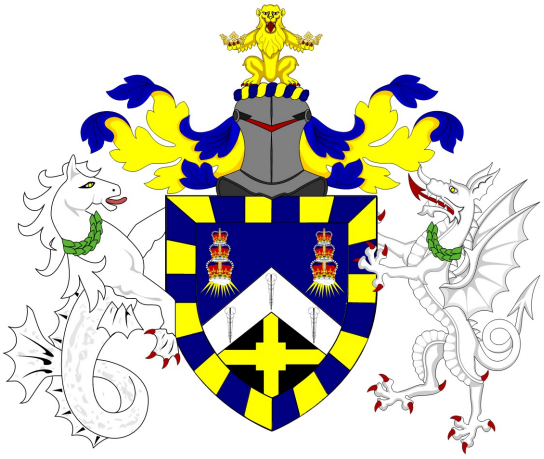
Data Analytics MSc Dissertation MTHM038, 2023/24

Differential Privacy

Sarthak Sulhan, 230767490

Supervisor: Prof. Ian Morris



A thesis presented for the degree of

Master of Science in Data Analytics

School of Mathematical Sciences

and *possible joint school*

Queen Mary University of London

Declaration of Original Work

This declaration is made on August 21, 2024.

Student's Declaration: I, Sarthak Sulhan, hereby declare that the work in this thesis is my original work. I have not copied from any other students' work, work of mine submitted elsewhere, or from any other sources except where due reference or acknowledgment is made explicitly in the text. Furthermore, no part of this dissertation has been written for me by another person, by generative artificial intelligence (AI), or by AI-assisted technologies.

Referenced text has been flagged by:

1. Using italic fonts, **and**
2. using quotation marks "...", **and**
3. explicitly mentioning the source in the text.

This work is dedicated to my grandfather Amarnath Sulhan.

Acknowledgements

My profound appreciation goes out to everyone who helped and advised me during this process.

First of all, I would want to express my gratitude to **Prof.Ian Morris** for his wise counsel, encouraging words, and helpful criticism that greatly influenced the course of this project.

I would also want to thank **Queen Mary University of London** for providing the resources and environment necessary for conducting the research.

Lastly, I would want to express my sincere gratitude to my friends, family, and coworkers for their unwavering support and encouragement, which kept me going during this journey.

Abstract

Differential privacy is an essential technique for safeguarding individual privacy in data analysis, particularly in sensitive domains such as crime data. It is a powerful concept that guarantees the protection of personal information by ensuring that the analysis outcomes are not significantly affected by the inclusion or exclusion of any individual in a dataset. This report delves into the theoretical foundations and practical implementations of differential privacy, with a specific focus on crime dataset of Los Angeles, U.S.A.

We delve into mechanisms such as the Laplace and discuss about Gaussian mechanism. The discussion revolves around the trade-offs between privacy and utility, highlighting the significance of choosing the right privacy budget (ϵ).

This report highlights the significance of preserving privacy while ensuring the usefulness of crime data through the application of differential privacy.

Contents

1 Introduction **7**

1.1 Motivation for this work 7

1.2 Background 8

2 Literature Review **9**

3 Basic Technique **11**

3.1 Randomised Response 11

3.2 Laplace Mechanism 11

3.2.1 Differential Privacy (ϵ-differential privacy) 12

3.2.2 Laplace Distribution 12

3.2.3 Sensitivity (Δf) 12

3.2.4 Global Sensitivity 13

3.2.5 Mechanism 13

3.3 Gaussian Mechanism 13

3.4 Lower Bounds 14

4 Need of Differential privacy **15**

4.1 Privacy Attacks 15

4.2 Obscuring Crime Locations with Noise 16

5 Data Collection and Preparation **18**

5.1 Libraries Used 18

5.2 Dataset Source and Information 19
5.3 Data Cleaning and Preprocessing 20

6 Applying Laplace Mechanism **23**
6.1 Setting Privacy Parameters 23
6.2 Scale Parameter 23
6.3 Applying Laplace noise 23
6.4 Applying Laplace Noise to Latitude and Longitude Columns . 24

7 Exploratory Data Analysis (EDA) **25**
7.1 Quantitative comparison 25
7.2 Summary statistics of noisy and original coordinates 26
7.3 Distribution of coordinates 26
7.4 Correlation Analysis 27
7.5 Spatial Visualizations 28
7.6 Temporal Analysis 28
7.7 Cluster Analysis 29

8 Results and Conclusions **30**

9 Improvements and Future Work **38**
9.1 Exploration of Alternative Noise Mechanisms 38
9.2 Utilize Batch Processing 39
9.3 Use of Hierarchical Models for Differential Privacy 39

Chapter 1

Introduction

The rise of big data has led to significant advancements in data-driven decision-making. However, it also brings up concerns regarding personal privacy. Differential privacy offers a robust mathematical framework for safeguarding privacy while enabling valuable data analysis. This work explores the principles of differential privacy, its application to sensitive datasets like crime data, and the potential vulnerabilities that can arise in privacy-preserving systems.

1.1 Motivation for this work

Protecting people's privacy is more important than ever as data collection becomes more and more important to making decisions. Giving out exact coordinates in crime datasets can violate privacy by revealing private details about victims, bystanders, or suspects. Most of the time, traditional methods of anonymisation don't offer enough security, especially when used with data from outside sources.

One option is differential privacy, which adds noise to the data so that it can't be used to identify people again while still keeping its usefulness. The goal of this work is to use differential privacy on crime dataset location data

in a way that protects privacy while also giving us useful information. In this way, we can safely and privately look at crime trends without putting people's privacy at risk.

1.2 Background

Differential privacy is a statistical method that guarantees the privacy of individual data entries in a dataset by adding random noise to the data. The objective is to facilitate the examination of datasets while safeguarding the confidentiality of persons. Differential privacy has emerged as the leading method for safeguarding privacy during data analysis, particularly in situations involving confidential data. It can be used to safeguard geographic data, particularly in cases where disclosing longitude and latitude coordinates could potentially expose sensitive information.

Chapter 2

Literature Review

When it comes to protecting sensitive information during data analysis, differential privacy has quickly become the method of choice. [1]Dwork et al (2006) initially proposed it as a way to guarantee robust privacy by making sure that an individual's data does not substantially impact the results of any research, regardless of whether it is in a dataset or not. The goal is to make it harder for attackers to deduce specific characteristics about individuals from aggregate query results by adding noise to the data.

There are special concerns about the privacy of geo-spatial data, including coordinates in the form of longitude and latitude. Because it might disclose private information such as where people work, live, and their habits, location data can be especially delicate. Differential privacy in geographical settings has been the subject of multiple investigations. An example of a system that secures location data is geo-indistinguishability, which was created by [4]Andres et al (2013). It uses noise to mask geographic coordinates, making it very unlikely that a place can be identified. The groundwork for using differential privacy in situations where location data is common has been created by this work.

Among the many popular approaches to differential privacy, the Laplace mechanism stands out. The process entails incorporating Laplace-distributed

noise into the data, with the amount of noise adjusted according to the query's sensitivity and the privacy budget (ϵ). A number of researchers, notably [2]McSherry (2009), have shown that the Laplace mechanism works well with several kinds of data, including continuous data like geographic coordinates.

Although differential privacy provides strong security, a significant obstacle is finding a balance between privacy and the usefulness of data. This trade-off has prompted researchers to investigate tools like improved composition methods and the Gaussian mechanism.

Applying differential privacy to coordinates safeguards sensitive location information in the context of crime data. In public safety applications, where anonymisation is not enough to prevent re-identification threats, differential privacy can successfully safeguard data [3]Bittau et al., 2017; others, 2017).

Chapter 3

Basic Technique

3.1 Randomised Response

Randomised response is a technique that was initially developed to protect the privacy of respondents when answering sensitive questions in surveys. Randomised response operates in the following manner in its original context: Participants are presented with a delicate inquiry (e.g., "Have you engaged in any illegal activities?"). Instead of providing a direct response, the individual resorts to a randomising method such as flipping a coin or using another similar device. When the coin lands on heads, they are honest in their response. However, if it lands on tails, they always answer "yes" without considering the truth. This process incorporates a level of randomness into the response, which effectively safeguards the privacy of the individual while enabling the data collector to make estimations about the overall distribution of answers.

3.2 Laplace Mechanism

The Laplace mechanism is a fundamental technique employed to guarantee differential privacy. The process entails incorporating noise sampled from a

Laplace distribution into the output of a query function. The level of noise is adjusted according to the sensitivity of the query and the privacy budget (ϵ) in order to uphold the privacy of individual entries in a dataset.

3.2.1 Differential Privacy (ϵ-differential privacy)

A privacy framework that ensures consistent query results, regardless of whether or not an individual's data is part of the dataset. One way to accomplish this is by introducing noise to the query output, effectively concealing the data of each individual.

3.2.2 Laplace Distribution

A probability distribution that looks like a double-exponential distribution is the Laplace distribution. It is utilised to provide noise for the Laplace mechanism and is symmetrical around zero. The Laplace distribution's probability density function is expressed as:

$$\text{Lap}(b) = \frac{1}{2b} \exp\left(-\frac{|x|}{b}\right)$$

where b is the scale parameter of the distribution, and x is a variable.

3.2.3 Sensitivity (Δf)

Understanding the sensitivity of a function is crucial for analysing how much the function's output can vary with a single change in the input dataset. In a formal context, the sensitivity of a query function f can be defined as follows:

$$\Delta f = \max_{x,x'} |f(x) - f(x')|$$

where x and x' are datasets differing in a single entry. The higher the sensitivity, the more noise must be added to ensure privacy.

3.2.4 Global Sensitivity

The differential privacy metric that is most frequently employed is global sensitivity. The output of a function is the most significant change across all inputs when a single dataset record is modified. In a crime dataset, the total would be influenced by the addition or removal of a single incident. In order to maintain privacy, queries with increased sensitivity necessitate additional noise, which may result in decreased accuracy.

3.2.5 Mechanism

The Laplace mechanism takes a dataset D, a query function f, and returns the function $f(D)$ with the addition of noise sampled from a Laplace distribution. The level of noise is adjusted according to the function's sensitivity and the privacy parameter (ϵ):

$$\text{Noisy Output} = f(D) + \text{Lap}\left(\frac{\Delta f}{\epsilon}\right)$$

Where:

- $f(D)$ is the true answer to the query.
- $\text{Lap}\left(\frac{\Delta f}{\epsilon}\right)$ is the Laplace noise, with scale $\frac{\Delta f}{\epsilon}$.

The additional noise guarantees that the resulting output cannot be distinguished from outputs produced by datasets that only differ by the data of a single individual.

3.3 Gaussian Mechanism

When working with more complicated data processing tasks, the Gaussian mechanism can be an essential tool in differential privacy. The Gaussian method, similar to the Laplace mechanism, adds noise to the query output

to make sure that the findings are unaffected by the existence or absence of any individual in the dataset. Nonetheless, the Laplace distribution and the Gaussian distribution are not interchangeable; the former draws its noise from the former, while the latter uses a normal distribution.

A symmetrical distribution of noise around the mean characterises the Gaussian distribution, which is defined by the mean and standard deviation. The sensitivity of the query and the privacy parameter, epsilon (ϵ), like the Laplace mechanism, decide the amount of noise produced in the Gaussian method. Nevertheless, epsilon alone provides stringent privacy assurances, but the Gaussian technique adds another parameter, delta (δ), which permits a relaxation of these constraints. Because of this easing, the Gaussian technique shines in situations when the data is very sensitive or the query function is very variable.

3.4 Lower Bounds

The bare minimum of noise that needs to be supplied to a query result in order to guarantee differential privacy is known as the lower bound. Finding the sweet spot between noise and utility is crucial for any dataset, but it takes on added significance when dealing with crime datasets, where data sensitivity might be high owing to the sensitive nature of the material.

For instance, the sensitivity of the query has a direct correlation with the amount of noise introduced when using the Laplace or Gaussian method on a crime dataset. How much a single person's data can impact the query result is measured by sensitivity. The use of lower bounds guarantees that a sufficient amount of noise is introduced to mask any discernible impact of individual data points.

Chapter 4

Need of Differential privacy

4.1 Privacy Attacks

When crime data includes coordinates that may be connected to other datasets, such as publicly accessible maps and demographic data, there is a high risk of re-identification through **reconstruction assaults**. As discussed, if the provided coordinates are in close proximity to a known address in the dataset, notwithstanding certain inaccuracies, an adversary could potentially deduce the precise location by cross-referencing it with additional data.

Moreover, reconstruction assaults were demonstrated to be effective in the U.S. Census despite the implementation of differential privacy techniques. Attackers could accurately infer private details about individuals by cross-referencing publicly available information with chaotic statistics [5]. The necessity for ongoing innovation in privacy-preserving techniques is emphasised by these challenges.

In order to reduce these risks, we need to take into account not just the sensitivity of the location data, but also the wider circumstances in which the data will be utilised and the probability that adversaries may connect it with additional datasets. Accurately adjusting the noise introduced by the Laplace process is crucial for safeguarding privacy while enabling meaningful

spatial analysis of crime data.

4.2 Obscuring Crime Locations with Noise

When analysing crime data, it is essential to use differential privacy to safeguard the privacy of individuals involved, especially victims and witnesses. This involves recording the latitude and longitude coordinates of crime locations. The Laplace mechanism is a commonly employed technique for achieving this objective by introducing random perturbations to the sensitive coordinates (latitude and longitude) prior to disclosing the data for analysis.

The Laplace process introduces noise that is directly proportional to the sensitivity of the data and the privacy budget (ϵ) . In the context of geographic coordinates, sensitivity refers to the extent to which the distance between two points, such as the actual crime location and a nearby point, can be altered by the inclusion or exclusion of a single data point in the dataset. Precise coordinates are crucial because they may readily pinpoint exact places, such as an individual's residence or place of employment, when combined with other accessible data.

When implementing the Laplace mechanism to conceal the locations of crime scenes, noise is generated from a Laplace distribution and applied to both the latitude and longitude coordinates. The level of noise introduced is determined by the sensitivity of the location data, which is usually calculated by considering the maximum possible change in distance when a single point is altered, along with the privacy parameter (ϵ). The coordinates with added noise are then released, preventing an attacker from accurately determining the precise location of a crime scene.

Mathematically, the noisy coordinates for a crime site, represented by

latitude (Lat) and longitude (Long), are calculated as:

$$\text{Lat}_{\text{noisy}} = \text{Lat} + \text{Laplace}\left(\frac{\Delta f}{\epsilon}\right)$$

$$\text{Long}_{\text{noisy}} = \text{Long} + \text{Laplace}\left(\frac{\Delta f}{\epsilon}\right)$$

The inclusion of this noise guarantees that the precise identification of any specific location (such as the residence of a crime victim) is not possible, thereby safeguarding individuals from potential re-identification. Nevertheless, the selection of the privacy budget (ϵ) is of utmost importance. A decreased value of (ϵ) enhances privacy by introducing a greater amount of noise; however, it diminishes the precision of the reported locations, hence complicating the analysis of spatial crime patterns.

Chapter 5

Data Collection and Preparation

5.1 Libraries Used

The code imports some essential Python modules commonly employed for data manipulation, statistical analysis, visualisation, and distance calculations. Here is a breakdown of each library and its function in the analysis:

- **Pandas:** Pandas is utilised for loading datasets, such as through the usage of the `pd.read_csv()` function. It is also employed for data cleaning tasks, such as managing missing values, and for preparing the data for analysis, such as grouping data based on dates.
- **NumPy:** NumPy is an essential library for doing numerical computations in the Python programming language. The `np.random.laplace()` function is utilised to generate random noise from the Laplace distribution, which is then employed to build the differential privacy method.
- **Matplotlib:** Matplotlib is a widely-used charting library in Python. Within Matplotlib, `pyplot` is a module that offers a MATLAB-like

interface for generating static, animated, and interactive visualisations. Pyplot is utilised to generate visual representations, such as scatter plots and line plots, for the purpose of comparing the original data with the data that contains noise.

- **Seaborn:** Seaborn is a Python package for visualising data that is constructed on the foundation of Matplotlib. This tool is commonly employed for generating complex visual representations such as heatmaps and KDE (Kernel Density Estimation) plots.
- **Scikit-learn:** 'KMeans' is a clustering algorithm used in the Scikit-learn library.
- **SciPy:** Scipy's spatial distance module provides a range of functions for calculating distances between objects.The pdist function, available in the SciPy library, is a powerful tool for calculating pairwise distances between observations in n-dimensional space.

5.2 Dataset Source and Information

The dataset utilised in this research comprises crime records obtained from the city of **LA,Chicago**, encompassing the time period from 2020. The dataset is accessible to the public on **Kaggle** and provides comprehensive data on recorded criminal activities, including the nature of the offence, date, location, and supplementary characteristics. This research explicitly examines the geospatial coordinates provided in the dataset, Latitude (LAT) and Longitude (LON).

A comprehensive description of each incident is provided by the dataset's extensive collection of attributes. Some of the most significant columns are as follows:

- **ID:** Unique identifier for each crime incident.

- **Date:** Date and time of the crime.
- **Primary Type:** Category of the crime (e.g., THEFT, BATTERY, NARCOTICS).
- **Description:** Detailed description of the crime.
- **Location Description:** Description of the location where the crime occurred (e.g., STREET, RESIDENCE, SCHOOL).
- **Arrest:** Boolean indicating whether an arrest was made.
- **Latitude (LAT):** The latitude coordinate representing the crime location.
- **Longitude (LON):** The longitude coordinate representing the crime location.
- **Community Area:** Community area number in LA, Chicago where the crime took place.

5.3 Data Cleaning and Preprocessing

The subsequent data cleaning and preprocessing processes were implemented to guarantee the dataset's integrity and quality prior to the application of differential privacy:

- **Date Formatting:** The `Date OCC` column, initially comprising both date and time details, underwent reformatting to exclusively include the date. The conversion was crucial for conducting temporal-spatial analysis, as it enabled to categorise the data by days without the inclusion of time components, which would have made the study more complex.

- **Geographic Bounds Filtering:** The dataset was refined to only contain crime records that were inside the specified geographical boundaries of the city, Los Angeles.

  These variables establish the geographical limits for latitude and longitude. The coordinates in this instance are exclusive to a certain geographical area, that is, Los Angeles. The purpose of these boundaries is to exclude data points that are located outside the specified area, so guaranteeing that only pertinent data is retained for subsequent analysis.

```
# Geographic bounds of Chicago (Source: Google)
min_lat, max_lat = 33.700, 34.350
min_long, max_long = -118.670, -118.150

# rows within the geographic bounds for Chicago
bounded_data = crime_data_LA[
    (crime_data_LA['LAT'] >= min_lat) &
    (crime_data_LA['LAT'] <= max_lat) &
    (crime_data_LA['LON'] >= min_long) &
    (crime_data_LA['LON'] <= max_long)
]

rem_rows = len(crime_data_LA) - len(bounded_data)
print(f"Number of rows removed due to anomalies: {rem_rows}")
```

Figure 5.1: Filtering Geographical Bounds Code

- **Counting Removed Rows:** The number of rows that have been eliminated by subtracting the number of rows in the cleaned dataset from the number of rows in the original dataset. The purpose of printing this statistic is to provide feedback on the amount of data that was eliminated as a result of abnormalities.

- **Handling missing values:** The algorithm initially examines the dataset to see whether there are any absent values in the LAT (latitude) and LON (longitude) columns.

- **Randomised Rows:** During the data processing phase, we selected a random sample of **50,000** rows from the dataset, specifically targeting the latitude and longitude columns that had been cleaned. This was done to ensure computational efficiency, since processing the entire set of 100,000 entries would place excessive strain on the hardware.

```
# Extracting the cleaned Latitude and Longitude columns

coord_cleaned = bound_data[['LAT', 'LON']].dropna().sample(n=50000, random_state=42)


# Changing coordinates to radians
coord_rad_cleaned = np.radians(coord_cleaned)

# Pairwise Euclidean distance

distances = pdist(coord_rad_cleaned, metric='euclidean')

# maximum pairwise distance (global sensitivity)

max_dist = np.max(distances)

print(f"The maximum pairwise distance (global sensitivity): {max_dist}")
```

Figure 5.2: Randomised Rows Code

Significantly, even if we were to utilise the complete dataset, the global sensitivity, quantified as the highest pairwise **Euclidean distance**, would dependably yield a value of **0.011** (i.e., when rounded to the second decimal place), which is identical to the result obtained when utilising a random sample of 50,000 rows. This suggests that the sampling method used does not undermine the accuracy of the global sensitivity measure, making it a feasible and successful strategy given the limitations.

Chapter 6

Applying Laplace Mechanism

6.1 Setting Privacy Parameters

- The privacy budget (ϵ) was set to **0.5**, balancing privacy and utility.
- Global Sensitivity was calculated as **0.011**, representing the maximum change when a data point is added or removed.

6.2 Scale Parameter

The Laplace distribution scale parameter is utilised for generating noise. The scale is determined by dividing the sensitivity by epsilon. This guarantees that the amount of noise introduced into the data is directly proportional to the privacy budget. For this particular scenario, the scale would be calculated as 0.011 divided by 0.5, resulting in a value of **0.022**.

6.3 Applying Laplace noise

Differential privacy was applied by adding noise to the latitude and longitude coordinates using the Laplace distribution. The `np.random.laplace()`

```python
# Applying Laplace noise
def add_laplace_noise(value, epsilon, sensitivity):
    scale = sensitivity / epsilon
    noise = np.random.laplace(0, scale)
    return value + noise
```

Figure 6.1: Enter Caption

function was used to generate noise for both the latitude and longitude.

- **Scale** = sensitivity / epsilon: The scale for the Laplace noise is recalculated based on the provided sensitivity and epsilon.
- `np.random.laplace(0, scale)`: This generates noise from a Laplace distribution with mean 0 and the calculated scale.
- Return value + Noise: The noisy value is returned by adding the generated noise to the original coordinate value.

6.4 Applying Laplace Noise to Latitude and Longitude Columns

- The `add_laplace_noise` function is applied to each value in the LAT and LON columns using the `apply()` method.
- **Lambda function:** `lambda x: add_laplace_noise(x, epsilon, global_sensitivity)` is used to apply the noise to each latitude (LAT) and longitude (LON) value.
- The noisy latitude values are stored in a new column called `Latitude_DP`, and the noisy longitude values are stored in `Longitude_DP`.

Chapter 7

Exploratory Data Analysis (EDA)

7.1 Quantitative comparison

Computed the disparities between the initial geographical coordinates (latitude and longitude) and their differentially private (DP) counterparts. It then presents these disparities using histograms that include superimposed Kernel Density Estimates (KDE).

```
# Quantitative comparison (original and noisy data)
coord_cleaned['Lat_Difference'] = coord_cleaned['LAT'] - coord_cleaned['Latitude_DP']
coord_cleaned['Long_Difference'] = coord_cleaned['LON'] - coord_cleaned['Longitude_DP']

coord_cleaned['Lat_Difference'] , coord_cleaned['Long_Difference']

# Displaying the distribution of differences
plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
sns.histplot(coord_cleaned['Lat_Difference'], kde=True, color='blue')
plt.title('Distribution of Latitude Differences:')

plt.subplot(1, 2, 2)
sns.histplot(coord_cleaned['Long_Difference'], kde=True, color='red')
plt.title('Distribution of Longitude Differences: ')

plt.show()
```

Figure 7.1: Quantitative comparison code

7.2 Summary statistics of noisy and original coordinates

Comparing the differentially private data Latitude(DP) and Longitude(DP) to the original data, the statistics show that there are only small differences caused by noise. Keeping the overall geographic centre in mind, the means and medians of the original and differentially private datasets are quite close. While differential privacy is supposed to safeguard the privacy of individual data points, the slightly larger standard deviations and wider ranges in the differentially private data indicate that the additional noise has created some dispersion. This proves that the privacy mechanism has balanced privacy with utility by modifying the data while preserving its overall structure and distribution.

```
Summary Statistics (Original Data:)
                LAT            LON
count  50000.000000  50000.000000
mean      34.075068   -118.355848
std        0.111270      0.104728
min       33.706500   -118.664400
25%       34.016300   -118.431800
50%       34.059500   -118.324800
75%       34.166400   -118.274900
max       34.328700   -118.156000

Summary Statistics (Differentially Private):
       Latitude_DP  Longitude_DP
count  50000.000000  50000.000000
mean      34.075212   -118.355914
std        0.115473      0.109389
min       33.613178   -118.803870
25%       34.008985   -118.430848
50%       34.064229   -118.330622
75%       34.162000   -118.273039
max       34.458302   -118.016278
```

Figure 7.2: Summary statistics

7.3 Distribution of coordinates

Displaying the distributions of both the original and differentially private (DP) versions of the latitude and longitude variables. It compares the original data (blue) with the noisy DP data (red) by creating side-by-side histograms using Kernel Density Estimates (KDE) for both longitude and latitude. This

makes it easy to see visually how the DP mechanism changed the data while keeping distribution patterns overall.

```
# Visualizing the distribution of Latitude and Noisy Latitude
plt.figure(figsize=(12, 6))

# Latitude
plt.subplot(1, 2, 1)
sns.histplot(coord_cleaned['LAT'], kde=True, color='blue', label='Original Latitude')
plt.title('Latitude Distribution (Original)')
plt.legend()

# Noisy Latitude
plt.subplot(1, 2, 2)
sns.histplot(coord_cleaned['Latitude_DP'], kde=True, color='red', label='Noisy Latitude')
plt.title('Latitude Distribution (Noisy)')
plt.legend()

plt.tight_layout()
plt.show()

# Visualizing the distribution of Longitude and Noisy Longitude
plt.figure(figsize=(12, 6))

# Longitude
plt.subplot(1, 2, 1)
sns.histplot(coord_cleaned['LON'], kde=True, color='blue', label='Original Longitude')
plt.title('Longitude Distribution (Original)')
plt.legend()

# Noisy Longitude
plt.subplot(1, 2, 2)
sns.histplot(coord_cleaned['Longitude_DP'], kde=True, color='red', label='Noisy Longitude')
plt.title('Longitude Distribution (Noisy)')
plt.legend()

plt.tight_layout()
plt.show()
```

Figure 7.3: Visualizing the distribution

7.4 Correlation Analysis

The correlation analysis was one of the initial analyses carried out. After applying differential privacy, the goal was to determine if the spatial linkages between the original and noisy coordinates were maintained. The goal was to determine the Pearson correlation between the original longitude (LON) and noisy longitude (Longitude DP), as well as between the original latitude (LAT) and noisy latitude (Latitude DP), using the `pandas.Series.corr()` function. High correlations (**close to 1**) suggest that the noisy data still preserves the relationships in the original data, indicating good utility.

```python
# Extracting the necessary columns
lat_clean = coord_cleaned['LAT']
lat_DP_clean = coord_cleaned['Latitude_DP']
lon_clean = coord_cleaned['LON']
lon_DP_clean = coord_cleaned['Longitude_DP']


# Correlation Analysis
lat_corr = lat_clean.corr(lat_DP_clean)
lon_corr = lon_clean.corr(lon_DP_clean)

print(f"Correlation between original and noisy Latitude: {lat_corr}")
print(f"Correlation between original and noisy Longitude: {lon_corr}")
```

Figure 7.4: Correlation Analysis

7.5 Spatial Visualizations

Further evaluation of differential privacy's effects was accomplished through the use of spatial visualisations. Before and after the addition of noise, the distribution of crime sites was visualised using scatter plots and heatmaps. We were able to directly compare the original and noisy coordinates by creating scatter plots using the `matplotlib.pyplot.scatter()` function.

7.6 Temporal Analysis

To determine if the patterns of crime occurrences over time were maintained when differential privacy was applied, temporal analysis was carried out. The number of crimes happening on each day was calculated by grouping the dataset by the `Date_OCC` column, which has been reformatted to include only the date. To accomplish this, we utilised the `pandas.DataFrame.groupby()` method. Then, to create time series graphs, we utilised the `matplotlib.pyplot.plot()` function.

7.7 Cluster Analysis

Lastly, the effectiveness of differential privacy in preserving the clustering of crime locations was assessed by a cluster analysis. In order to find groups of similar crime scenes, the original and noisy coordinates were clustered, using the `sklearn.cluster.KMeans` algorithm. Cluster analysis on both the original and noisy datasets were performed to check if the clusters of crime locations remain same.

```
# clustering (Original data)
kmeans_o = KMeans(n_clusters=5).fit(coord_cleaned[['LON', 'LAT']])
coord_cleaned['Cluster_Original'] = kmeans_o.labels_

# clustering (Noisy data)
kmeans_n = KMeans(n_clusters=5).fit(coord_cleaned[['Longitude_DP', 'Latitude_DP']])
coord_cleaned['Cluster_Noisy'] = kmeans_n.labels_

# Comparing both clusters
plt.figure(figsize=(10, 5))

plt.subplot(1, 2, 1)
plt.scatter(coord_cleaned['LON'], coord_cleaned['LAT'], c=coord_cleaned['Cluster_Original'], cmap='viridis')
plt.title('Clusters - Original Data')

plt.subplot(1, 2, 2)
plt.scatter(coord_cleaned['Longitude_DP'], coord_cleaned['Latitude_DP'], c=coord_cleaned['Cluster_Noisy'], cmap='viridis')
plt.title('Clusters - Noisy Data')

plt.show()
```

Figure 7.5: Cluster Analysis

Chapter 8

Results and Conclusions

By contrasting the original data with the noisy data, we were able to assess how well the differential privacy application worked:

- **Distribution of Original and Noisy Coordinates:** The graphs illustrate the distribution of latitude values in the original dataset and a noisy dataset. The initial distribution, represented in blue, exhibits clear, pronounced peaks, suggesting that latitude values are clustered in specific places. Conversely, the distribution that is characterised by noise, depicted in red, maintains the general pattern but exhibits displacements and a more polished appearance, which indicates the influence of additional noise for the sake of privacy. The smoothing effect of this process somewhat conceals the exact specifics of the original data while preserving its overall structure, demonstrating a compromise between data confidentiality and precision.

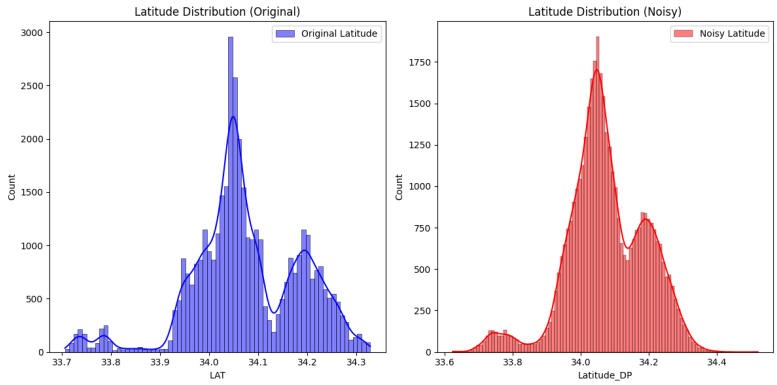


Figure 8.1: Latitude Distribution

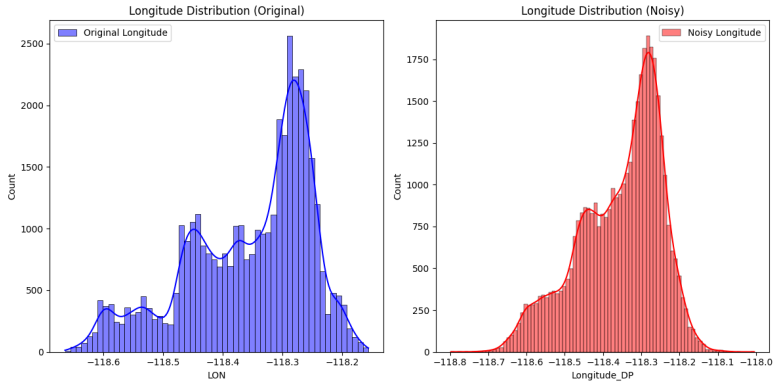


Figure 8.2: Longitude Distribution

- **Distribution of Coordinates Differences:** The figures illustrate the distributions of discrepancies between the original latitude and longitude values and their altered counterparts, which were presumably implemented to safeguard data confidentiality. Both distributions display a pronounced peak around zero, suggesting that the bulk of these differences are minimal, indicating that the perturbed coordinates remain in close proximity to the original values. The symmetrical and tightly clustered nature of these distributions indicates that the disturbances are uniformly distributed around the initial values, resulting in mini-

mal divergence. This indicates that the employed noise or perturbation method has successfully maintained the precision of the geographic data while introducing minor alterations for the sake of privacy. The kernel density estimate (KDE) is superimposed on the histograms to provide a smooth image, which helps to emphasise the clustering of differences around zero.

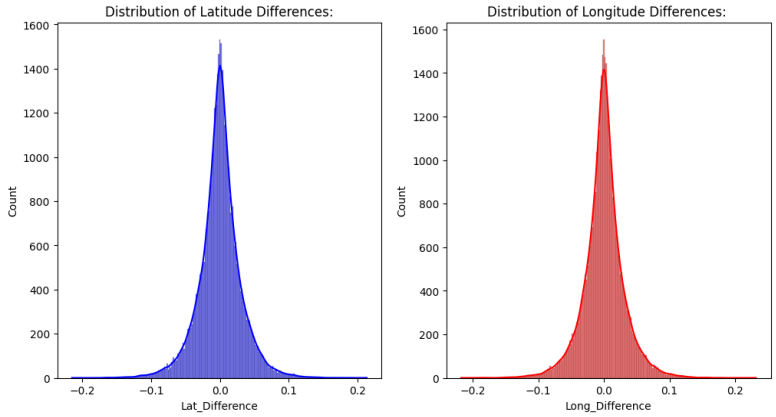


Figure 8.3: Distribution of Coordinates Differences

- **Correlation Comparison:** The results showed a moderate to high level of correlation between the original and noisy coordinates, indicating that the spatial relationships in the dataset were not considerably disrupted by the noise created by differential privacy. There was strong evidence from the correlation values that the Laplace noise maintained the data's broad geographic patterns. Correlation between original and noisy Latitude comes out to be **0.9631** and Correlation between original and noisy Longitude comes out to be **0.9579**.

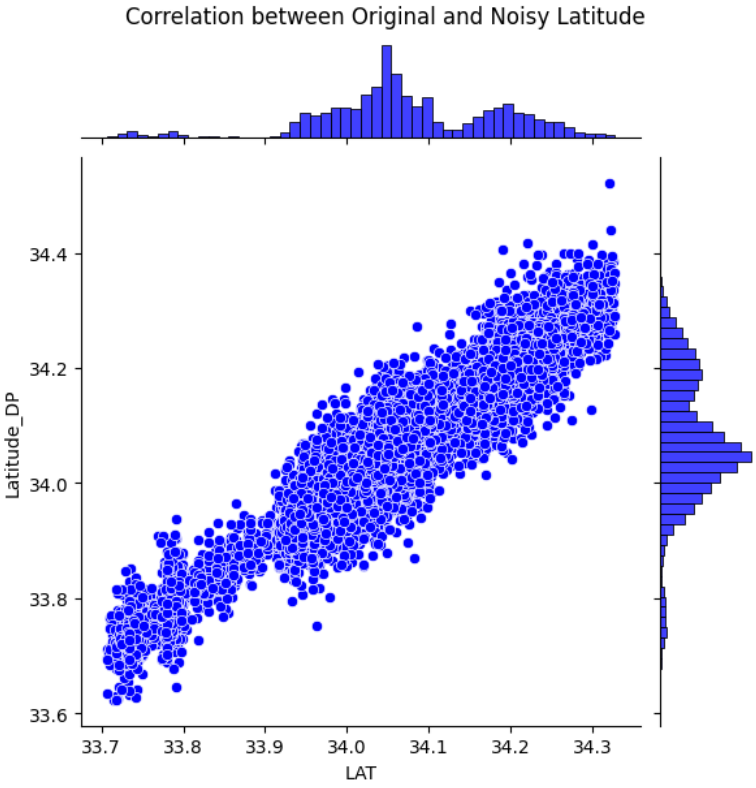


Figure 8.4: Correlation between Original and Noisy Latitude

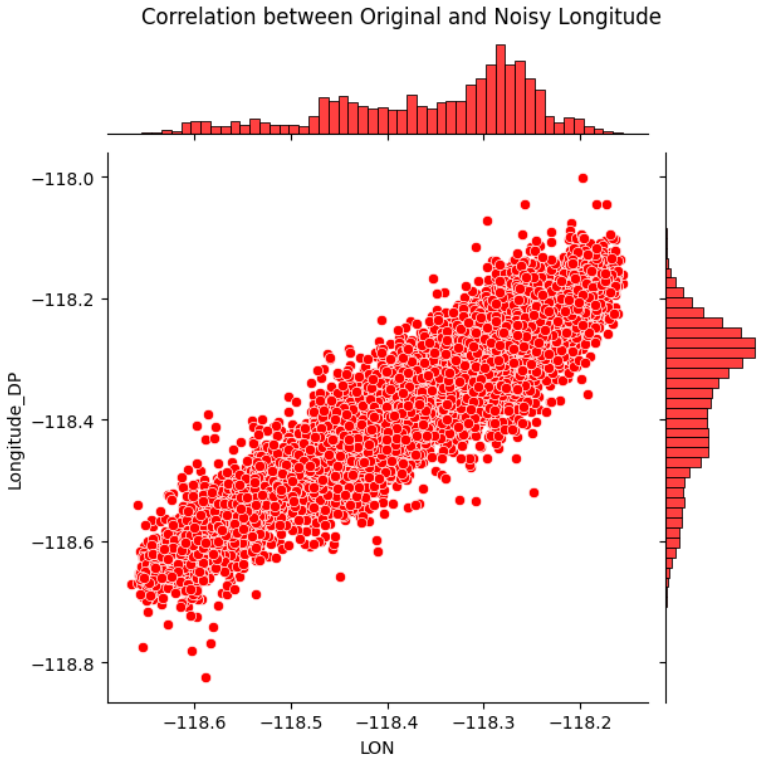


Figure 8.5: Correlation between Original and Noisy Longitude

- **Spatial Visualizations:** The noisy coordinates were seen to cluster near the original coordinates in these plots, however they did exhibit some dispersion as a result of the noise. This indicated that the differential privacy noise preserved the dataset's general geographical structure while protecting individual data points. To better understand the distribution of criminal activity across regions, heatmaps were created using the `seaborn.kdeplot()` method. According to the heatmaps, the key crime hotspots were the same in both the original and noisy datasets. Noise somewhat masked the edges of these hotspots, but overall patterns remained, suggesting that differential privacy did not drastically alter the data's central spatial trends.

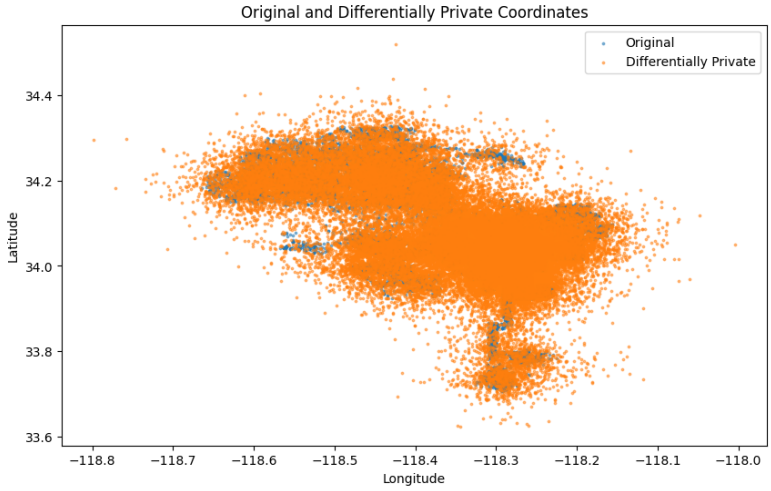


Figure 8.6: Crime Hotspots (Original and Noisy)

- **Temporal Analysis:** These graphs contrasted the daily crime counts from the clean dataset with those from the noisy one. Overall, the results demonstrated that the noisy data kept the temporal crime trends, with the noisy dataset mirroring the major peaks and troughs in crime occurrences. This suggested that the larger temporal structure of the data was unaffected by the noise introduced by differential privacy.

  As a result of the privacy mechanism's successful balancing of privacy protection and data utility in the temporal dimension, the larger trends were unaffected by the small deviations caused by random noise.

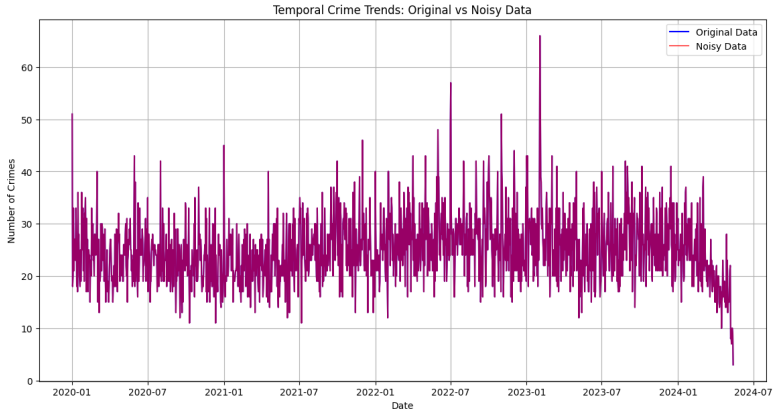


Figure 8.7: Temporal Crime Trends

- **Clustering:** Scatter plots created using the `matplotlibpyplotscatter()` tool were subsequently used to display these clusters. Although the borders between clusters were less apparent in the noisy dataset, the results demonstrated that the noisy coordinates yet generated clusters that were comparable to the original data. Particularly at the borders, where the noise was most concentrated, the cluster boundaries were rather blurry as a result of the additional noise.

  The fact that the differential privacy technique managed to keep the data's important spatial patterns intact suggests that the main clusters were still discernible despite this. Even after noise was applied, the clusters that matched major crime hotspots were mostly unaltered, enabling useful spatial analysis.

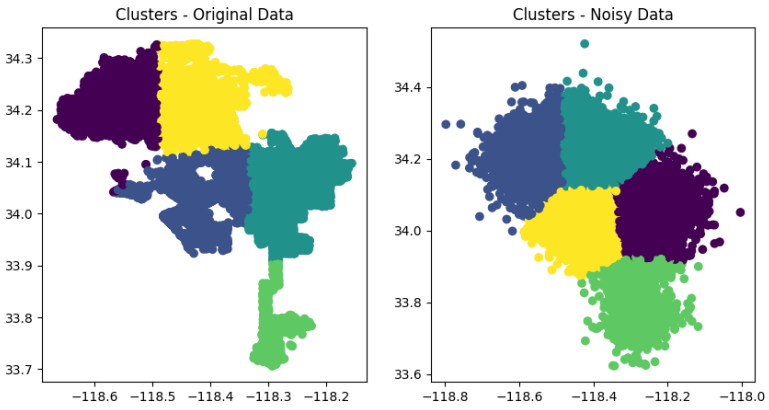


Figure 8.8: Original vs Noisy cluster

Chapter 9

Improvements and Future Work

Here are three key points for improvements and suggestions for future work:

9.1 Exploration of Alternative Noise Mechanisms

Although the Laplace mechanism is frequently employed for differential privacy, it may not always achieve the most favourable trade-off between privacy and data value, particularly in datasets with diverse sensitivities. Subsequent research could investigate alternate techniques, such as the Gaussian mechanism, which offers greater flexibility in introducing noise by integrating an additional parameter (δ) that can loosen the stringency of privacy assurances. This investigation could aid in identifying situations when alternative noise processes may surpass the Laplace mechanism, namely in maintaining data accuracy while also guaranteeing privacy.

9.2 Utilize Batch Processing

Instead of doing computations on the entire dataset simultaneously, the data might be partitioned into smaller, more manageable chunks. Each batch can be processed individually, and the outcomes can be combined afterwards. This method minimises the memory and computational burden at any given moment, enabling you to handle the entire dataset while yet preserving efficiency. In addition, if the infrastructure allows it, batch processing can be parallelised, resulting in further acceleration of the analysis.

9.3 Use of Hierarchical Models for Differential Privacy

Hierarchical models are applicable for implementing differential privacy across several levels of the data hierarchy, including city, neighbourhood, and street levels in a crime dataset. By implementing privacy noise at various levels, it is feasible to optimise the quantity of noise added according to the level of detail in the data. This method can assist in preserving a greater degree of data usefulness at larger geographical scales while also safeguarding privacy at more specific levels. Additionally, it enables the implementation of more advanced privacy controls that are customised to the individual requirements of various data layers.

Bibliography

[1] Dwork, C., McSherry, F., Nissim, K., Smith, A. (2006). *"Calibrating Noise to Sensitivity in Private Data Analysis."* Journal of Privacy and Confidentiality, 7(3), 17-30. doi:10.29012/jpc.v7i3.404.

[2] McSherry, F. (2009). *"Privacy Integrated Queries: An Extensible Platform for Privacy-Preserving Data Analysis."* Communications of the ACM, 53(9), 89-97. doi:10.1145/1787234.1787249.

[3] Bittau, A., Erlingsson, Ú., Maniatis, P., Mironov, I., Raghunathan, A., Lie, D., Boneh, D. (2017). *"Prochlo: Strong Privacy for Analytics in the Crowd."* In Proceedings of the 26th Symposium on Operating Systems Principles (pp. 441-459). doi:10.1145/3132747.3132769.

[4] Andres, M. E., Bordenabe, N. E., Chatzikokolakis, K., Palamidessi, C. (2013). *"Geo-Indistinguishability: Differential Privacy for Location-Based Systems."* In Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security (pp. 901-914). doi:10.1145/2508859.2516735.

[5] TechXplore. (2023). *Census Vulnerable to Privacy Attacks Despite Differential Privacy*. Available at: https://techxplore.com/news/2023-02-census-vulnerable-privacy.html#google_vignette.