The Washington Post's National Effort to Catalogue Fatal Police Shootings: Dissecting Racial and Age-Based Disparities through Clustering analysis and Logistic Regression Techniques.

Submitted By,

Pradeep Bolleddu - bpradeep@umassd.edu
Sourabh Pratapwar - spratapwar@umassd.edu
Abhishek Yernagula - ayernagula@umass.edu
Sohel Najeer Shaikh - sshaikh@umassd.edu

The Issues:

When investigating the police shooting dataset for the Data Science Report, potential biases in the collection of data create a challenge because various governments may have different reporting techniques. It might be challenging to make conclusions regarding demographics, court outcomes, and community impact indicators when there are contradictions and incomplete data. Since this is a delicate subject, it is important to maintain the privacy of every individual who engage. Furthermore, it's possible that certain details—such as the locations of these incidents and the interactions between the public and law enforcement—are absent from the report.
We need to thoroughly clean the data, be aware of any biases, and understand the limitations of the dataset to ensure that what we discover are accurate as well as fair.

Here are the main questions we want to answer:

1. What are the age, gender, and race of the individuals who have been shot by police, and are there any significant variations or patterns?
2. Are there any consistent patterns or variances in the public's behavior to police shootings?
3. Do police shootings follow any patterns over time, and do certain times see a rise or fall in the number of incidents?
4. Are there any areas with a high incidence rate of police shootings? How does geographic location affect t
5. What legal steps or measures are available to ensure that the officers involved in police shootings remain responsible, depending on the situation?
6.Do events surrounding police shootings influence public opinion?
7.Is there a correlation between events surrounding police shootings and public views?

Findings:

Findings for Police Shooting Dataset:

1. There are observable trends when age, gender, and racial demographics related to police shootings are examined. Although a correlation has been observed between specific demographic parameters and police shootings, it is important to recognize that this association may be strengthened with more data.
2. Examining the geographic distribution of shooting incidents involving police reveals specific hotspots or areas with greater incidence frequencies. A fuller understanding will require a larger dataset, even though the dataset reveals probable links between specific places and the prevalence of these instances.
3. Research on the legal outcomes of police shootings shows that the officers' outcomes vary. But the small size of the sample can limit how well legal

outcomes can be predicted, highlighting the need for a larger, more reliable dataset.

4. A study of how police shootings affect communities reveals a range of public reactions, demonstrations, and degrees of confidence between the public and law enforcement. The dataset indicates connections between the qualities of the community and the events that followed, but more information can clarify these conclusions.
5. Analyzing the various ways that the frequency and impact of police shootings have changed over time. While the data provides some fundamental understanding, further data collection is necessary for understanding the connections between these incidents and other events.
6. Examining connections between age, gender, and the outcome of court cases involving police shootings. We can gain so much from the limited dataset, though, and more data would enable us to improve the accuracy of these correlations.
7. Upon identifying outliers in the dataset, special conditions surrounding individual episodes are taken into consideration. When working with sensitive data, ethical issues must be addressed, and debates about responsible data use are critical to deriving fair and impartial conclusions from the dataset.
8. Findings of Distribution of Ages for black people and white people

```
Descriptive statistics of the distribution of ages for black people

Maximum Age 88.0
Minimum Age 8.0
mean of Age 32.99899142713061
standard deviation of Age 11.453543201042658
Median Age 31.0
kurtosis of Age 0.8622643324497616
skewness of Age 0.9581948659842209
—----------------------------------------
Descriptive statistics of the distribution of ages for white people
Maximum Age 91.0
Minimum Age 2.0
mean of Age 40.196642685851316
standard deviation of Age 13.145678731904734
Median Age 38.0
kurtosis of Age -0.09179403359817284
skewness of Age 0.5366708974598733
```

<u>Discussions:</u>

To truly comprehend the police shooting dataset and identify significant relationships, we thoroughly examined it in our study. Our objective was to use real data to produce meaningful forecasts without oversimplifying the situation. We had a problem since there were inconsistent amounts of data points (information on many things) available for variables including community effect, legal results, and demographics in relation to police shootings. Thus, by decreasing the dataset to __ data points, we were able to make it more focused and balanced.

As soon as the dataset was prepared, to facilitate the process of assembling the data for an in-depth investigation, we resolved this issue. For individual plots, we first used logistic regression, a technique that aids in the prediction of outcomes connected to police shootings; for points with __ input values, we used logistic regression. We used quadratic models to better understand the correlations since they explain changes in variables in greater detail, especially when the data is modified using logarithms. We were able to better grasp the links between these models by adding additional terms and features.

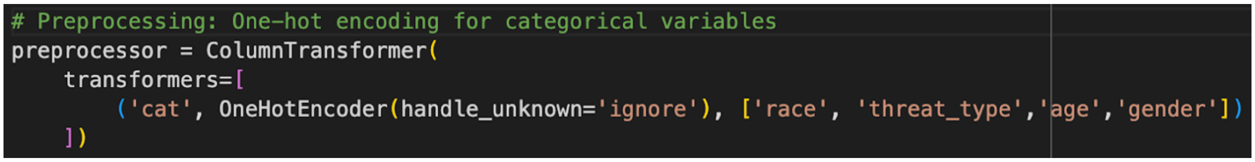
Knowing that our data had limitations, we investigated additional statistical tests such as T-tests and Monte Carlo simulations. To forecast the results of our study pertaining to police shootings, we employed logistic regression. To ensure the accuracy of our predictions and account for the complexities included in the police shooting dataset, our research adopts a variety of techniques.

We used Different kinds of Clustering Algo like K-means, DB SCAN & hierarchical clustering and Logistic regression.

Appendix A: Method:

1. Preparing Data for Logistic Regression Analysis

   The logistic regression model implemented necessitates specific input features to be in a format amenable for model processing. The categorical attributes, namely 'race', 'threat_type', and 'gender', undergo a transformation process utilizing One-Hot Encoding. This technique converts these categorical variables into a series of binary columns, each representing a unique category within the original attribute. This transformation is critical for adapting categorical data for compatibility with the logistic regression algorithm, which inherently requires numerical input. The following code snippet illustrates the application of One-Hot Encoding to these specified columns:



```
# Preprocessing: One-hot encoding for categorical variables
preprocessor = ColumnTransformer(
    transformers=[
        ('cat', OneHotEncoder(handle_unknown='ignore'), ['race', 'threat_type','age','gender'])
    ])
```

2. Variable creation:

Logistic regression Variables

The variables used to make predictions using logistic regression are as below:

(i) ‘was_mental_illness_related’: The column in the dataset is to better capture instances where information from media sources or police documentation suggests the individual involved may have had prior mental health issues or was experiencing a mental health emergency at the time of the shooting.

(ii) ‘Age: The victim's age when the incident occurred.

(iii)‘threat_type’: The behaviors exhibited by the victim prior to the fatal shooting incident.

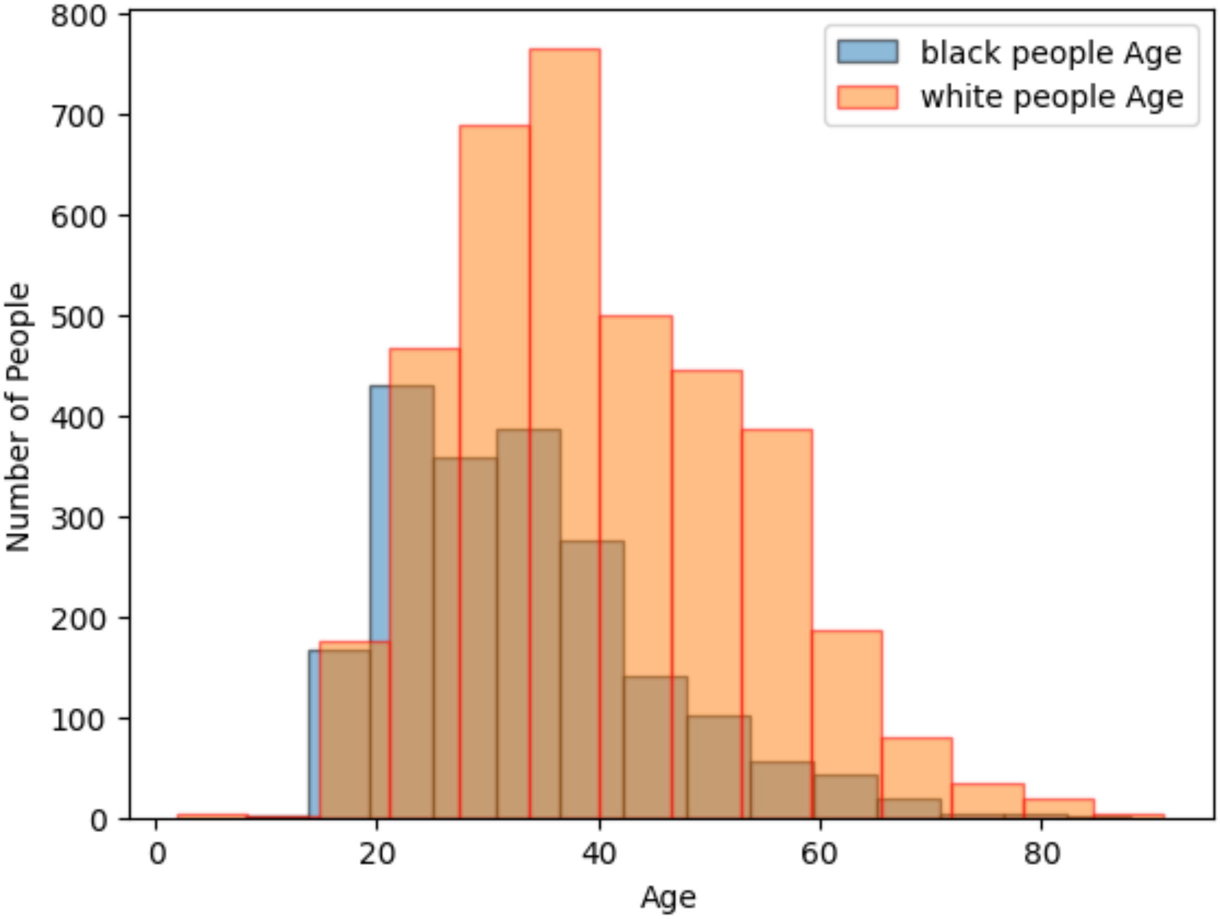
(iv) ‘race’: The known race and ethnicity of the victim, which can include multiple identifiers to account for individuals of mixed race or those with various racial and ethnic backgrounds.

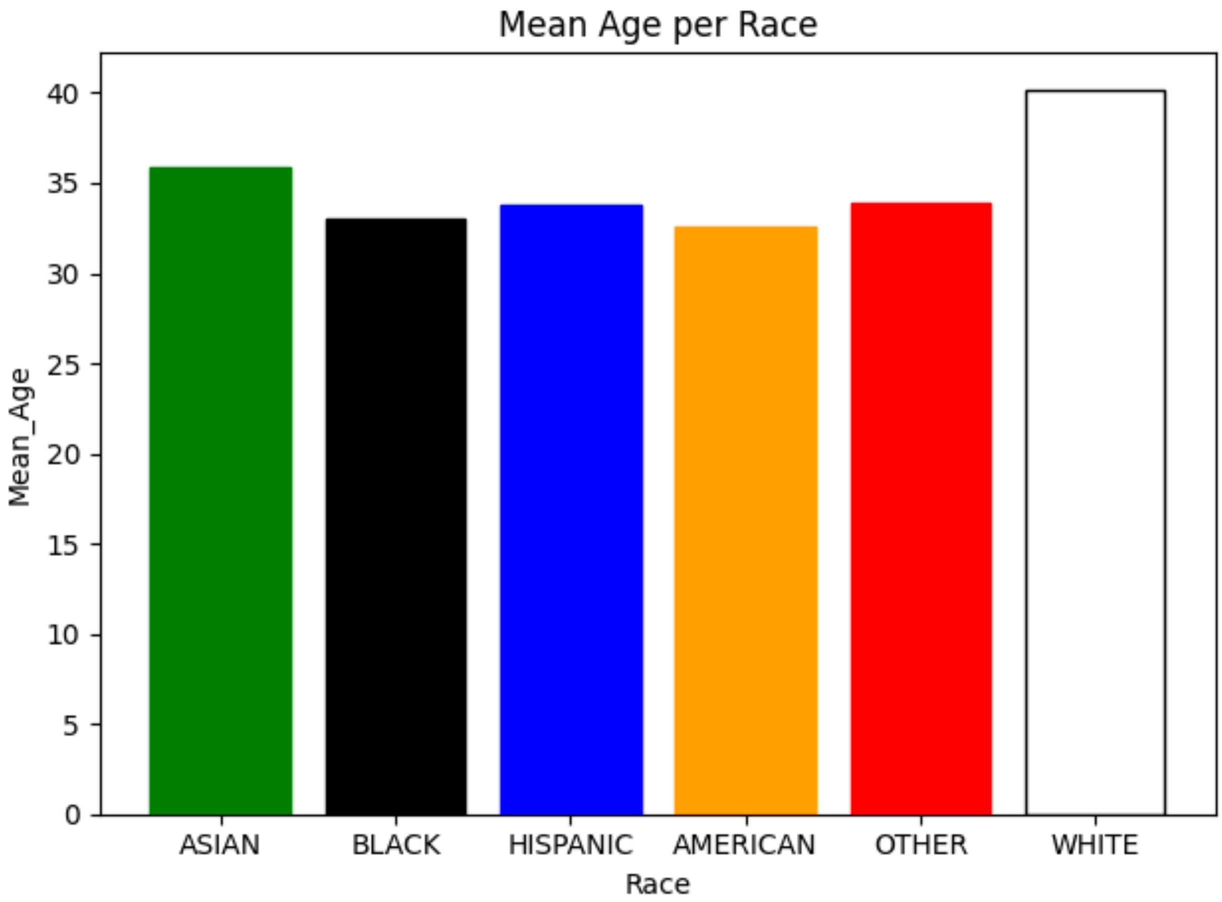
(v) ‘Gender’: The sex of the individual affected.

3. Analytic Methods:

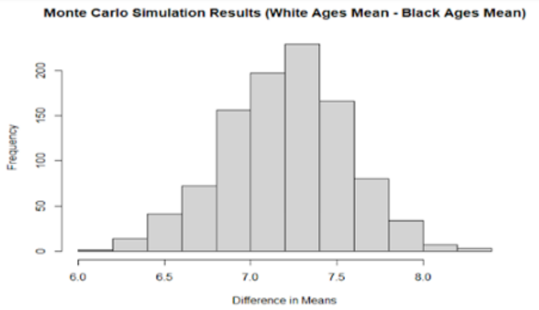
In this section, we are presenting a comprehensive analysis of the data columns to unearth deeper insights and understanding. Our objective is to extract more meaningful information from the data through these analytical techniques.

1.Enhanced Understanding: By applying analytical methods, we aim to achieve a clearer and more detailed perspective of the data, which aids in making informed decisions.
These methods enable us to identify underlying patterns and trends within the data, which can be crucial for strategic planning and forecasting.


black people Age
white people Age
Number of People
800
700
600
500
400
300
200
100
0
0
20
40
60
80
Age



The chart presents a visual comparison of average ages among various racial demographics, labeled as Asian, Black, Hispanic, American (likely referring to Native Americans), Other, and White. Each group is represented by a distinct color, such as green for Asian and blue for Hispanic, making the distinctions clear. The heights of the bars are quite similar, suggesting that the average ages across these groups don't vary widely. The highest average age is attributed to the 'Other' category, with 'White' slightly lower, while 'Asian' and 'Black' categories show the lowest average ages on the chart. However, the lack of numerical values on the vertical axis leaves the exact ages unspecified. The overall impression is one of relative age parity across racial lines, with a slight edge in higher average ages for the 'Other' and 'White' categories.



It mentions an "observed mean difference" of approximately 7.197 years and a "proportion of simulated differences >= observed difference" with a p-value of 0.522. This p-value indicates that there is not a statistically significant difference between the groups with resp`ect to the characteristic being measured at the common alpha levels of 0.05 or 0.01, since the p-value is greater than these thresholds.

Clustering

Clustering is a Type of Classification Algorithm; We Need to perform some clustering algorithms like K-Means and Density Based clustering.

1. K-means Clustering:

- Type: Centroid-based clustering.
- Mechanism: Partitions data into K clusters by minimizing the distance between data points and the centroid of clusters.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise):

Type: Density-based clustering.

Mechanism: Forms clusters based on dense regions of data points, considering two parameters: minimum points (MinPts) and epsilon (ε).

Hierarchical Clustering:
Type: Connectivity-based clustering.
Mechanism: Builds a hierarchy of clusters either by a divisive method (splitting) or agglomerative method (merging).

K-means is simple and efficient for spherical clusters but requires pre-defined cluster numbers. DBSCAN is excellent for irregularly shaped clusters and noise handling but struggles with parameter selection and high dimensions. Hierarchical clustering provides detailed data insight and is great for smaller datasets but is less suitable for large datasets due to its computational demands.

Appendix B: RESULTS

Clustering Between Hierarchical, K-Means & DB Scan, After the Data analysis, we come to Notice that.

1. **Hierarchical Clustering:** This method builds a hierarchy of clusters either agglomerative (merging smaller clusters into larger ones, for this method, we will create a dendrogram to help decide the number of clusters, and then apply the Agglomerative Clustering algorithm.
2. **K-Means Clustering:** This is a centroid-based clustering method that partitions the data into K clusters, where each point belongs to the cluster with the nearest

mean. We will need to decide the number of clusters (K). One common method to determine K is the Elbow Method.

3. **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):** This algorithm groups together points that are closely packed together and marks as outliers the points that lie alone in low-density regions. This algorithm does not require the number of clusters as an input, but it has two main parameters: eps (the maximum distance between two samples for them to be considered as in the same neighborhood) and min_samples (the number of samples in a neighborhood for a point to be considered as a core point)

The process of generating the hierarchical clustering dendrogram took too long, likely due to the size of the dataset. Sample a Subset of the Data: We could use a smaller sample of the dataset for hierarchical clustering. This would make the computation more manageable.

K-Means Observations:

- Cluster Formation: K-Means creates spherical clusters, dividing the data into distinct, non-overlapping groups based on distance to the centroid.
- Race and State Distribution: The clusters in K-Means appear to be more uniformly distributed, which may or may not reflect the underlying relationships in the data. This method assumes that clusters are of similar size and density, which might not be the case for categorical data like "race" and "state."

DBSCAN Observations:

For categorical data like "race" and "state", DBSCAN's ability to handle non-uniform clusters can be advantageous.

- Density-based Clustering: DBSCAN focuses on the density of data points, forming clusters where data points are densely packed and identifying outliers in sparse areas.
- Handling Variability: This method is more adept at handling clusters of varying shapes and sizes, which might be more representative of the real-world distribution of the "race" and "state" data.
- Outlier Detection: DBSCAN can identify outliers, which are data points that do not fit well into any cluster. This could be insightful for understanding anomalies or unique patterns in the data.

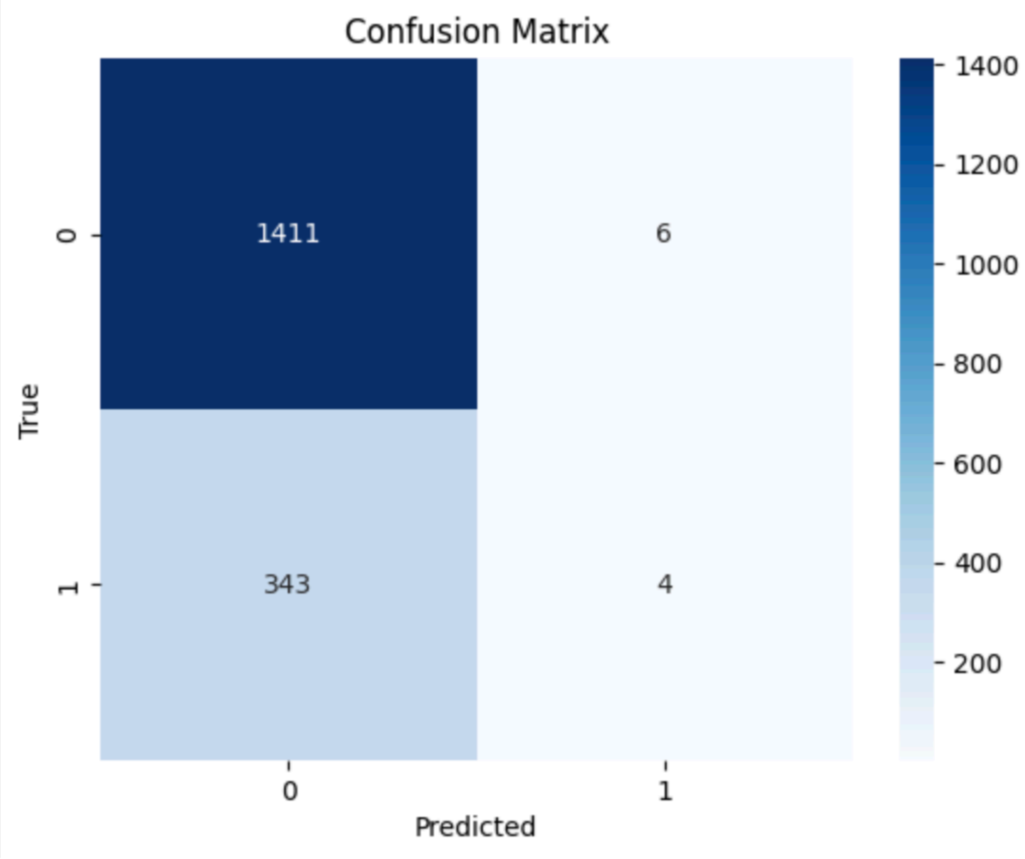
Implication for "Race" and "State”:

For data like "race" and "state", which can have complex and non-uniform distributions, DBSCAN's ability to form clusters of different shapes and sizes allows it to potentially capture more accurate and meaningful groupings. Rigid vs. Flexible Clustering: K-Means might not capture the nuances in the data as effectively, given its tendency to create more uniform and geometrically constrained clusters.

So, to conclude from the above, DBSCAN may offer a more nuanced understanding of the "race" and "state" data due to its flexibility in handling varying cluster densities and shapes, and its ability to identify outliers. However, if a simple and clear segmentation is required, K-Means provides an easy-to-interpret alternative.

Results of Logistic regression:

A logistic regression model was employed to forecast the psychological state of the individual now when they were subjected to gunfire by law enforcement. The outcomes are delineated based on the confusion matrix presented below.



The confusion matrix shows the performance of the model in terms of the number of correct and incorrect predictions. There were a total of 1417 cases where the person was not mentally ill (False). Out of these, the model correctly predicted 1411 as not mentally ill and incorrectly predicted 6 as mentally ill. There were 347 cases where the person was mentally ill (True). The model correctly predicted only 4 of these cases and incorrectly predicted 343 as not mentally ill.

F1-Score: This is a balance between precision and recall. For non-mentally ill cases, the F1-score is high (0.89), but for mentally ill cases, it is very low (0.02), reflecting poor performance in identifying mentally ill cases.

Accuracy: Overall, the model correctly predicted 80% of the cases. However, this high accuracy is largely due to its ability to identify non-mentally ill cases, which are more numerous in the dataset.

Data Cleaning:

The dataset of Police shootings has many Missing values and Different Data object types. So, It's Important to clean the data according to our requirements, so these are the Steps we followed to Data cleaning.

1. **Examine Missing Values:**

- Check for missing values in your dataset. If there are missing values, your data may not be fully cleaned.
  After examining a dataset using pandas in Python, it was found that certain columns have missing values. Specifically, there are 454 missing entries in 'name', 211 in 'armed', 503 in 'age', 31 in 'gender', 1517 in 'race', 966 in 'flee', and 840 each in 'longitude' and 'latitude'. Other columns like 'id', 'date', 'manner_of_death', 'city', 'state', 'signs_of_mental_illness', 'threat_level', 'body_camera', and 'is_geocoding_exact' have no missing values.

2. **Data Consistency & Data Duplicates:**

- Look for inconsistencies or outliers in your data, such as values that don't make sense in the context of your dataset and Check for duplicate records in your data. Duplicates can be a sign of incomplete cleaning.

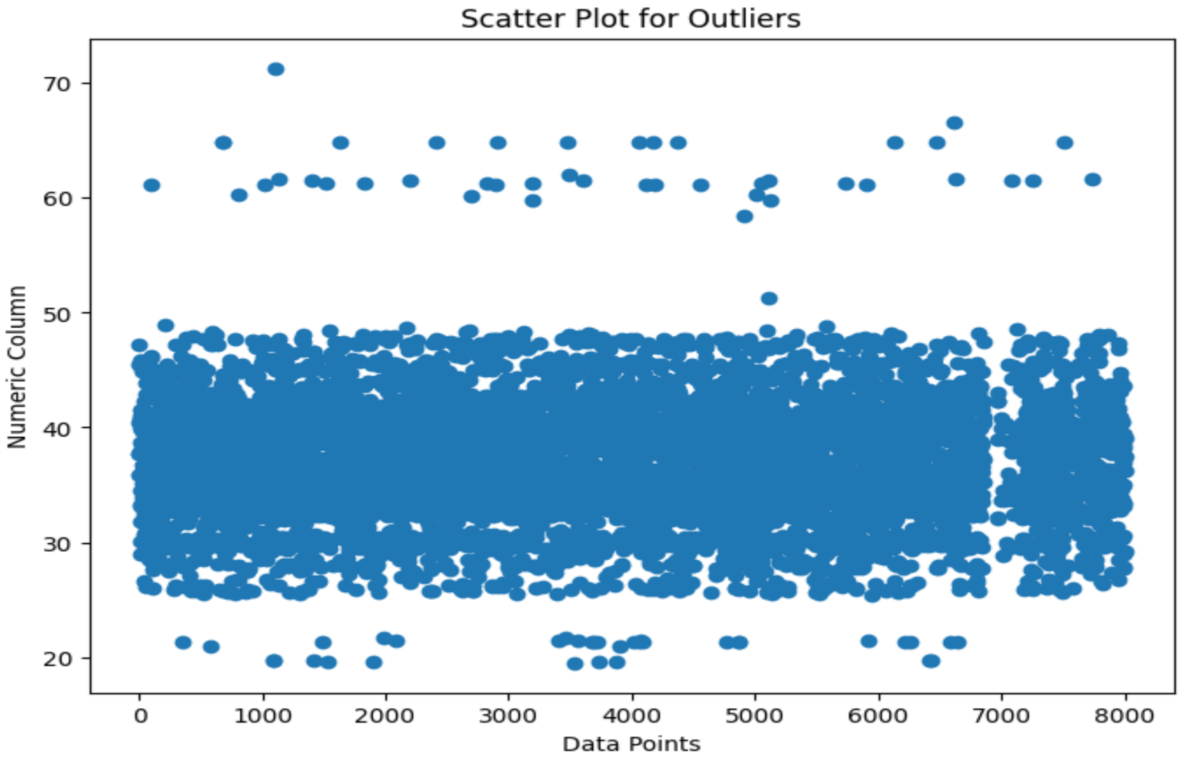
3. **Data Descriptive:**

- Calculate summary statistics (mean, median, standard deviation) and visualize distributions to identify anomalies.

```
id         age    longitude     latitude
count  8002.000000  7499.000000  7162.000000  7162.000000
mean   4415.429643    37.209228   -97.040644    36.675719
std    2497.153259    12.979490    16.524975     5.379965
min       3.000000     2.000000  -160.007000    19.498000
25%    2240.250000    27.000000  -112.028250    33.480000
50%    4445.500000    35.000000   -94.315000    36.105000
75%    6579.750000    45.000000   -83.151500    40.026750
max    8696.000000    92.000000   -67.867000    71.301000
```

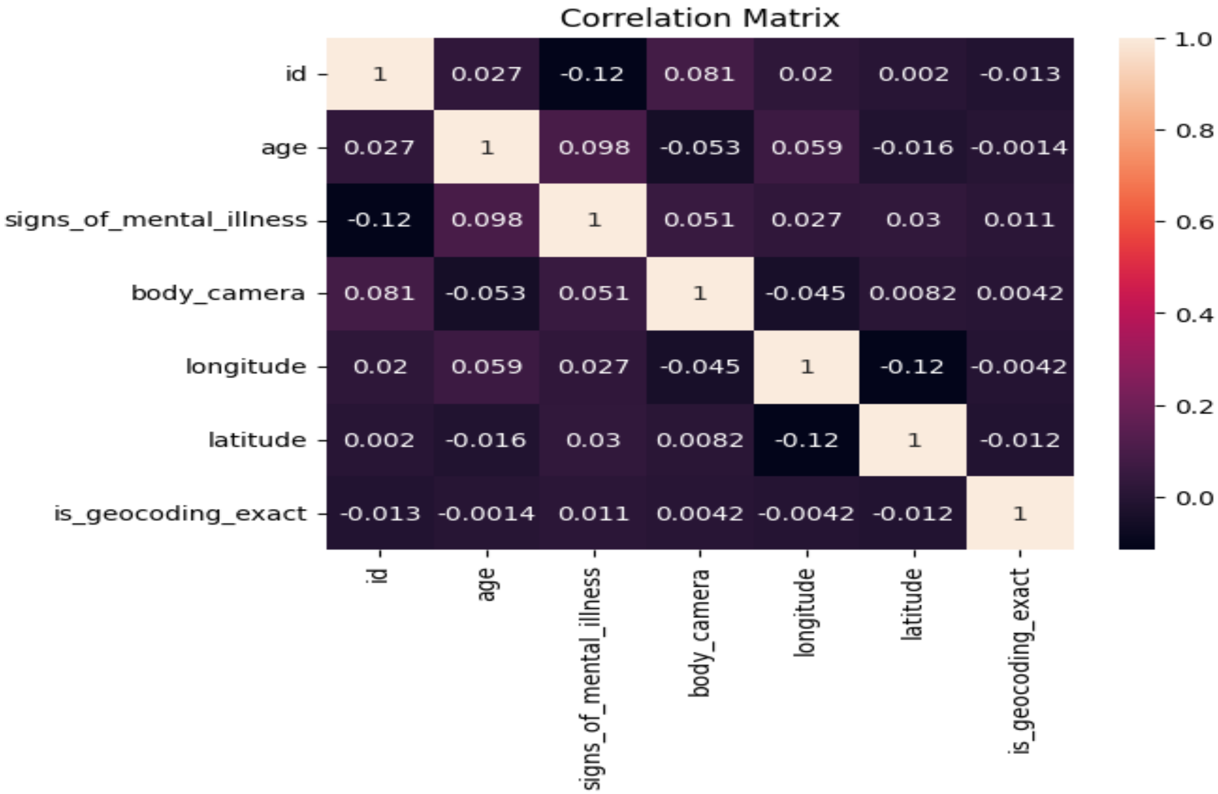
4. **Outliers:**

- Identify and handle outliers that may affect the integrity of your data.
  Outliers appear to be the data points that fall significantly above the main concentration of data, specifically those above the upper threshold of the main data band between the numeric values of 50 and 70 on the Y-axis. Identifying these outliers is crucial for statistical analyses, as they can significantly affect the results and interpretations of the data.



5. Correlation Matrix

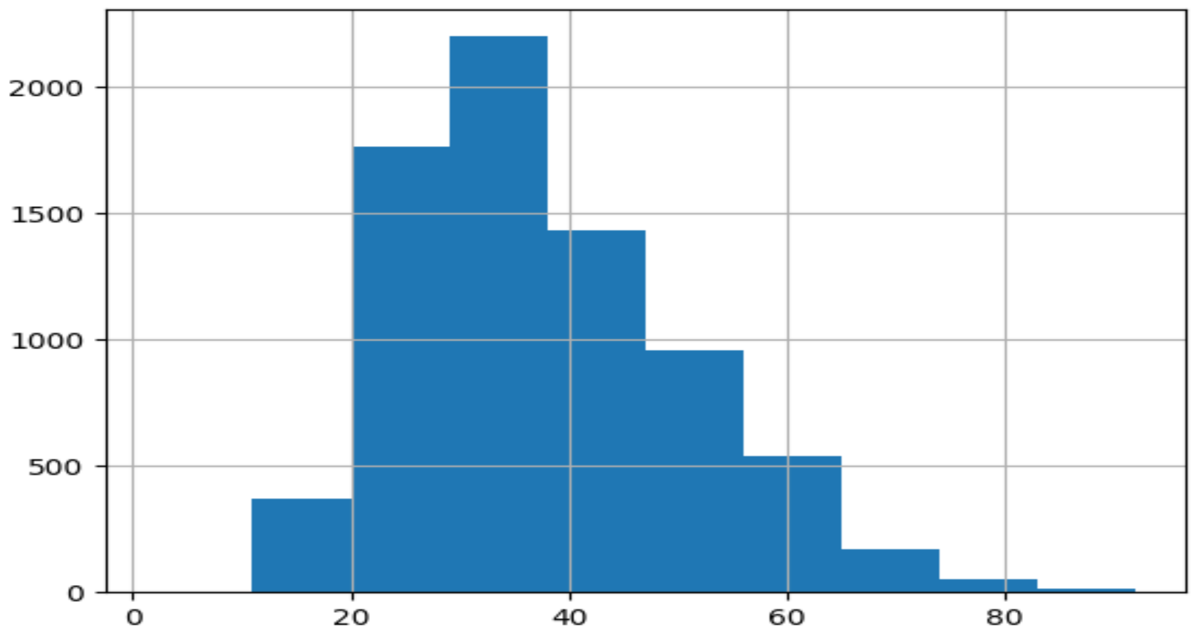
The image shows a table called a "Correlation Matrix." It has a bunch of squares with different colors and numbers that tell us how much two things are related in a group of data. Each row and column have names like "age," "signs_of_mental_illness," and "body_camera." If the number is close to 1 or -1, it means they have a strong connection. If it's close to 0, they don't really have much to do with each other. The colors go from dark purple (low connection) to dark red (high connection). This table helps people see which things might affect each other.



6. Data Exploration:

- Perform exploratory data analysis (EDA) to gain insights and verify that the data behaves as expected. The highest frequency of individuals in the 20 to 40 age range, indicating this is the most common age group. The frequency declines with increasing age, with fewer individuals in the 40 to 60 range, and even fewer from 60 to 80. The lowest frequency is observed in the youngest age group, from 0 to 20. This suggests a dataset where younger adults are the majority, and there is a gradual decrease in population as age increases.

The Below Image we can see the Histogram of Age distribution:

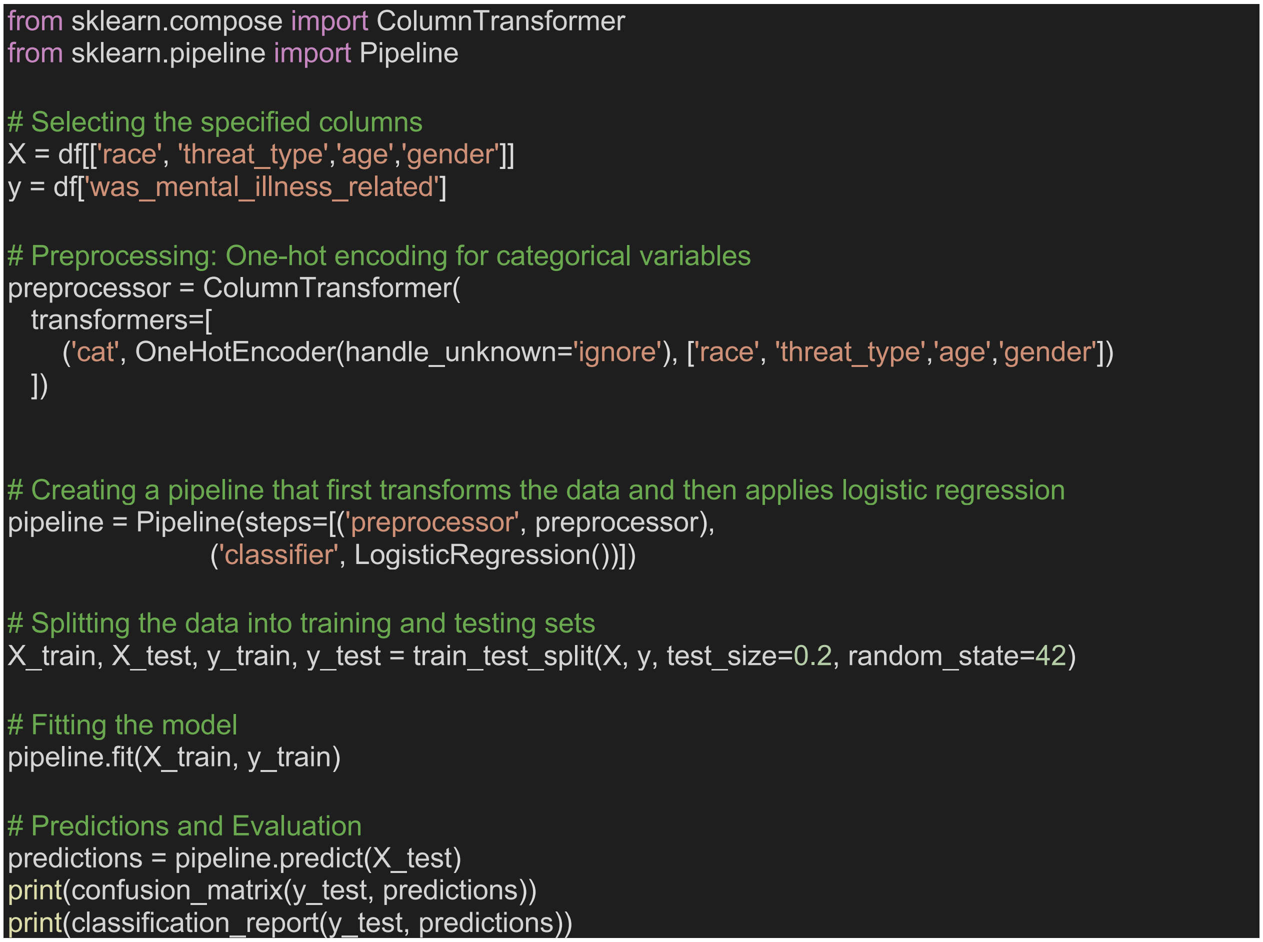


Appendix C: DATA AND CODE

The Washington Post maintains a comprehensive database that records every instance of a fatal shooting by on-duty police officers in the United States since the beginning of 2015. Starting in that year, The Post diligently gathered a wide array of details for each incident, including the race of the individual killed, the context of the shooting, whether the person was armed, and if they were undergoing a mental health crisis, among others. This information is compiled from various sources such as local news, police reports, social media, and oversight databases like Killed by Police and Fatal Encounters. The database is kept current with ongoing reports of shootings and new information that comes to light about each case.

Code for Logistic Regression

```
##Implement logistic regression to predict people shot were mentally-ill based on
race,threat_type,age and gender
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.preprocessing import OneHotEncoder
```



```
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline

# Selecting the specified columns
X = df[['race', 'threat_type','age','gender']]
y = df['was_mental_illness_related']

# Preprocessing: One-hot encoding for categorical variables
preprocessor = ColumnTransformer(
  transformers=[
     ('cat', OneHotEncoder(handle_unknown='ignore'), ['race', 'threat_type','age','gender'])
  ])


# Creating a pipeline that first transforms the data and then applies logistic regression
pipeline = Pipeline(steps=[('preprocessor', preprocessor),
                           ('classifier', LogisticRegression())])

# Splitting the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Fitting the model
pipeline.fit(X_train, y_train)

# Predictions and Evaluation
predictions = pipeline.predict(X_test)
print(confusion_matrix(y_test, predictions))
print(classification_report(y_test, predictions))
```

Clustering:

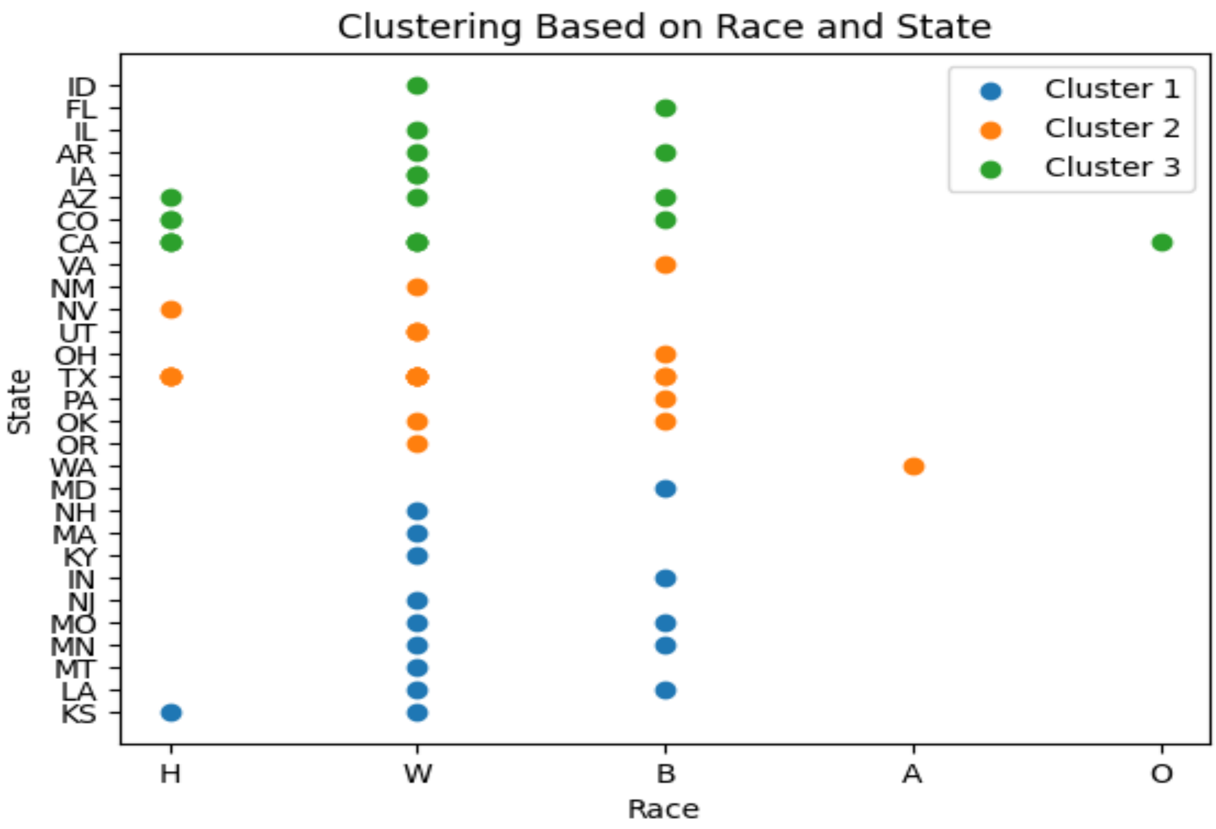
K-Means Clustering

The race categories (H, W, B, A, O) are plotted on the x-axis, and the states (with various abbreviations like ID, FL, IL) are on the y-axis.

- Cluster 1 (blue points): This cluster groups certain states with specific races that are similar to each other according to the clustering criteria. For example, it might represent states where a particular race (H) is more prevalent or has certain characteristics in common.
- Cluster 2 (orange points): Similarly, this cluster signifies another group of states associated with different races (W, B).
- Cluster 3 (green points): This cluster is again a different grouping of states and races (A, O), showing yet another pattern.

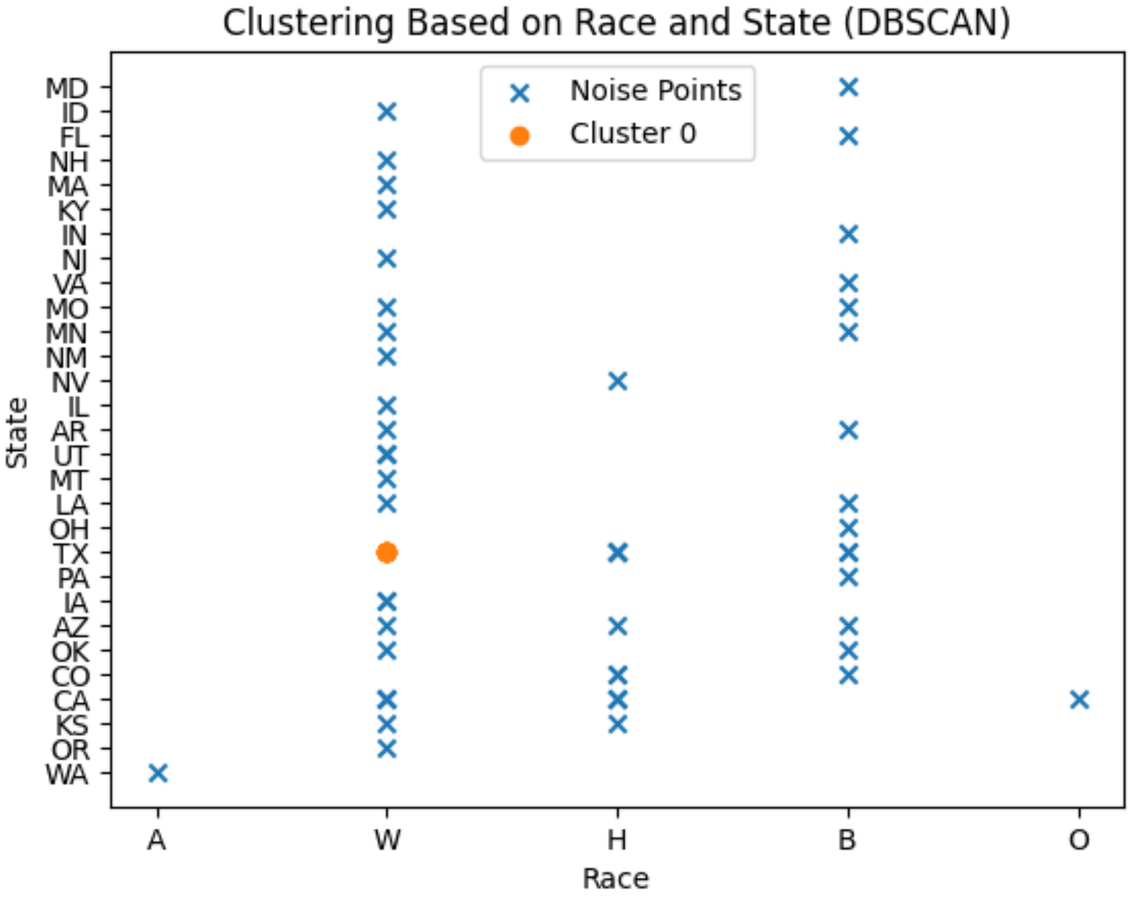
Code:

```
# Kmeans clustering
# Encode categorical variables
label_encoder_race = LabelEncoder()
label_encoder_state = LabelEncoder()
data['Race'] = label_encoder_race.fit_transform(data['Race'])
data['State'] = label_encoder_state.fit_transform(data['State'])

# Prepare data for hierarchical clustering
X = data[['Race', 'State']].values

# Perform hierarchical clustering
linkage_matrix = linkage(X, method='ward')
n_clusters = 3  # Adjust the number of clusters as needed
hierarchical = AgglomerativeClustering(n_clusters=n_clusters, affinity='euclidean', linkage='ward')
data['Cluster'] = hierarchical.fit_predict(X)

# Inverse transform the label encoding for visualization
data['Race'] = label_encoder_race.inverse_transform(data['Race'])
data['State'] = label_encoder_state.inverse_transform(data['State'])

# Visualize the clusters using a dendrogram (optional)
plt.figure(figsize=(12, 6))
dendrogram(linkage_matrix)
plt.title('Dendrogram for Hierarchical Clustering')
plt.show()

# Visualize the clusters
for i in range(n_clusters):
    cluster_data = data[data['Cluster'] == i]
    plt.scatter(cluster_data['Race'], cluster_data['State'], label=f'Cluster {i + 1}')

plt.xlabel('Race')
plt.ylabel('State')
plt.title('Clustering Based on Race and State (Hierarchical Clustering)')
plt.legend()
plt.show()
```

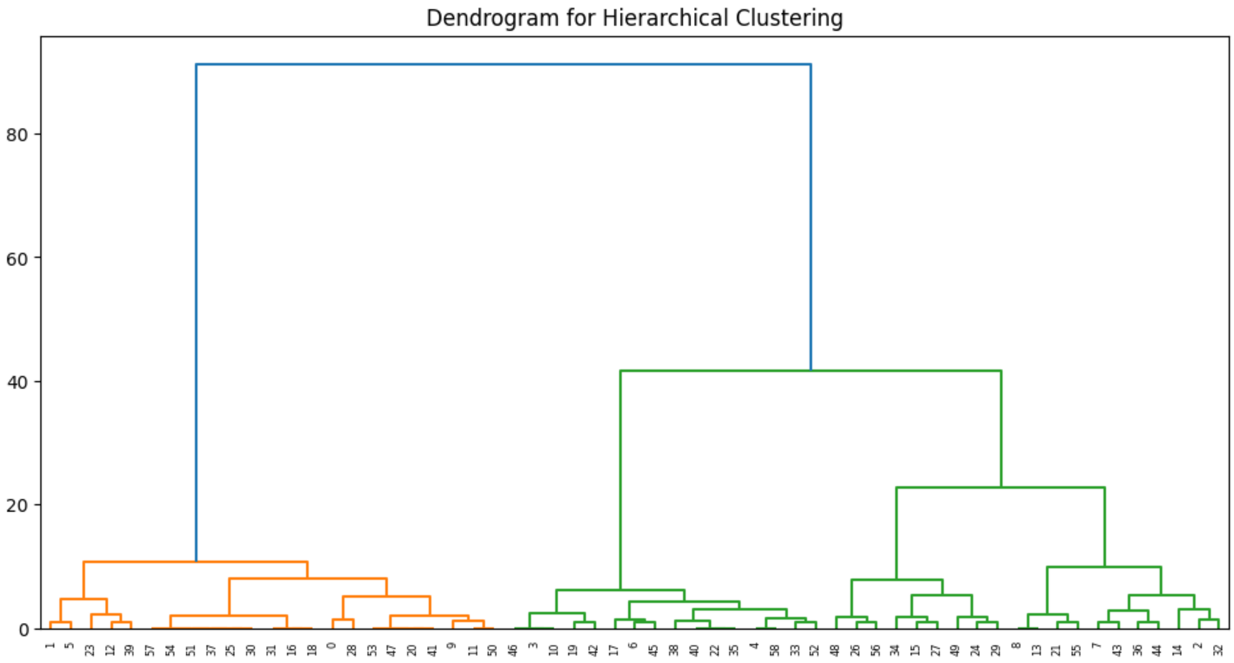


DB SCAN Clustering:

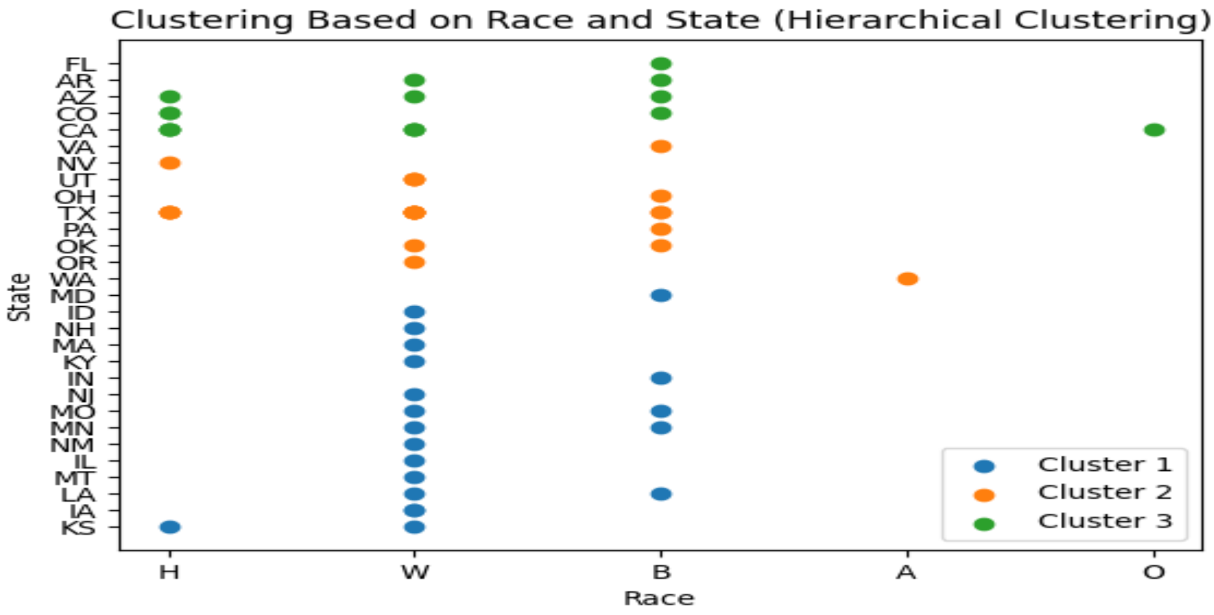
```
#DBSCAN clustering
# Encode categorical variables
label_encoder_race = LabelEncoder()
label_encoder_state = LabelEncoder()
data['Race'] = label_encoder_race.fit_transform(data['Race'])
data['State'] = label_encoder_state.fit_transform(data['State'])

# Prepare data for clustering
X = data[['Race', 'State']].values

# Perform DBSCAN clustering
dbscan = DBSCAN(eps=0.3, min_samples=5)  # Adjust the parameters as needed
data['Cluster'] = dbscan.fit_predict(X)

# Inverse transform the label encoding for visualization
data['Race'] = label_encoder_race.inverse_transform(data['Race'])
data['State'] = label_encoder_state.inverse_transform(data['State'])

# Visualize the clusters
unique_clusters = data['Cluster'].unique()
for i in unique_clusters:
    if i == -1:  # Noise points
        cluster_data = data[data['Cluster'] == i]
        plt.scatter(cluster_data['Race'], cluster_data['State'], label='Noise Points', marker='x')
    else:
        cluster_data = data[data['Cluster'] == i]
        plt.scatter(cluster_data['Race'], cluster_data['State'], label=f'Cluster {i}')

plt.xlabel('Race')
plt.ylabel('State')
plt.title('Clustering Based on Race and State (DBSCAN)')
plt.legend()
plt.show()
```



<u>Hierarchical Clustering: -</u>



Dendrogram for Hierarchical Clustering



Clustering Based on Race and State (Hierarchical Clustering)

REFERENCES:

All images attached here can found in colab file.

1. https://mth522.wordpress.com/about/12-extra-topic-material/
2. https://www.dropbox.com/scl/fi/xhhajad4rn8xxt4gl6sog/Age-race-data-police-shootings.nb?rlkey=dc1fe8ldryv0x6z28fen7xdle&dl=0
3. https://medium.com/swlh/what-is-clustering-and-common-clustering-algorithms-94d2b289df06
4. https://github.com/pradeepbolleddu15/Project2

CONTRIBUTION

Everyone participated in this group activity equally. We equally divided the work for writing. report or in implementation of code by the help of Google collab, google doc and google meet.