

Table of Contents

Contents

[CHAPTER ONE 3](#_Toc198465472)

[1. INTRODUCTION 3](#_Toc198465473)

[1.1 Background 3](#_Toc198465474)

[1.1.1 ABOUT GTD 3](#_Toc198465475)

[1.2 Problem Statement 4](#_Toc198465476)

[1.3 Objectives 5](#_Toc198465477)

[1.3.1 General Objective 5](#_Toc198465478)

[1.3.2 Specific Objectives 5](#_Toc198465479)

[1.4 Contribution of the Research 6](#_Toc198465480)

[1.5 Methodology 7](#_Toc198465481)

[1.5.1 Data Preprocessing 7](#_Toc198465482)

[1.5.2 Exploratory Data Analysis (EDA) 7](#_Toc198465483)

[1.5.3 Predictive Modeling 8](#_Toc198465484)

[1.5.4 Model Evaluation 8](#_Toc198465485)

[CHAPTER TWO 9](#_Toc198465486)

[2.Data Preprocessing 9](#_Toc198465487)

[2.1 Initial Data Inspection and Column Reduction 9](#_Toc198465488)

[2.2 Handling Missing Values 10](#_Toc198465489)

[2.3 Renaming Columns and Removing Duplicate Records 12](#_Toc198465490)

[2.4 Creating an Interactive Search System for Dataset Access 13](#_Toc198465491)

[2.4.1 Purpose and Relevance 13](#_Toc198465492)

[2.5 Visualization of Terrorism Data 14](#_Toc198465493)

[CHAPTER THREE 16](#_Toc198465494)

[3. Model Development and Evaluation 16](#_Toc198465495)

[3.1 Predicting Whether an Attack Was Successful or Not Using Random Forest and SMOTE 16](#_Toc198465496)

[3.2 Predicting High-Risk Region Terrorist Attacks Using Random Forest 17](#_Toc198465497)

[3.2.1 Dataset Overview and Cleaning 17](#_Toc198465498)

[3.2.2 Model Training and Evaluation 18](#_Toc198465499)

[3.3 Association Rule Mining for Attack Types and Target Types 19](#_Toc198465500)

[3.3.1 Data Preparation and Transaction Encoding 19](#_Toc198465501)

[3.3.2 Mining Frequent Itemsets and Generating Rules 19](#_Toc198465502)

[3.3.3 Results: Top Association Rules 20](#_Toc198465503)

[3.3.4 Evaluation Summary 20](#_Toc198465504)

[3.4 Time Series Forecasting of Monthly Terrorism-Related Deaths (2018–2020) 21](#_Toc198465505)

[3.4.1 Data Preparation and Modeling 21](#_Toc198465506)

[3.4.2 Forecasting Procedure 21](#_Toc198465507)

[3.4.3 Forecast Summary 21](#_Toc198465508)

[3.4.4 Model Evaluation on Training Data 21](#_Toc198465509)

[3.4.5 Rationale for Model Selection 22](#_Toc198465510)

[3.4.6 Explanation of Information Criteria 22](#_Toc198465511)

[CHAPTER FOUR 23](#_Toc198465512)

[4. CONCLUSIONS 23](#_Toc198465513)

[4.1 Data Preparation and Preprocessing 23](#_Toc198465514)

[4.2 Exploratory Data Analysis (EDA) 23](#_Toc198465515)

[4.3 Predictive Modeling 24](#_Toc198465516)

[4.4 Contributions and Implications 24](#_Toc198465517)

# CHAPTER ONE

# INTRODUCTION

## Background

The Global Terrorism Database (GTD) is a comprehensive, structured dataset of terrorist attacks worldwide from 1970 to 2017 year. It is an essential resource for data scientists, analysts, and researchers who want to access insights in terrorism trends, attack trends, and determinants of risk. The data set, compiled by the National Consortium for the Study of Terrorism and Responses to Terrorism (START), has over 180,000 events with features such as the location of an event, date, perpetrators, target types, attack methods, and consequences.

From a data science perspective, GTD allows for exploratory data analysis, predictive modeling, and trend analysis. It allows for applications in machine learning, clustering, time-series forecasting, and geospatial analysis. In contrast to most other data sets, GTD includes successful and failed attacks, providing a more complete snapshot of terrorism activity.

But processing the dataset requires careful preprocessing since it contains missing values, inconsistencies in data, and categorical variables that require encoding. Knowledge of these problems is critical for complete analysis. The dataset serves as a foundation for building data- driven counterterror policy, making estimates of global security threats, and developing policy recommendations based on empirical research.

### ABOUT GTD

The Global Terrorism Database (GTD) is a comprehensive, open-source dataset documenting global terrorist incidents from 1970 onward, enabling data-driven analysis of terrorism trends

## Problem Statement

Terrorism is a major global issue, and understanding its patterns is important for many fields. The Global Terrorism Database (GTD) provides a large amount of data on terrorist incidents, but analyzing it effectively is challenging due to missing values, high dimensional, and data complexity.

As a data science student, this study focuses on exploring the GTD data set to uncover meaningful patterns and trends using data analysis techniques.

The goal is not to solve terrorism itself, but to address the difficulties in working with the data set such as handling incomplete data, reducing complexity, and identifying useful insights — to show how data science can help make large, real-world datasets more understandable and usable.

1.2.1 Research Questions and Corresponding Objectives

1. What are the major patterns and trends of global terrorism for the last fifty years?  
 Objective: To perform exploratory data analysis (EDA) and visualization to identify key trends and temporal patterns in global terrorism from 1970 to 2017.

2. How do terrorist attack methods and targets vary by region and over time?  
 Objective: To analyze and compare attack types and target groups across different regions and time periods using grouping, filtering, and visualization techniques.

3. How can statistically modeling and machine learning improve terrorism risk assessment and forecasting?  
 Objective: To build and evaluate machine learning models (e.g., classification, time series forecasting) to predict the number of killed (deaths predicted ) for 3 consequent years 2018-2020

4. What are the preprocessing and analysis challenges of working with the GTD dataset?  
Objective: To identify and document common data quality issues in the GTD (e.g., missing values, inconsistent entries) and demonstrate how preprocessing techniques like data cleaning, imputation, and feature selection can address them.

## Objectives

### General Objective

The overall objective of this study is to apply data science techniques to the Global Terrorism Database (GTD) in order to build predictive models and extract meaningful patterns. This includes analyzing historical trends, identifying key factors influencing terrorist activities, and addressing data challenges such as missing values and high dimensionality. The goal is to demonstrate how data science can be used to better understand complex real-world datasets like the GTD

### Specific Objectives

· To **analyze** historical trends and patterns in terrorist activity by geographic region and time period using exploratory data analysis (EDA) and visualizations.

· To **examine** the relationships between attack methods, target types, and socio-political factors using correlation analysis and cross-tabulation techniques.

· To **evaluate** the impact of missing values and data inconsistencies on model performance using data preprocessing and model comparison.

· To **implement** machine learning and statistical models to estimate terrorism risk and forecast future attacks.

· To **design** an optimized analytical framework for interpreting terrorism data that can support evidence-based insights for potential policy considerations.

## Contribution of the Research

1. The findings of the research have significant implications among different stakeholders including policymakers, security agencies, researchers, and data scientists. Drawing on data analytics from the Global Terrorism Database (GTD), the research contributes to enhanced understanding of terrorism dynamics as well as improved counterterrorism efforts.
2. For security institutions and policymakers, this study provides empirical evidence for strategic planning and allocation of resources in counterterrorism initiatives. Identifying key trends and patterns in terrorist behavior enables authorities to forecast threats and implement proactive security strategies.
3. To data scientists and researchers, this paper offers an organized approach to handling big terrorism data. It discusses the issues and best practices in preprocessing, modeling, and interpreting the data, and forms the foundation for future research into terrorism analytics and risk forecasting.
4. Also, this study supports the use of machine learning and statistical approaches in improving terrorism event prediction models. By addressing data deficiencies and constructing optimized analytical structures, this analysis makes contributions to data science applications in general across security and intelligence.

## Methodology

We use a systematic approach in this study to analyze the Global Terrorism Database (GTD) and uncover notable patterns in terrorism data. The procedure involves data preprocessing, exploratory data analysis, and predictive modeling techniques. The steps below are followed to attain an effective and easy-to-follow workflow.

### Data Preprocessing

The GTD dataset has a huge amount of data with missing values, outliers, and inconsistencies. The first part of our methodology is to preprocess and clean the data by:

* **Handling Missing Data**:  
  Missing values in the Killed and Wounded columns were replaced with 0, as missing often means no casualties were reported. Categorical fields like Target, Motive, and Group were filled with 'Unknown'. Rows missing latitude, longitude, or propvalue were dropped because geographic information is essential. A new Casualties column was created by summing Killed and Wounded.
* **Encoding Categorical Data**:  
  Categorical features were converted to numeric using one-hot encoding. Each unique category was turned into a binary column (e.g., "Bombing", "Shooting", "Hijacking"), allowing the data to be used in machine learning models.

### Exploratory Data Analysis (EDA)

* We conduct exploratory data analysis to continue understanding the structure of the dataset and identify patterns:
* Data Visualization: Basic plots like histograms, bar plots, and scatter plots are made to visualize trends, modes of attack, geographical variations, and other key attributes of the data.

### Predictive Modeling

1.Classification (Attack Success Prediction)

A Random Forest classifier was trained to predict whether a terrorist attack would be successful. We handled class imbalance using **SMOTE** and evaluated the model using **accuracy, confusion matrix**, and **classification report**. The model achieved strong predictive performance, and feature importance was visualized to identify key predictors.

2. Risk Classification (High-Risk Regions)

Another Random Forest model was developed to classify regions as high or low risk. We addressed issues like **constant/ID-like columns** and potential **target leakage**. Model evaluation included **cross-validation**, **OOB score**, and **learning curves**. Feature importance plots revealed the most influential attributes in determining regional risk.

3.Association Rule Mining

We used **Apriori algorithm** to find frequent itemsets and association rules between attack types and target types. The strongest rules were selected based on **lift**, revealing hidden relationships and frequent patterns within the data.

4. Time-Series Forecasting

We used **ARIMA models** to forecast terrorist-related deaths for the next 30 days. The data was aggregated daily, and predictions were visualized alongside actual trends. This helps in understanding temporal dynamics and planning future responses.

### Model Evaluation

After using the above models, we evaluate their performance against metrics like accuracy, precision, recall, and F1-score in the case of classification models. In the case of time-series models, we evaluate forecasting accuracy against Mean Absolute Error (MAE) or Root Mean Square Error (RMSE).

# CHAPTER TWO

# 2.Data Preprocessing

## 2.1 Initial Data Inspection and Column Reduction

The dataset initially consisted of 181,691 rows and 135 columns, which is quite large. Before conducting any analysis or building predictive models, it was crucial to understand the structure of the dataset and identify unnecessary or problematic columns.

Missing Values Overview

* We reviewed all columns to check for missing data. This helped us identify which features were incomplete or had too many null values to be useful. Some columns had no missing values, while others had significant gaps, indicating they were either rarely recorded or inconsistently reported.

Purpose of Dropping Columns

* Based on the inspection, we removed 34 columns for the following reasons:
* Too many missing values: Columns like ransomamt, claimmode2, and weapsubtype2 had a high percentage of missing entries, making them unreliable for analysis.
* Redundant or repetitive information: Features such as alternative\_txt, gname2, and attacktype3 provided alternative or secondary information that overlapped heavily with primary columns.
* Irrelevant for analysis: Fields like approxdate, location, or resolution did not add predictive value or were too vague.
* Low variance or constant values: Some columns had nearly the same value across all rows, contributing little to modeling or pattern detection.
* After dropping these columns, the dataset became cleaner and more focused, allowing us to concentrate on the most informative and complete features. This step was essential to improve data quality and reduce noise before applying machine learning models.

## 2.2 Handling Missing Values

Handling missing values is a crucial step in the preprocessing pipeline, as it directly affects the accuracy and generalizability of any predictive model built upon the dataset. In this project, both categorical and continuous features contained missing values. Therefore, different strategies were applied depending on the nature and purpose of the columns.

Categorical Features: Replacing NaNs with 'Unknown'

The categorical columns with missing values included:

* Target (638 missing values)
* Motive (131,130 missing values)
* Group (0 missing, but included for completeness in preprocessing)

For these columns, missing values were not dropped, as doing so would result in substantial data loss—especially for Motive, which had over 130,000 missing values. Instead, these missing values were replaced with the string 'Unknown'.

Justification:

* The absence of a value does not imply irrelevance; it simply indicates that the information was not reported.
* Replacing with 'Unknown' allows the model to treat this category explicitly during training, preserving the integrity and size of the dataset.

This strategy ensured the retention of as many rows as possible while still capturing the unknown nature of certain records.

Continuous Features: Using Zero to Fill NaNs

The continuous features that had missing values included:

* Killed (10,313 missing values)
* Wounded (16,311 missing values)

In the context of the Global Terrorism Database (GTD), missing values in Killed or Wounded columns typically imply that no casualties were reported—not that the data is unknown.

Thus, all missing values in these columns were replaced with 0.

Justification:

* GTD documentation and data behavior suggest that missing Killed or Wounded values imply zero casualties.
* Imputing 0 avoids misleading the model with artificial assumptions while preserving the records.
* It is consistent with real-world interpretations and allows accurate calculation of total casualties.
* After this replacement, a new feature Casualties was created as the sum of Killed and Wounded.

Dropping Rows with Missing Critical Data (Latitude, Longitude, Property Value)

Some columns, such as latitude, longitude, and propvalue, had missing values:

* Latitude: 4,556 missing values
* Longitude: 4,557 missing values
* Property Value (propvalue): 142,702 missing values

Rows missing these values were dropped.

Justification:

* Latitude and Longitude are essential for any form of spatial analysis, mapping, or geolocation-based modeling. If coordinates are missing, the location of the attack is unknown and cannot be visualized or clustered meaningfully.
* Property Value was dropped if missing because its absence prevents reliable financial impact analysis. Including such rows could distort any model attempting to predict or assess property damage severity.

The decision to drop these rows was based on the fact that retaining them would add noise or nullify the purpose of the related analyses.

* The tailored approach to handling missing values—using 'Unknown' for categorical variables, 0 for casualty-related features, and dropping rows only when the missing data impaired essential analysis—preserved the maximum amount of meaningful data. It also ensured that the dataset remained valid, clean, and representative of real-world conditions, satisfying both academic rigor and model integrity

## 2.3 Renaming Columns and Removing Duplicate Records

As part of the data preprocessing phase, column renaming and duplicate record removal were essential to ensure clarity, consistency, and reliability in the dataset.

Renaming for Improved Readability

Several column names were renamed to make the dataset more interpretable and intuitive for analysis.

For instance:

* iyear, imonth, and iday were renamed to Year, Month, and Day
* nkill and nwound were changed to Killed and Wounded
* attacktype1\_txt, country\_txt, targtype1\_txt, and others were converted to more readable

terms like AttackType, Country, and Target\_type

* These changes made the dataset easier to work with for modeling and visualization, especially when interpreting the results.

Removing Duplicate Records

After cleaning and transforming the dataset, it was necessary to verify whether there were any duplicate rows. Duplicate records can distort statistical analysis and model performance by inflating the significance of specific patterns or features.

Using the complete dataset, duplicate rows across all columns were identified and removed. The system also displayed the exact rows that were considered duplicates. This step ensured that each event in the dataset was unique and that repeated records—often due to merging or entry errors—were eliminated.

* The dataset, now referred to as df cleaned, became a more accurate and streamlined version, free from redundancy and misinterpretation. By renaming columns and removing duplicates, the dataset reached a cleaner state, making it reliable for the next stages of analysis and modeling.

## 2.4 Creating an Interactive Search System for Dataset Access

Given the large size of the Global Terrorism Dataset (GTD), which contains over 170,000 records, it is critical to implement a user-friendly mechanism for efficiently accessing and exploring individual events. For this reason, an interactive search function was developed using Python to allow users to filter and display specific incidents based on Event ID, Country, or Group.

### 2.4.1 Purpose and Relevance

* Improved Accessibility and Navigation:

Manually browsing a 170,000+ row dataset in Excel or raw CSV form is impractical.

The interactive search tool allows users to retrieve specific and relevant rows without scrolling or manually searching.

* Enhanced Usability for Non-Technical Users:

Even users without data science or programming backgrounds can run the script and input simple search criteria (e.g., a country name, terrorist group, or event ID).

The function prompts users with clear choices and guides them through the process of querying the dataset.

* Time Efficiency and Clarity:

By filtering records with simple inputs, analysts can quickly inspect attacks by country (e.g., “Iraq”), attacker group (e.g., “Taliban”), or a unique identifier (eventid).

The filtered output is formatted for readability, showing key information such as date, casualties, attack type, motive, and geolocation.

* Data Exploration and Verification:

During analysis, researchers or developers often need to verify whether a specific record is available and view its complete attributes.

This function provides a structured overview of the selected event, reducing errors and promoting accurate interpretation of results.

* This interactive search tool adds functional value to the project. It transforms the dataset from a static collection of records into a dynamic, queryable resource—making it more practical for academic, professional, and research-oriented use cases. Especially when working with large datasets like GTD, such features enhance data accessibility, integrity, and user experience, which are all essential components of modern data analysis.

## 2.5 Visualization of Terrorism Data

Visualization plays a crucial role in understanding patterns, trends, and relationships within a large dataset. In this project, multiple visual techniques were employed to analyze and interpret the Global Terrorism Dataset of over 170,000 records.

A. Casualties Over Time

A line plot was used to display the total number of casualties per year. This visualization revealed that terrorism-related casualties increased significantly after 2010, peaking in 2014. This spike indicates a period of intensified global terrorist activities, which aligns with known geopolitical events in regions such as the Middle East and Africa.

📈 Line Plot: Total Casualties per Year

This chart provides:

A time-based trend in casualties,Evidence of how terrorist violence evolved,

A sharp increase post-2010, suggesting changes in terrorist operations or data reporting.

B. Countries with the Highest Casualties

A horizontal bar chart was created to show the top 15 countries with the highest number of casualties. The results clearly show that Iraq experienced the most casualties, followed by countries like Afghanistan and Pakistan. These findings highlight geopolitical instability in specific regions.

📊 Bar Chart: Top 15 Countries by Total Casualties

This helps to:

Compare country-level casualty impact,Identify geographic hotspots for attacks,

Visualize the disproportionate impact on certain nations.

C. Most Common Attack Types

A bar chart of attack types was plotted based on frequency. The most frequent attack method was “Bombing/Explosion”, followed by “Armed Assault” and “Assassination”. This insight is important for understanding the operational preferences of terrorist groups.

📊 Bar Chart: Most Common Attack Types

It shows:

The dominance of explosive attacks as a method,Differences in the tactics used by different groups or regions.

D. Mapping Terrorist Attacks by Location (Geospatial Map)

An interactive map was developed using folium to visualize attacks by geographic coordinates. Only the top 100 events were used for clarity. The map uses circles with color coding:

Red: indicates incidents with casualties > 0

Other colors represent different attack types:

Gray = Bombing/Explosion

Yellow = Armed Assault

Blue = Assassination

🗺️ Interactive Map of Top 100 Attacks

The map allows users to:

Visually analyze the location-based spread of attacksIdentify areas of concentrated violence,

Explore individual events by clicking on map points to see country, type of attack, and casualties.

* Summary of Key Insights from Visualization
* Country with the most attacks: Iraq
* Region with the most attacks: Middle East & North Africa
* Year with the most attacks: 2014
* Month with the most attacks: May
* Group with the most attacks: Taliban
* Most common attack type: Bombing*/Explosion*

# CHAPTER THREE

# 3. Model Development and Evaluation

## 3.1 Predicting Whether an Attack Was Successful or Not Using Random Forest and SMOTE

In this section, we focus on predicting whether a terrorist attack was successful or not using a machine learning model. A Random Forest Classifier was used as the core model, and the issue of class imbalance was addressed using SMOTE (Synthetic Minority Over-sampling Technique).

**Why SMOTE Was Used?**

The dataset exhibited a significant class imbalance in the success variable, with many more records of successful attacks than unsuccessful ones. This imbalance can negatively impact the model’s ability to learn patterns of the minority class, often causing it to predict most attacks as successful regardless of input features.

To address this, SMOTE was applied. SMOTE generates synthetic examples of the minority class (unsuccessful attacks) based on feature-space similarities between existing minority instances. Unlike simple duplication, this method increases the diversity of the minority class and improves the model's performance in recognizing it. SMOTE was only applied to the training data to avoid data leakage.

**Model Performance Summary**

* **Accuracy**: 98.39%
* **Confusion Matrix**:

[ 182 72] → Class 0: Unsuccessful

[ 51 7341] → Class 1: Successful

**Classification Report**:

* **Class 0 (Unsuccessful Attacks)**:
  + Precision: 78%
  + Recall: 72%
  + F1-Score: 75%
* **Class 1 (Successful Attacks)**:
  + Precision: 99%
  + Recall: 99%
  + F1-Score: 99%

**Interpretation of Results**

The model achieved very high performance, especially in detecting successful attacks. With a **recall of 99%** for the successful class, very few successful attacks were missed. For the unsuccessful class, performance is lower but still meaningful, with a recall of 72%, indicating room for improvement.

This difference is expected in imbalanced datasets, even after applying SMOTE. The overall **accuracy of 98.4%** demonstrates that the model is reliable in identifying both successful and unsuccessful attacks, with strong generalization capabilities

**Important Features Influencing Success Prediction**

After training, the Random Forest model provides **feature importance scores** that show which factors were most influential in determining whether an attack was successful. The top features included:

* Total number of casualties
* Attack type
* Target type
* Region
* Terrorist group involved

A horizontal bar chart was used to visualize the **top 10 features**, helping interpret the model's decision-making process.

## 3.2 Predicting High-Risk Region Terrorist Attacks Using Random Forest

In this section, we focus on identifying and predicting high-risk regions based on patterns of terrorism activity in the GTD dataset. First, we labeled each region as **high-risk or low-risk** by calculating the number of recorded attacks. Regions with **more than 500 incidents** were considered high-risk and assigned a value of 1 in a new binary column called high\_risk, while others were labeled 0. This region-based risk classification served as the **target variable** for training a predictive model.

To handle the highly **imbalanced dataset**, we implemented a **Random Forest classifier** with built-in class balancing. The model was trained to distinguish high-risk from low-risk regions using relevant features from the dataset. We evaluated performance using accuracy, confusion matrices, and cross-validation scores. Additionally, we visualized **feature importances** to understand which variables contributed most to regional risk classification, and plotted a **learning curve** to assess model generalization. This approach demonstrated how data-driven methods can effectively support geographic risk assessment in counterterrorism analysis.

### 3.2.1 Dataset Overview and Cleaning

The original dataset shape is **(38,229 rows × 31 columns)**. The target variable high\_risk is heavily imbalanced:

* Class 1 (high risk): 38,030 instances
* Class 0 (not high risk): 199 instances

To ensure model generalizability, we removed constant columns and ID-like columns (those with near-unique values). In this case:

* Constant columns: none
* ID-like columns: none

We then checked for **target leakage** by computing the correlation of all features with the target. A feature with a correlation greater than 0.9 was removed if found. This step ensures that the model does not rely on features that are highly predictive due to data leakage.

### 3.2.2 Model Training and Evaluation

We split the data into training and test sets (80/20 split) using stratified sampling to maintain class distribution. We trained a **Random Forest Classifier** with the following hyperparameters:

**Hyperparameter Settings**

* **n\_estimators = 100**  
  *Number of decision trees in the forest.*  
  Chosen to provide strong ensemble learning while maintaining manageable computation time.
* **max\_depth = 10**  
  *Maximum depth each tree can grow.*  
  Helps control overfitting by limiting how deep trees can go.
* **min\_samples\_split = 5**  
  *Minimum number of samples required to split an internal node.*  
  Avoids overly specific trees and improves generalization.
* **min\_samples\_leaf = 2**  
  *Minimum number of samples required to be at a leaf node.*  
  Ensures leaf nodes contain enough samples to prevent noise fitting.
* **class\_weight = 'balanced'**  
  *Automatically adjusts weights inversely proportional to class frequency.*  
  Essential due to extreme class imbalance (only ~0.5% are class 0). It helps the model avoid always predicting the majority class.
* **oob\_score = True**  
  *Uses Out-Of-Bag samples for internal validation.*  
  Provides an unbiased estimate of model accuracy without a separate validation set.

**Model Performance**

The trained model produced the following results:

* **Training Accuracy:** 1.0000
* **Test Accuracy:** 0.9999
* **OOB Score:** 0.9998

**Confusion Matrix**:

Predicted

0 1

Actual 0 39 1

1 0 7606

**Classification Report**:

precision recall f1-score support

Class 0 1.00 0.97 0.99 40

Class 1 1.00 1.00 1.00 7606

Overall Accuracy: 1.00

Macro Average: 0.99

Weighted Average: 1.00

Despite the class imbalance, the model performs remarkably well across all metrics, showing **high recall for both classes**, especially the minority class (0), which is typically hard to detect.

**Cross-Validation Results**

To validate generalization further, 5-fold cross-validation was conducted using **balanced accuracy** as the scoring metric:

* **Cross-validation scores:** [1.0000, 0.9500, 0.9500, 1.0000, 0.9872]
* **Mean CV Score:** 0.9774

These results confirm that the model is consistently accurate across different data splits.

### 3.2.3 Learning Curve Analysis

A learning curve was plotted to observe model performance as a function of training data size. The plot revealed the following:

1. The **training score** remained constant and near 1.0 regardless of data size.  
   *This indicates the model learns the training data perfectly and consistently.*
2. The **validation score** started lower but increased with more training data.  
   *This demonstrates that the model benefits from more data and generalizes well.*

This trend suggests that while the model was slightly data-hungry at smaller sizes, it stabilized quickly and maintained excellent performance.

**3.2.4 Feature Importance Analysis**

Feature importance was calculated to identify the most predictive features. The top features were visualized using a horizontal bar chart, which showed that a small number of variables contribute significantly to the prediction of high risk. These insights can be useful for domain experts and decision-makers.

* This Random Forest model, properly configured and balanced, successfully distinguishes between high-risk and non-high-risk cases with excellent performance. Class balancing via class\_weight, careful parameter tuning, and validation through OOB score and cross-validation have ensured robust and reliable results.

## 3.3 Association Rule Mining for Attack Types and Target Types

In this section, association rule mining is applied to identify significant relationships between types of attacks (AttackType) and their corresponding target types (Target\_type). This technique helps uncover hidden patterns and co-occurrences in the data that may be useful for strategic planning, threat assessment, and resource allocation.

### 3.3.1 Data Preparation and Transaction Encoding

We first extracted the two relevant columns AttackType and Target\_type and removed any missing values to ensure clean input data. Each row, representing an event, was converted into a transaction consisting of the attack type and target type pair.

The transactions were then transformed into a one-hot encoded boolean matrix using the TransactionEncoder from the mlxtend library, enabling the application of frequent itemset mining algorithms.

### 3.3.2 Mining Frequent Itemsets and Generating Rules

The **Apriori algorithm** was applied to identify frequent itemsets with a minimum support threshold of 1%. Subsequently, association rules were generated using the metric of **lift** with a minimum threshold of 1.0 to ensure only positively associated itemsets were retained.

To avoid redundancy, reversed duplicate rules (where antecedents and consequents are swapped) were removed, resulting in a final set of unique association rules.

### 3.3.3 Results: Top Association Rules

| **Antecedents** | **Consequents** | **Support** | **Confidence** | **Lift** | **Interpretation / Suggestion** |
| --- | --- | --- | --- | --- | --- |
| Government (General) | Assassination | 0.0220 | 0.2027 | 4.105 | Strengthen protection of government officials; increase surveillance to avoid assassinations. |
| Facility/Infrastructure Attack | Business | 0.0333 | 0.2582 | 1.965 | Secure critical infrastructure to protect business operations. |
| Bombing/Explosion | Utilities | 0.0797 | 0.1333 | 1.569 | Maintain and protect utilities from bomb-related disruptions. |
| Armed Assault | Police | 0.0345 | 0.2025 | 1.491 | Enhance police safety measures to ensure law enforcement effectiveness. |
| Armed Assault | Transportation | 0.0122 | 0.0720 | 1.339 | Protect transportation systems to prevent disruption of mobility. |
| Armed Assault | Private Citizens & Property | 0.0472 | 0.2772 | 1.321 | Increase community vigilance to protect civilians and property. |
| Armed Assault | Military | 0.0293 | 0.1724 | 1.307 | Keep military personnel and assets secure to maintain defense. |
| Facility/Infrastructure Attack | Government (General) | 0.0155 | 0.1200 | 1.106 | Guard government infrastructure to prevent destabilization. |
| Military | Bombing/Explosion | 0.0863 | 0.6544 | 1.095 | Protect military facilities from bombings to maintain readiness. |
| Bombing/Explosion | Government (Diplomatic) | 0.0103 | 0.0172 | 1.023 | Secure diplomatic locations to avoid escalations. |

### 3.3.4 Evaluation Summary

* **Total rules:** 11
* **Avg. support:** 3.52%
* **Avg. confidence:** 24.63%
* **Avg. lift:** 1.575
* **Max lift:** 4.105 (Government → Assassination)
* **Min lift:** 1.0036

Some rules show strong lift, indicating real and useful patterns—not just chance.

* The rules reveal key links between attack types and targets. For example, government attacks are often linked to assassinations, and infrastructure attacks target businesses. These insights can help decision-makers strengthen protection where it's most needed.

## 3.4 Time Series Forecasting of Monthly Terrorism-Related Deaths (2018–2020)

This section presents the development, evaluation, and forecasting results of a time series model applied to terrorism-related monthly death data, projecting future fatalities for the three-year period from January 2018 through December 2020.

### 3.4.1 Data Preparation and Modeling

* A new datetime column was constructed by combining the Year, Month, and Day variables, enabling proper temporal indexing of the dataset.
* Monthly aggregation of total deaths was performed to produce a consistent time series representing the number of fatalities per month.
* The Seasonal Autoregressive Integrated Moving Average (**SARIMA**) model was selected due to its ability to capture both non-stationary behavior and seasonal patterns commonly observed in monthly terrorism death data.
* The SARIMA model was trained on historical monthly fatalities to learn underlying temporal dependencies and seasonal trends.

### 3.4.2 Forecasting Procedure

* The trained SARIMA model generated **monthly forecasts for a 36-month horizon**, spanning January 2018 to December 2020.
* These forecasts represent the predicted number of deaths for each month within the period.
* To provide summary insights, yearly totals were computed by summing the monthly forecasts within each calendar year.

### 3.4.3 Forecast Summary

| **Metric** | **Value** |
| --- | --- |
| Total predicted deaths (2018–2020) | 28,458 |
| Average monthly deaths (2018–2020) | 790 |
| Predicted deaths in 2018 | 9,163 |
| Predicted deaths in 2019 | 9,486 |
| Predicted deaths in 2020 | 9,808 |

### 3.4.4 Model Evaluation on Training Data

| **Metric** | **Value** |
| --- | --- |
| Mean Absolute Error (MAE) | 62.55 |
| Root Mean Square Error (RMSE) | 142.63 |
| Akaike Information Criterion (AIC) | 6320.52 |
| Bayesian Information Criterion (BIC) | 6341.51 |

### 3.4.5 Rationale for Model Selection

* The **SARIMA model** was chosen for its effectiveness in modeling time series data with both trends and seasonal fluctuations, characteristics evident in monthly terrorism-related death counts.
* Accuracy metrics (MAE and RMSE) demonstrate the model’s capability to closely fit historical data while maintaining predictive validity.
* The information criteria (AIC and BIC) were used to ensure an optimal balance between model fit and complexity, minimizing risks of overfitting or underfitting.

### 3.4.6 Explanation of Information Criteria

* **Akaike Information Criterion (AIC):** Quantifies model quality by penalizing complexity relative to goodness of fit; lower values signify preferable models.
* **Bayesian Information Criterion (BIC):** Similar to AIC but imposes a stronger penalty on model complexity, favoring simpler models with good explanatory power.

# CHAPTER FOUR

# 4. CONCLUSIONS

This study has successfully applied a wide range of data science techniques to the **Global Terrorism Database (GTD)**, a comprehensive dataset documenting terrorist incidents worldwide from 1970 to 2017. The **primary objective** was to uncover meaningful patterns and develop predictive models that enhance the understanding of global terrorism dynamics. In doing so, the study also addressed the inherent challenges of working with a large, complex, and often incomplete dataset.

## 4.1 Data Preparation and Preprocessing

A foundational component of this research involved extensive **data preprocessing**. The original dataset comprised **181,691 rows and 135 columns**, but due to issues such as missing values, low variance, and redundancy, **34 columns were removed**. Key numeric columns such as nkill (killed) and nwound (wounded) had missing values, which were replaced with zeros—reflecting the assumption that missing entries likely indicated no reported casualties.

For categorical features like targtype1\_txt (target type) and motive, missing values were imputed with **'Unknown'** to preserve dataset completeness without introducing bias. Records with missing spatial data (latitude and longitude) or property damage estimates were **excluded**. Columns were **renamed for clarity**, **duplicates were removed**, and an **interactive search functionality** was integrated to enhance data accessibility for both technical and non-technical users.

## 4.2 Exploratory Data Analysis (EDA)

Comprehensive **EDA** revealed key patterns in the GTD data:

* A **significant increase in terrorist casualties** occurred post-2010, peaking in **2014**.
* **Iraq** experienced the highest number of casualties, followed by **Afghanistan** and **Pakistan**.
* The **Middle East & North Africa** emerged as the most attack-prone region.
* The **most common attack type** was "Bombing/Explosion", followed by "Armed Assault".
* May was identified as the month with the most attacks, and the **Taliban** as the most active terrorist group.
* These findings contextualize the temporal and geographical patterns of terrorism globally.

## 4.3 Predictive Modeling

A variety of machine learning models were implemented to extract insights and make predictions:

* **Random Forest Classifier (Attack Success)**:  
  Used to predict whether an attack would be successful, achieving **98.39% accuracy** and **99% recall** using **SMOTE** to address class imbalance. Important features included number of casualties, attack type, target type, region, and terrorist group.
* **Random Forest Classifier (Regional Risk Classification)**:  
  Classified global regions as **high-risk** or **low-risk**, with balanced class weighting. The model achieved **near-perfect accuracy and recall**, identifying critical features for geopolitical risk analysis.
* **Association Rule Mining (Apriori Algorithm)**:  
  Uncovered strong relationships between attacktype1\_txt and targtype1\_txt. For instance, **assassinations were strongly linked to government targets**, and **bombings were associated with infrastructure and utility targets**—insights that can support proactive threat assessments.
* **Time Series Forecasting (SARIMA Model)**:  
  Forecasted future terrorist-related deaths from 2018 to 2020. The model predicted a total of **28,458 deaths**, with an increasing trend year-over-year. Evaluation metrics such as **MAE and RMSE** confirmed the model’s validity and accuracy in capturing historical trends.

## 4.4 Contributions and Implications

The results of this study have **significant implications** for various stakeholders:

* For **policy-makers and security agencies**, the findings support data-driven **strategic planning** and **resource allocation** for counter-terrorism initiatives.
* For **data scientists and researchers**, the project demonstrates an effective methodology for **cleaning, modeling, and interpreting** terrorism-related datasets.
* The success of predictive models reinforces the importance of **machine learning and statistical techniques** in developing systems for **early warning and risk forecasting**.

**Final Remarks**

* In conclusion, this research has effectively demonstrated how **systematic data science methodologies** can be applied to complex, real-world challenges like terrorism. By addressing data quality issues, conducting deep exploratory analysis, and deploying advanced machine learning models, this study extracted valuable insights that contribute to our understanding of global security threats. These findings not only serve academic interests but also provide practical value for decision-makers and institutions focused on **enhancing global safety and counter-terrorism strategies**.