UNIVERSITÉ DE NEUCHÂTEL
Institut de statistique

Modelling Terror Attacks in Europe

Binary classification through space & space-time models and threat level classification through hurdle models with overdispersion

Master Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science in Statistics

by

Mauro Schlaepfer

Thesis Director

Dr. Alina Matei
University of Neuchâtel,
Faculty of Science
Institute of Statistics

Neuchâtel, December 2020

Acknowledgement

Please acknowledge that the data used belongs to the Federal Office of Police (FEDPOL) of Switzerland, but any statement or categorization made was done by the researcher and does not reflect the opinion of the federal police. I thank Dr. Alina Matei, professor on the department of statistics at the University of Neuchâtel, for supervising and guiding me through my master thesis. I further thank the head of the 'Commissariat IV Analysis' of FEDPOL, for the opportunity to cooperate with the federal police. I also thank Thomas Fischer and Janine Schläpfer for their general support.

Abstract

Using a unique data set we try to predict islamic, left and right terror attacks in Europe using a neighbourhood matrix with a single parameter, the regression gradient boosting method following James et al. [2000] and a spatio-temporal method featuring a Gaussian spatial kernel with an autoregressive time process of order one loosely following Morris [2016]. We find a strong spatial correlation, but fail to find a spatio-temporal correlation. We further try to quantify the threat of an attack through three hurdle count models with overdispersion; the quasi-Poisson model, a regular negative binomial model and a negative binomial model with a linear and a quadratic mean-variance relationship following Lindén & Mäntyniemi [2011] and Ver Hoef & Boveng [2007]. We compare these models using the Wilcoxon signed-rank test [1945] and a method based on the quantiles, which we develop ourselves. We find that both negative binomial models quantify the threat of a terror attack better than the quasi-Poisson model.

Contents

Chapter 1

Introduction

1.1 Motivation for the Study

According to Europol [2019, p. 4] continues terrorism to be a major threat to the security of the EU states. Hence it is not a surprise that there have been several studies trying to predict terror attacks such as Porter & White [2010] with a hurdle model in Indonesia or Tolan & Soliman [2015] in Egypt using machine learning algorithms. These studies only look at the time dimension in a specific area using open source data from the Global Terrorism Database [2020]. Working together with the federal police of Switzerland (FEDPOL), we got the unique opportunity to create a terror attack dataset being far more complete and extensive than the open source terror data banks around. Using this data we will try to predict left, right and islamic terror attacks in the EU and EFTA states. The attacks, and the threat of these attacks, vary over time and space. This analysis will be split into two parts. The first part handles the time and space dimension of the attacks and tries to classify if an attack happens on a specific date in a specific area. The second part models the threat of any given attack based on covariates.

1.2 Structure of the Study

1. Part: Binary Classification

In section 2.1, 'Binary Classification Models', we model terror attacks through spatial and spatio-temporal models. The goal is to find spatial and temporal patters in past attacks that allow a prediction of future attacks. We split Europe into 6561 grid cells and include 10'348 days from 1992 to 2020 into the analysis. This leads to a high computational effort due to the high dimensional data structure. The data points in this section are not attack but region and time based, therefore we will use only spatial covariates in this section. The models are implemented in chapter 4.

2. Part: Threat Level Classification

This section contains the main part of our analysis. It is based on the threat variable, which contains the amount of killed and the amount of wounded victims per attack. In section 'Threat level Classification Models' we introduce three models well-equipped to model the unique behaviour of that variable with a strong excess of zeros, which stands for no wounded or killed victim in an attack, and some extreme values with over 400 victims. In chapter 5 we estimate the models with different covariates, simulate from them and compare the fit of these models.

3. Part: Data exploration

The descriptive and the exploratory statistics of the part of the data that was used for the analysis can be found in chapter 3. The exploratory analysis of the whole dataset is attached in the appendix under 'Whole Data'. A part of the R-code used in the thesis is given in Appendix C: R-code

Chapter 2

Literature Review

2.1 Binary Classification Models

2.1.1 Spatial Models

We divide the European map into 81·81 grid cells, shown in section 'Spatial covariates' or section 'Implementation of Spatial Models'. That value is chosen arbitrary but leads to convenient 0.5 degree steps of the latitude and the longitude per grid cell. Counting all cells we accumulate 6561 different regions. Some of them are unpopulated sea regions and will not be assessed an attack probability.

Neighbourhood matrix

Lets start with the most trivial method. Let A be a neighbourhood matrix from the $n = 6561$ grids.

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,j} & \cdots & a_{n,n} \end{pmatrix}$$

Grid cells that communicate with each other correlate by ρ. The cells only communicate with their direct neighbour cells. The diagonal entry $a_{i,i}$ is then $1 - \sum_{j=1, j\neq i}^{n} a_{i,j}$. The other entries are zero. Each grid cell has between two and four neighbour cells. For example, grid cell 1, communicates with cell 2 and 82. Let **E** be the relative amount of terror attacks per grid cell:

$$\mathbf{E} = \frac{1}{\sum_{j=1}^{n} e_j} \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{pmatrix}$$

where e_j is the amount of attacks in cell j. The probability vector $\mathbf{P}$ of an attack is then:

$$\mathbf{P} = \mathbf{A} \cdot \mathbf{E} = \begin{pmatrix} 1-2\rho & \rho & \cdots & 0 \\ \rho & 1-4\rho & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1-2\rho \end{pmatrix} \cdot \begin{pmatrix} \text{Relative \# Attacks in cell 1} \\ \text{Relative \# Attacks in cell 2} \\ \vdots \\ \text{Relative \# Attacks in cell 6561} \end{pmatrix}$$

Notable that $\sum_{i=1}^{6561} P_i = 1$. In most neighbourhood matrices is the diagonal element zero and the normalization is done through dividing each row by the row sum [Anselin, 2020]. A drawback of that method is the change of the internal weighting structure of $\mathbf{A}$ [Smith, 2020]. We keep the correlation intact by using non-zero diagonal entries, notable that other methods are possible such as $\frac{1}{\lambda_{max}}\mathbf{A}$, where λ_{max} corresponds to the largest eigenvalue of $\mathbf{A}$ [Smith, 2020]. We prefer this approach due to less computational effort by not computing an eigenvalue decomposition and by giving each cell the opportunity to communicate with itself. Following the principle of parsimony [1]., we only estimate a single parameter in the model.

Regression Gradient Boosting

Spatial classification is often done through statistical learning methods. There are a lot of applicable methods for this dataset. We only use the gradient boosting method, since it's performance is quite good and there is no need to use more complex models such as neural networks since the amount of spatial covariates is small. We are inspired by the approach of Hao et al. [2019] to classify terror attacks with spatial covariates. They divide their spatial covariates into social elements, natural elements and geographical elements in order to predict terror attacks with the random forest method. We use the population density as a social element, the average temperature in January as a natural element and the altitude as a geographical element. These variables are closely described in section 3.3. We use the gradient boosting method over the random forest method since it's performance is generally better [Deléamont, 2020], while having a computational training cost of $\mathcal{O}(npn_{trees})$ compared to $\mathcal{O}(n^2 pn_{trees})$ for the random forest method [Computational complexity of machine learning algorithms, 2018]. The number of covariates is shown as p and the number of trees computed as n_{trees}.

The gradient boosting algorithm combines the outputs of many weak classifiers, here decision trees, to produce a good classifier. The trees are grown sequentially and not parallel as in the ran-

[1] The principle that the best explanation of a phenomenon or event is the simplest, with the least amount of assumptions or changes [Martin & Hine, 2008]

dom forest method. Every new tree uses the information from previously grown trees. [Deléamont, 2020] According to Natekin & Knoll [2013, p. 1] is "the principle idea behind this algorithm to construct the new base-learners to be maximally correlated with the negative gradient of the loss function." We use boosting for regression rather than classification due to the nature of the data. We would lose information by not accounting that certain regions have had multiple attacks. The algorithm for gradient boosting for regression in James et al. [2000] (as cited in Deléamont [2020]) is shown in (2.1). The amount of grid cells is displayed by n, $(y_1,...,y_n)$ are the amount of attacks per grid cell, X are the spatial covariates and $\hat{p}(x)$ represents the estimated attack probability vector for the grid cells. $\hat{f}_b$ stands for the b^{th} fitted tree, which updates the estimated fitted values $\hat{f}^{(b-1)}$. Boosting regression trees generally leads to three tuning parameters:

B : Number of trees, selected by cross-validation

λ : Shrinkage parameter, controls learning rate, positive scalar

d : Number of splits in each tree, d=1 is common and leads to an additive model

[Deléamont, 2020].

Step 1 sets the fitted function $\hat{f}^{(0)}(x)$ to zero and the residuals $\left(r_1^{(0)},...,r_n^{(0)}\right)$ to the target values $(y_1,...,y_n)$, here the amount of attacks per grid cell. Step 2 builds the trees inside a loop, one after another. The first tree is build with the covariates and the target value. The gradient step updates the fitted values and the residuals by the new decision tree multiplied with the learning rate. The next tree in the loop will be build with the updated residuals. Every step explains a part of the remaining variance in the residuals. Step 3 is the boosting step, where the updates are combined to the output values. Step 4 transforms these values to probabilities that sum up to 1.

$$
\begin{aligned}
&\texttt{Step 1: Set } \hat{f}^{(0)}(x) = 0 \texttt{ and } r^{(0)} = \left(r_1^{(0)},...,r_n^{(0)}\right) = (y_1,...,y_n) \\
&\texttt{Step 2: For } b = 1,...,B\texttt{:} \\
&\qquad \texttt{a) Fit a tree } \hat{f}_b \texttt{ with } d \texttt{ splits to } \left(X, r^{(b-1)}\right) \\
&\qquad \texttt{b) Update the fitted function: } \hat{f}^{(b)}(x) = \hat{f}^{(b-1)}(x) + \lambda \hat{f}_b(x) \\
&\qquad \texttt{c) Update the residuals: } r_i^{(b)} = r_i^{(b-1)} - \lambda \hat{f}_b(x_i)\ ,\ i = 1,...,n \\
&\texttt{Step 3: Output } \hat{f}^{(B)}(x) = \sum_{b=1}^{B} \lambda \hat{f}_b(x) \\
&\texttt{Step 4: Normalize } \hat{p}(x) := \hat{f}^{(B)}(x) \div \sum_{i=1}^{n} \hat{f}^{(B)}(x_i)
\end{aligned}
\tag{2.1}
$$

[Deléamont, 2020].

2.1.2 Space-Time Model

We again split Europe into 81·81 grid cells. We model the time period between the 1.1.1992 and 30.4.2020. The starting date is chosen after the fall of the Soviet Union. Let I be the amount of grids, T the amount of days and J the amount of left, right and islamic motivated terror attacks in the dataset, then:

$$I \in \{1,...,6561\},\ T \in \{1,...,10348\},\ J \in \{1,...,776\}.$$

The computational effort of space-time models with these features is high, hence most methods were not applicable. Therefore does the following method use shortcuts that can not be used in many otherwise appealing methods.

Gaussian Spatial Kernel & Autoregressive Time Process

Loosely following Morris [2016] approach to model rare binary events for spatial models, we create our own space-time model to classify attacks through time and space. For the spatial modelling of the attacks we use a Gaussian kernel suggested by Morris [2016]:

$$W_{ij} := \exp\left(-\frac{|\mathbf{s_i} - \mathbf{v_j}|}{\sqrt{2}\rho}\right)^2 \tag{2.2}$$

where $|\mathbf{s_i} - \mathbf{v_j}|$ is the distance between the spatial grids s and the location of the attack v. ρ is the kernel bandwidth that determines the spatial dependency, with large ρ standing for a long-range dependency and vice versa [Morris, 2016]. W is a matrix with dimensions of 6561 × 776. It might be intuitive to see W_j as a spatial probability vector for each attack. The time dimension is split into days with attacks and days without attacks to decrease the computational cost. Let $\sum J_{(t-1)}$ be the amount of attacks on day $t-1$, then the probability for an attack on day t is:

$$P(Y = 1, t = T \mid \rho, \phi, \delta) = \pi_t :=$$

$$\pi_{(t-1)}\phi \qquad \sum J_{(t-1)} = 0 \tag{2.3}$$

$$\pi_{(t-1)}\phi + \exp\left(-\frac{1}{\delta}\right) \sum_{j=\,(t-1)} W_j \qquad \sum (t-1) > 0 \tag{2.4}$$

Where $\phi \in [0,1]$, to guarantee a positive stationary process. Equation (2.3) implies that if no attack happened today, the risk of a terror attack tomorrow is the risk of an attack on the current day multiplied by ϕ. Otherwise the additional risk is added through an exponential weight function, that guarantees the values to be in the desired range, multiplied by the spatial probabilities of the attacks on that day as shown in equation (2.4). $\sum_{j=\,J_{(t-1)}} W_j$ sums up the spatial probabilities of all attacks on day $t-1$. The sum symbol is redundant if only one attack happened on day $t-1$. Then the final probability matrix is:

$$P\left(Y = 1 \mid \rho, \phi, \delta\right) = \pi \in \mathbb{R}^{10348 \times 6561}$$

We have an autoregressive process of order 1 (AR1) for the time dimension and a Gaussian kernel for the spatial dimension.

2.2 Threat level Classification Models

Section 2.1 predicted where and when an attack will happen. Section 2.2 classifies these attacks into different threat levels using the amount of wounded and killed victims per attack. With the aim of achieving this classification we present three models well equipped to model count data. Our dataset contains an excess of zeros, standing for no injured or killed people in the attack. We model the excess of zeros with a hurdle model which according to Porter & White [2010] is better equipped to model terrorism data than a zero-inflated model, which assumes that the extra zeros observed are due to censoring. The hurdle model assumes that the extra zeros are due to a separate hurdle process, which must be overcome before the number of corresponding incidents are determined. This seems more reasonable for terrorism data, since deaths or injuries are rare not because they are unobserved, but because they are indeed rare. Terrorism data also includes some incidents with an extraordinary high amount of harmed victims, hence a Poisson might not be well-equipped to model that kind of data since it assumes the variance to be equal to the mean. We therefore introduce the negative binomial model with two overdispersion parameters to control for a linear and a quadratic mean variance relationship. This model is then compared to a quasi-Poisson model, which allows a linear mean-variance relationship, and a regular negative binomial model with only the quadratic mean-variance relationship.

2.2.1 Poisson Hurdle Model with Overdispersion

Poisson Model

The Poisson distribution is a discrete probability distribution used to model rare count data. Its derivation is shown in equations (1) to (8) in the Appendix A. The probability mass function:

$$P(Y = n) = \frac{\lambda^n \, e^{-\lambda}}{n!} \tag{2.5}$$

corresponds to the Poisson distribution with $\lambda > 0 \quad \& \quad n \in \mathbb{N}_0$.
This distribution possesses only one parameter, λ, which is also equal to the mean and the variance of a Poisson-distributed random variable [Weisstein, 1984].

Poisson Hurdle Model

Welsh et al. [1996, p. 298] show that independent count data with an excess of zeros can be split into:

$$P(Y=0|x) = 1-p(x) \tag{2.6}$$

$$P(Y=n|x) = p(x)\,\frac{\lambda^n}{\left(e^{\lambda}-1\right)\,n!}\,,\quad n\in\mathbb{N}_+ \tag{2.7}$$

Equation (2.6) models the observed zeros, while equation (2.7) is a zero-truncated Poisson distribution multiplied by $p(x)$, the probability of observing a value higher than zero given covariates x.

In terms of expectation and variance the quasi-Poisson model can be seen as a special case of the negative binomial model with two overdispersion parameters, where θ is set to zero. Notable that they are not exactly the same because one uses a quasi-likelihood method to estimate the parameters, while the negative binomial uses a maximum likelihood approach. It is shown in equation (2.15).

2.2.2 Negative Binomial Hurdle Model with Overdispersion

One of the crucial features of the Poisson distribution is that its expectation is equal to its variance. In nature, however, we often find data with a variance larger than the mean [Rodriguez, 2013]. Correcting such overdispersion is often done through models with different assumptions about the mean-variance relationship [Lindén & Mäntyniemi, 2011]. We propose a parameterization of the negative binomial distribution, with two overdispersion parameters and one parameter for the hurdle part to control for an excessive amount of zeros and for quadratic mean–variance relationships.

Negative Binomial Model

The probability mass function of the negative binomial given in Bhaktha [2018, p. 14] is:

$$P(Y=k) = \binom{k+r-1}{k}\,\pi^r(1-\pi)^k\,,\ k\in\mathbb{N}_0 \tag{2.8}$$

where Y is a random variable representing the number of failures before the r^{th} success in a sequence of Bernoulli trials of parameter π.

Negative Binomial Hurdle Model

Applying the concept of the hurdle model in equation (2.6) to the Negative Binomial (NB) model in (2.8) leads to:

$$P(Y=0|x) = 1-p(x) \tag{2.9}$$

$$P(Y=k|x) = p(x)\binom{k+r-1}{k}\frac{\pi^r(1-\pi)^k}{1-\pi^r},\ k\in\mathbb{N}_+ \tag{2.10}$$

Equation (2.9) again only models the observed zeros, while equation (2.10) is a zero-truncated NB distribution, multiplied again by p(x), the probability of observing values higher than zero given covariates x.

Negative Binomial Model with Overdispersion

In equation (2.8) r and π can be written in terms of mean and variance as:

$$r = \frac{\mu^2}{\sigma^2-\mu} \qquad \pi = \frac{\mu}{\sigma^2} \tag{2.11}$$

where μ is the mean and σ^2 the variance of the target variable.
Following Lindén & Mäntyniemi [2011, p. 1415], we write the variance as a quadratic function of the mean:

$$\sigma^2 = \omega\mu + \theta\mu^2 \tag{2.12}$$

The parameters ω and μ are the overdispersion parameters. This process is defined as long as $\sigma^2 > \mu$ or in terms of the overdispersion parameters, when $\omega+\theta\mu > 1$. ω covers the linear mean-variance relationship used in the quasi-Poisson, θ covers the quadratic relationship used by the negative binomial. Lindén & Mäntyniemi [2011] derive the parameters of the negative binomial distribution as:

$$r = \frac{\mu}{\omega+\theta\mu-1} \qquad \pi = \frac{1}{\omega+\theta\mu} \tag{2.13}$$

These values can then be substituted into equation (2.10) to obtain the relevant likelihood in the equations (2.14) and (2.15)

$$P(Y=0|x) = 1-p(x) \tag{2.14}$$

$$P(Y=k|x) \propto p(x)\left(\theta\mu+\omega-1\right)^k,\ k\in\mathbb{N}_+ \tag{2.15}$$

A linear mean–variance relationship corresponds to the assumption of the quasi-Poisson regression. It is obtained by fixing θ to zero. If ω is fixed to 1, we get the quadratic mean–variance relationship of the regular negative binomial regression [Lindén & Mäntyniemi, 2011]. We follow from equation (2.13) that r is not defined for $\theta = 0$ and $\omega = 1$. Hence it is possible to use the quasi-Poisson as a special case of the negative binomial model, where $\omega \neq 1$, but not the regular Poisson model.

Chapter 3

Data

3.1 Description

The dataset has been build in cooperation with the federal police of switzerland (FEDPOL). It contains information from the unclassified Global Terrorism Database [2020] and from Jane's Information Group [2020]. The dataset used will be the property of FEDPOL and not publicly available. We looked at terrorist attacks between 1992 and 2020 in European countries without a war inside their territory during that period. Predicting attacks in war zones has been done multiple times already, see Tolan & Soliman [2015] or Uddin et al. [2020], and is easier than predicting attacks in peaceful areas because of an excessive amount of data and strong spatial and temporal dependency. We focus on right, left, and islamic attacks since these type of terrorist attacks are harder to predict than attacks from separatist groups, which have a strong spatial dependence. There are over 600'000 observations in total, but only 776 match our restrictions. The classification of the terror attacks into one of these categories led to cases, where it was not immediately clear what type of terrorism should be assigned. Hence the classification might contain errors and doesn't represent the opinion of the federal police. We created the variable 'Wounded & Killed' to quantify the threat of an attack. The variable sums up the amount of injured and murdered victims per attack. The perpetrator is not part of that variable. We are aware that there are many ways to measure threat, also that a kill could be considered worse than a wounding. We decided to not weight up a death against an injury, because this isn't a philosophical paper. The most threatening attacks were two of the attacks on the metro station in Spain with 487, respectively 469 wounded and killed people. The following section covers the data used for the analysis. Table 2 in the section 'Appendix B Tables & Pictures' lists the terror organisations in the analysis data set. This section further provides additional material describing the analysis data as well as the whole dataset.

3.2 Exploratory Data Analysis

Table 3.1 shows the summary statistics for the threat variable, the amount of 'Wounded & Killed' per attack, which is modelled through a quasi-Poisson hurdle- and a negative binomial hurdle model with overdispersion parameters in chapter 5. The first column shows the three types of attacks included in the analysis, μ stands for the mean and σ^2 for the variance. The amount of attacks is displayed by n. It is immediate to see that these categories are fundamentally different from each other. It seems that left-wing terror groups perform the most amount of attacks while threatening the least amount of lives, while islamic terror is rather rare but threatening to civilians. Hence the mean-variance relationship ($\frac{\sigma^2}{\mu}$) between these three groups is totally different. Table 3.2 shows the threat variable per country. We observe that the attacks mostly happened in Western Europe and Greece. Spain has the biggest amount of Wounded & Killed people on average and in total. The most amount of attacks have happened in Germany and Greece. They however have not been threatening on average.

Table 3.1: Summary Statistics of the Variable Wounded & Killed per type of terrorism for the data used in the analysis

Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	n
Islamic Terror	29.37	6111.87	0-487	208.1	99
Right Terror	2.66	180.17	0-164	67.8	245
Left Terror	0.48	6.32	0-40	13.3	432

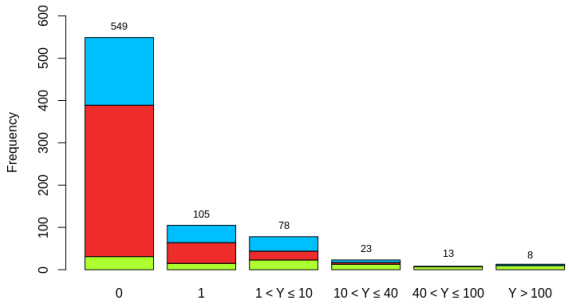
Table 3.2: Summary Statistics of the Variable Wounded & Killed per country for the data used in the analysis

Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	n
Austria	0.56	1.2	0-4	2.1	16
Belgium	27.00	2441.3	0-131	90.4	9
Bulgaria	41.00	-	41-41	-	1
Czech Republic	0.40	0.3	0-1	0.7	5
Czechoslovakia	1.00	-	1-1	-	1
Denmark	0.88	2.1	0-4	2.4	8
Finland	3.33	33.3	0-10	10.0	3
France	12.99	1560.6	0-288	120.2	69
Germany	1.91	65.6	0-74	34.3	230
Greece	0.20	0.6	0-7	3.0	226
Ireland	0.67	0.3	0-1	0.5	3
Italy	0.35	0.4	0-3	1.1	60
Netherlands	0.67	1.5	0-3	2.2	6
Norway	29.00	4383.2	0-164	151.1	6
Poland	1.25	6.3	0-5	5.0	4
Spain	49.56	17430.9	0-487	351.7	25
Sweden	2.00	25.9	0-19	12.9	14
Switzerland	0.00	0.0	0-0	-	7
United Kingdom	7.19	728.7	0-179	101.3	83

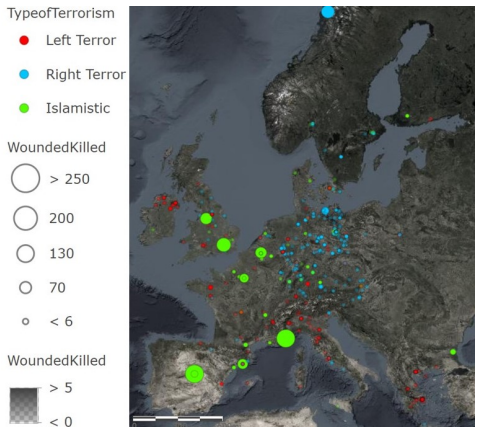
Figure 3.1a shows a barplot for the variable Wounded & Killed victims per attack. Attacks with multiple victims were merged together into categories to provide more visibility. The frequency of each category stands on top of each bar. The different colors symbolize the three types of terrorism, where Islamic terror attacks are in green, Left terror attacks in red and Right terror attacks in blue. Islamic terror is close to a uniform distribution over the categories. There is no left terror attack recorded with more than 40 wounded and killed victims. We observe that the amount of attacks without any victim is over 70%, but that there were also attacks with a high amount of victims.
Figure 3.1b visualizes these attacks. Transparent data points indicate less than five victims. The biggest circles refer to attacks with more than 250 victims. The attacks are again split into the three types of terrorism. We observe strong spatial clusters for the three types of terrorism. Italy and Greece were mostly exposed to left terror attacks while Germany had mostly right terror attacks. Generally do attacks from right extremist seem to occur mostly in Germanic[1] countries. Islamic terror attacks are by far the most deadly ones and primarily occur in big city areas.

[1] As a linguistic group, modern Germanic peoples include the Afrikaners, Austrians, Danes, Dutch, English, Flemish, Frisians, Germans, Icelanders, Lowland Scots, Norwegians, Swedes, and others [Lumen, n.d.]

Figure 3.1: 'Wounded & Killed' per terror attack for the three types of terrorism



(a) Split into categories

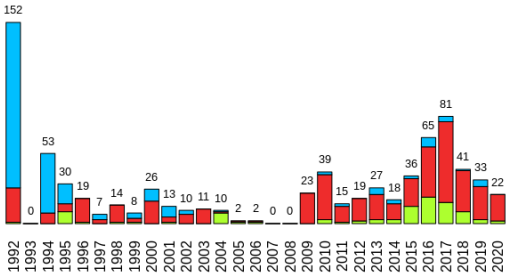


(b) ArcGIS Map [a]

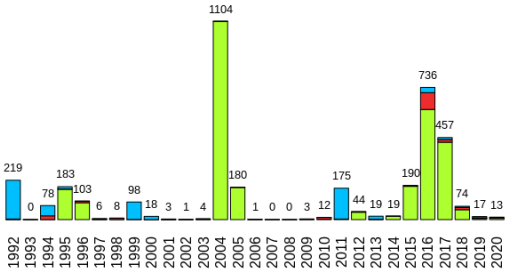
[a] Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AeroGRID, IGN, and the GIS User Community

The same classification and colors are used in figure 3.2 to visualize the amount of attacks over time, in 3.2a, and the amount of wounded and killed over time in 3.2b. The axes show the years and the numbers on top of the bar show the amount of attacks or the amount of victims per year. We observe that the amount of right terror attacks decline, while the threat per right terror attack increases. We further notice that the amount of left terror attacks has been increasing. Islamic terror attacks seem to take place in waves, where the first wave occurred in 1995/1996, the second wave in 2004/2005 and the third wave between 2015 and 2018.

Figure 3.2: The three types of terrorism over time



(a) Amount of attacks



(b) Amount of 'Wounded & Killed'

Table 3.3 lists the summary statistics for the threat variable Wounded & Killed for the different type of weapons used in the attacks. The last three columns show the preferred weapons per terrorism type. Vehicle impacts are the most threatening on average while mostly being used by islamic terror groups. Arson, mostly used to damage infrastructure, and small caliber weapons are generally used by left terrorists. They don't use knives and axes at all. Explosives are used by all three groups. The explosive devices used in left-wing attacks were made from an array of readily available materials. [Europol, 2020, p. 20] By looking at the average amount of wounded and killed, μ, in table 3.1 we assume that left terrorists use the explosives to cause damage to buildings and infrastructure, while islamic terrorists try to maximise the damage under civilians.

Table 3.3: Summary Statistics of the Variable Wounded & Killed per Weapon used for the data used in the analysis

					n			
Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	Σ	Islamic	Right	Left
Arson	0.12	0.12	0-1	1.0	8	-	2	6
Edged & Improvised	2.34	52.23	0-40	22.3	32	11	2	19
Explosives	11.11	2700.69	0-487	243.1	208	30	34	144
Firearm	7.64	702.66	0-164	92.0	39	13	5	21
Incendiary device	0.93	41.79	0-74	45.2	241	9	99	133
Knife & Axe	1.55	1.90	0-6	1.2	29	13	16	-
Other	1.17	11.08	0-24	9.5	82	9	20	53
Rocks	1.85	30.98	0-22	16.7	20	-	13	7
Small caliber	1.31	1.06	0-4	0.8	13	-	2	11
Unknown	0.68	2.89	0-12	4.3	93	5	51	37
Vehicle impact	66.56	8145.78	1-288	122.4	9	8	1	-
Virus/Malware	0.00	0.00	0-0	-	2	1	-	1

Table 3.4 splits the variable 'Wounded & Killed' into different types of terror attacks. The least threatening type of an attack is the 'Facility/Infrastructure Attack', where wounded or killed victims are unintentional. The attacks in the category 'Assassination' intentionally wounded or killed but were not threatening for civilians since there often was only a single target. The category 'Bombing/Explosives' in table 3.4 contains less attacks than the category 'Explosives' in table 3.3, since organised attacks are covered by the category NSAG (Non-State Armed Group). The categories can therefore not be one-to-one compared.

Table 3.4: Summary Statistics of the Variable Wounded & Killed per Event type used for the data used in the analysis

					n			
Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	Σ	Islamic	Right	Left
Armed Assault	2.87	152.44	0-74	53.1	62	2	51	9
Assassination	1.38	0.77	0-4	0.6	24	2	10	12
Bombing/Explosion	16.37	4872.27	0-487	297.7	103	19	37	47
Facility/Infrastructure Attack	0.18	1.35	0-12	7.4	143	1	94	48
NSAG Attack	4.21	455.52	0-288	108.2	418	75	31	312
Unarmed Assault	3.91	49.69	0-22	12.7	11	-	8	3
Unknown	2.67	9.24	1-12	3.5	15	-	14	1

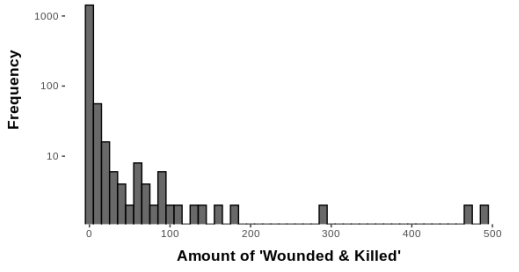
Table 3.5 shows the weapons used in the attacks and the type of an attack for the three different types of terrorism. NSAG attacks cover most of the organized terror attacks in Europe. Left and islamic terrorists appear more organized than right terrorist, with a relative percentage of NSAG attacks of over 70%, compared to around 13% for right terrorists. On first glance table 3.5 might show some surprising results such as an armed assault by islamic terrorists or two unarmed assaults by left terrorists carried out with explosives. Some attacks used different type of weapons in the process, in these cases only the most threatening one will be picked for the chart.

Table 3.5: Weapons versus Attack Type for different types of terrorism

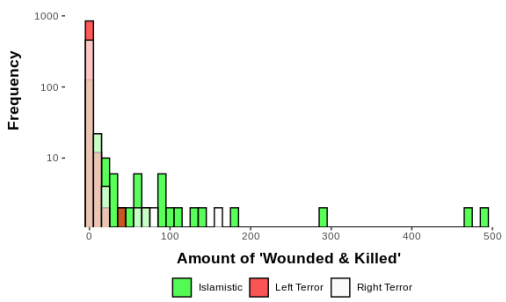
	Armed Assault	Assassination	Bombing/Explosion	Infrastructure Attack	NSAG Attack	Unarmed Assault	Unknown
Type of Terrorism: Islamic							
Edged & Improvised					11		
Explosives	1	1	15		13		
Firearm	1				12		
Incendiary device				1	8		
Knife & Axe					13		
Other					9		
Unknown		1	4				
Vehicle impact					8		
Virus/Malware					1		
Σ	2	2	19	1	75	0	0
Type of Terrorism: Left							
Arson	1			5			
Edged & Improvised					33		
Explosives	1		28	1	112	2	
Firearm		2			19		
Incendiary device	3			22	95		
Other		1		1	51		
Rocks				5		1	
Small caliber	2	9					
Unknown	2		19	14	1		1
Virus/Malware					1		
Σ	9	12	47	48	312	3	1
Type of Terrorism: Right							
Arson		1		1			
Edged & Improvised					2		
Explosives		1	28	2	3		
Firearm					5		
Incendiary device	34	1		55	9		
Knife & Axe	12	3			1		
Other	2	1		1	10	6	
Rocks				11		2	
Small caliber	2						
Unknown	1	3	9	24			14
Vehicle impact					1		
Σ	51	10	37	94	31	8	14

Figure 3.3 splits the target variable 'Wounded & Killed' into different subgroups and displays their histograms on the $\log_{10}$ scale. The data is duplicated to avoid the log of one. We observe a high frequency for low values, a sparsely populated middle section and some very high values. We further observe again the pattern that the most threatening attacks are committed by Islamic terrorists, while terror attacks carried out by left terrorists are barely ever threatful. The categories 'Weapons' and 'Event type' are less distinct, but we still observe patterns. Only the event type categories 'NSAG Attack' and 'Bombing/Explosion' have led to over 100 victims, while 'Unknown' and 'Facility/Infrastructure Attack' have not been threatening so far. The most damaging terrorist weapons have been 'Explosives' and trucks, shown as 'Vehicle Impact'. The two most threatening attacks were both committed with explosives through islamic terrorists. We will model the variable based on the shown categories in chapter 5.

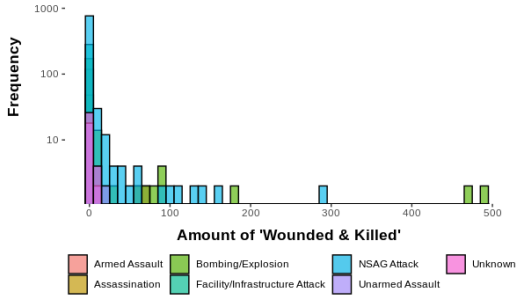
Figure 3.3: Histograms of the amount of 'Wounded & Killed' on the $\log_{10}$ scale



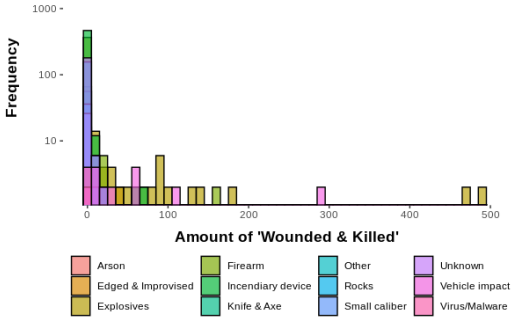
(a) No subgroups



(b) Split into Types of Terrorism



(c) Split into Event type

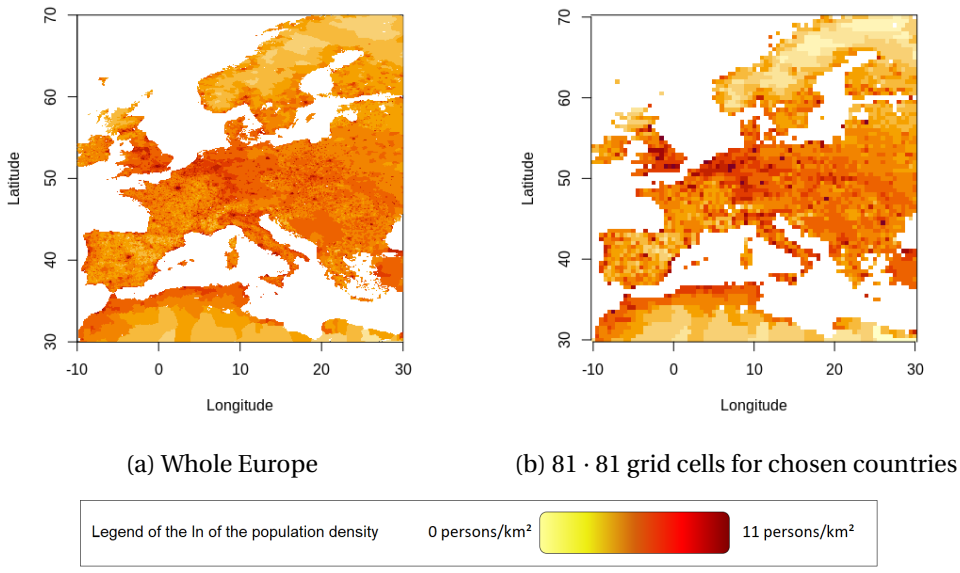


(d) Split into Weapons used

3.3 Spatial covariates

We used the gridded 'Population of Europe' data from 2000 provided by the 'Center for International Earth Science Information Network' and the 'Socioeconomic Data and Applications Center'. The altitude data and the temperature data were also from the year 2000 and provided by 'WorldClim.org' [2000]. Figure 3.4 shows the natural logarithm of the population density. Darker colors mark a higher density. 3.4a shows the density per km^2 for whole Europe , while 3.4b shows the data split into 81·81 grid cells for the countries included into the analysis. We include the spatial covariates for the regression gradient boosting method implemented in section 4.1. We also use the population density to distinguish between land and sea areas for the boosting method, the neighbourhood matrix and the spatio-temporal model, since with the exception of three cells in the alpes only ocean grids are truly unpopulated. These unpopulated areas have an attack probability of zero. Figure 3.5 and figure 3.6 show the temperature and the altitude for the chosen European countries. The altitude is measured by meters over the mean sea level while the temperature is given in degree Celsius. Darker colors again mark higher values. 'WorldClim.org' did not provide the altitude for a latitude higher than 60, hence to upper section of figure 3.6a is blank. We substitute the missing values with tiny values to differentiate them from ocean regions shown in figure 3.6b.

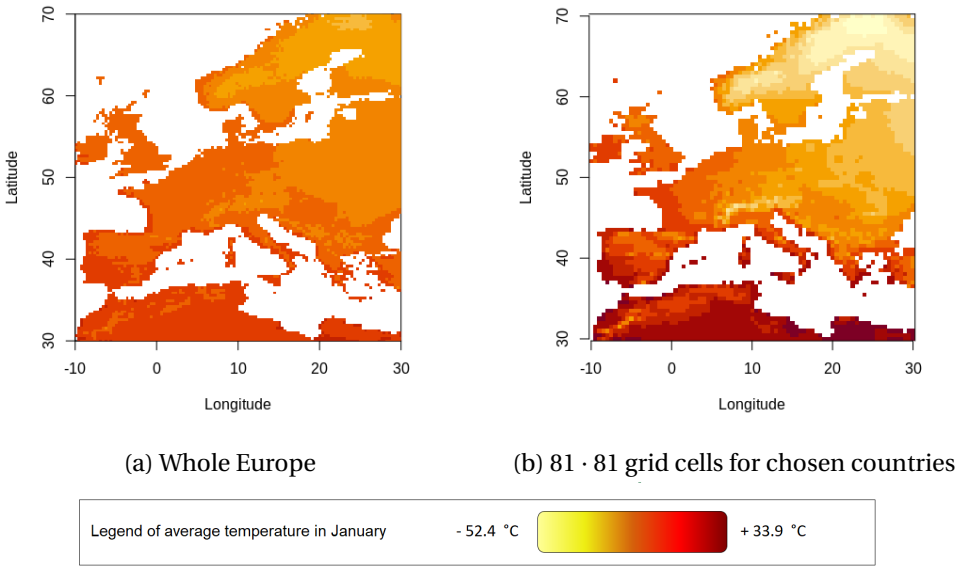
Figure 3.4: Natural logarithm of the Population Density of Europe



(a) Whole Europe

(b) 81 · 81 grid cells for chosen countries

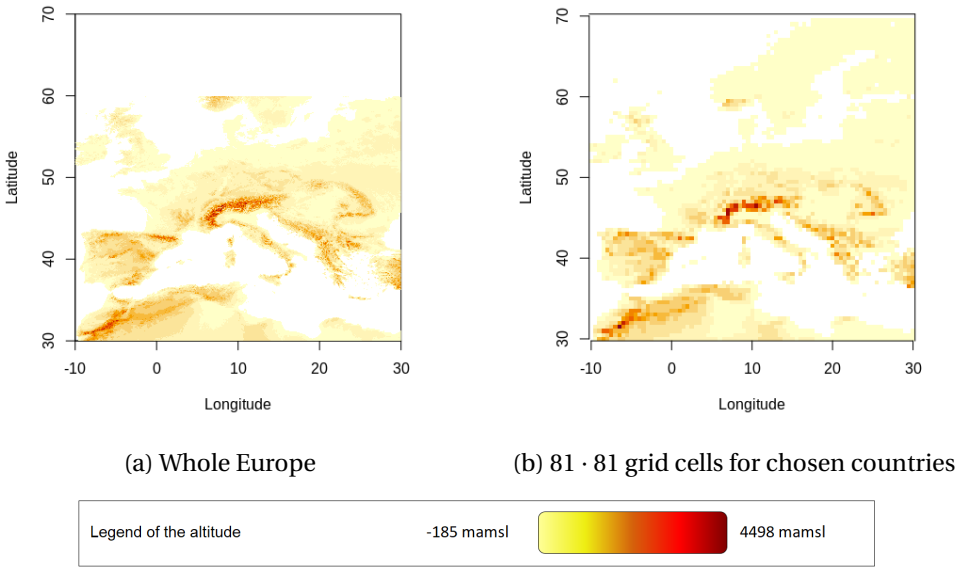
Figure 3.5: Average Temperature in January in Europe



(a) Whole Europe

(b) 81 · 81 grid cells for chosen countries

Figure 3.6: Topography of Europe in meters above the mean sea level



(a) Whole Europe

(b) 81 · 81 grid cells for chosen countries

Chapter 4

Binary Classification

4.1 Implementation of Spatial Models

To implement the spatial models introduced in section 2.1.1, we split the dataset into a training and a test set, where $\frac{2}{3}$ of the 776 attacks are part of the training set. Hence the amount of attacks $\mathbf{E}$ is split into $\sum \mathbf{E}_{Train}$ = 517 and $\sum \mathbf{E}_{Test}$ = 259. The latitude and longitude of the attacks included in the dataset are transformed to values between 1 and 81. The grids are then calculated by the transformed latitude times 81 plus the transformed longitude minus 81. Next we sum up the amount of attacks in the training data for each of these 6561 grid cells. We add the population density times zero to the spatial data points to get missing values at unpopulated grids to save some computational effort. These unpopulated grids only appear in sea regions, where no terror attack can happen.

Neighbourhood matrix

We estimated the only parameter in the AR(1) Neighbourhood model introduced in 2.1.1 through 10-fold cross-validation with the training set. The optimal value of the spatial correlation parameter is 0.15 indicating that an attack in a non-boundary grid cell adds $(1-4\cdot 0.15)/\sum \mathbf{E} = 0.4/\sum \mathbf{E}$ to its probability to be the next cell attacked, while each of its neighbour cells add $0.15/\sum \mathbf{E}$ to their probabilities.

Regression Gradient Boosting

We implemented the tree based boosted regression model through the gbm [Greenwell et al., 2020] package using the population density, the altitude and the temperature in January. The spatial predictors are explained in section 3.3. First we removed the missing values, the ocean grid cells, from the predictors. These grids will not get an attack probability accessed. We fitted an additive model on the training set with the shrinkage parameter λ equal to 0.1. We further used 10-fold cross-validation to access the optimal number of regression trees, which was 443.

We used a Poisson based loss function to model the amount of attacks per grid cell. Hence the normalizing step 4 of the algorithm in (2.1) for the transformation of the fitted values into a probability vector becomes:

$$\hat{\pi}_i = \frac{\exp(\hat{Y}_i)}{\sum \exp(\hat{Y})}, \quad i \in \{1, ..., 6561\} \tag{4.1}$$

4.2 Comparison of Spatial Models

Estimated probability matrices

Figure 4.2 shows the natural logarithm of the two estimated probability matrices based on the training set on top and the actual amount of attacks in the test- and training set split into the $81 \cdot 81$ grid cells on the bottom. The image of the Neighbourhood matrix looks like a blurred image of the attacks in the training set. It has attack probabilities of zero in cells faraway from the main attack areas. The estimated probabilities for the regression gradient boosting method are less distinct. These attack probabilities are not based on the spatial distance but on spatial covariates, hence they will never go to zero. The relative influence of the covariates on the prediction is shown in figure 4.1. Temperature and the population density explain the majority of the variance together. This result is supported by the literature, see Guo [2019] and Craig et al. [2019].

Figure 4.1: Relative influence of the spatial covariates as predictors in 'Regression Gradient Boosting'

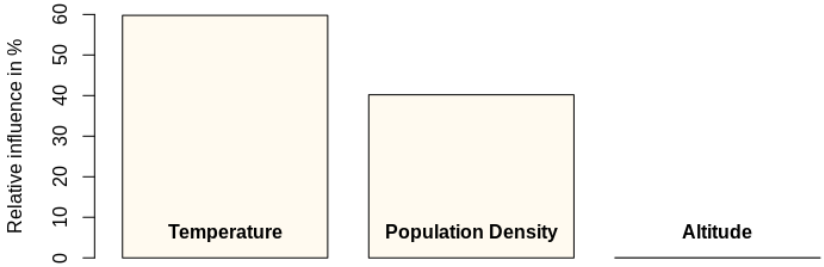
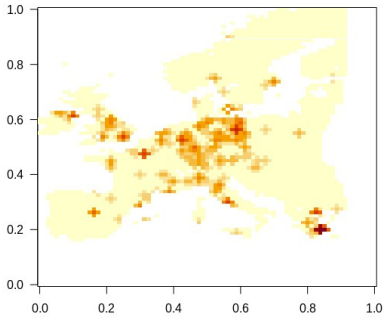
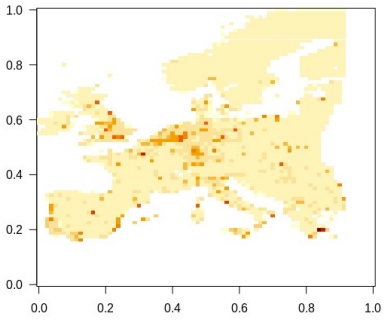


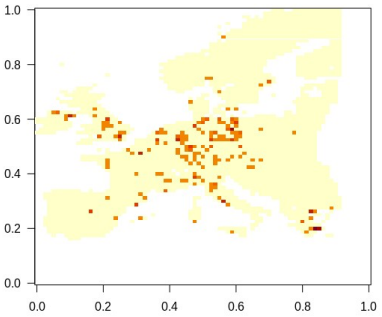
Figure 4.2: Natural logarithm of the attack probabilities of the spatial methods based on the training set compared to the actual amount of attacks in the training and test set



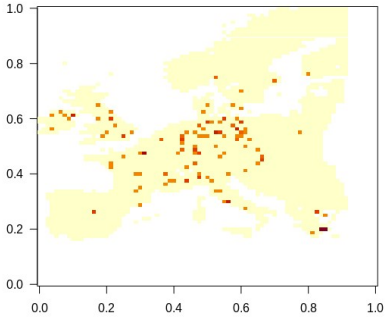
(a) Neighbourhood matrix



(b) Regression Gradient Boosting



(c) Attacks in the training set



(d) Attacks in the test set

Comparison on test set

The test set contains 259 attacks which occurred in 106 different grid cells. We therefore simulate 106 grid cells for both spatial methods based on their estimated probability vector. We repeat it 1000 times to compute confidence intervals for the percentage of correctly predicted attacked regions and compare them between the spatial models. We use a uniform probability vector as reference category. It gives each grid cell the same probability to be attacked. We compute $\hat{\bar{\pi}}$ as the average of each fraction of correctly predicted terror attack. Table 4.1 shows that both models perform significantly better than the random guess model, but it also indicates that using a complex model based on spatial covariates is worse than a neighbourhood matrix for terrorism data. We used our prior knowledge to construct a trivial model that outperforms the more complex model. This is not surprising since we observed a large cluster of attacks and regions without any attacks at all, which can not be explained by only looking at spatial covariates. The estimated probabilities should only be used to compare the models and not be looked at in absolute values since these values depend at the amount of grid cells and attacks. In the next section we add an additional time dimension to the strong spatial correlation.

Table 4.1: Average probability to correctly predict a terror attack through spatial models

	Uniform probabilities	**Neighbourhood matrix**	**Regression Gradient Boosting**
$\hat{\bar{\pi}}$	0.0404	0.2652	0.1735
SE[$\hat{\bar{\pi}}$ **]**	(0.0006)	(0.0009)	(0.0008)

Note: $\hat{\bar{\pi}} \in (0,1)$

4.3 Implementation of Spatio-Temporal Model

We use the same split into training and test set that we have used for spatial models. We again use the transformed latitude and longitude of the attacks to fit every attack into one of the 6561 grid cells.

Compute Spatio-Temporal Probability Matrix

Step1: Get Spatial Probabilities
In the first part of the implementation we look at all the days with an attack in the training set. Then we compute W_{ij} for every attack following equation (2.2). This results in a spatial probability vector for a day with attacks.

Step2: Add Spatial Probabilities to Days after Attack
We multiply the spatial probability vector with the exponential weight function, following equation (2.4), and add it to the next day.

Step3: Create Autoregressive time process
We look at the spatial probability vector for each day starting at 2.1.1992. We add the probability vector of the previous day multiplied with ϕ to the vector of the current day, following equation (2.3).

Step4: Normalize
We divide the probability matrix by its total sum. We use this step to simplify the optimization procedure. In theory does δ guarantee the values to sum up to one, but we save computational time by adding this step and relax the importance of δ in the process.

Parameter Optimization

The spatio-temporal model contains three parameter to estimate, ϕ, ρ and δ. We find the optimal parameters by using 5-fold cross-validation. This procedure compares the estimated probabilities based on 80% of the training set to the actual values of the 20% left out attacks in the training set through a squared loss function. The value of these five runs sums up to the value to minimize. The optimization procedure has been done manually for chosen combinations of parameter values to safe computational time. We can therefore not guarantee that these values reflect the global minima of the function. The results for different combinations are listed in Appendix B: Tables & Pictures as table 1 . We got the following optimal values for the parameters:

$$\hat{\rho} = 0.1, \qquad \hat{\delta} = 0.1, \qquad \hat{\phi} = 0.99 \tag{4.2}$$

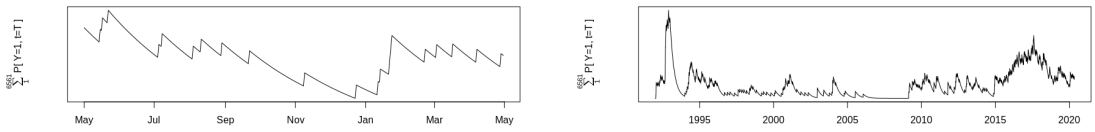
$\hat{\rho}$ represents the spatial spread. A small value indicates no long range dependency but a high probability in the same grid. $\hat{\delta}$ stands for the variance of the Gaussian kernel and functions as a

scaler. It's influence is marginal due to addition of the normalization step. $\hat{\phi}$ is the autoregressive time parameter and indicates that the risk of a terror attack decreases by 1% per day passed. $\hat{\phi}$ being that high indicates a weak temporal correlation.

Analysis

We use the parameters, estimated through cross-validation, to construct a probability matrix based on the training set. The r-code is attached in the 'Appendix C: R-code'. Figure 4.3 displays the accumulated attack probabilities per day based on the training set. Figure 4.3b presents the attack probabilities for all the days in the analysis. We observe that there was a long period without any attacks between 2005 and 2010 in the training set. We further observe that the amount of attacks increases over time. Figure 4.3a zooms into the last year. We notice a sudden jump followed by a weak exponential, close to a linear, decrease.

Figure 4.3: Summed up attack probabilities per day



(a) 1. May 2019 - 30. April 2020

(b) 1. January 1992 - 30. April 2020

Table 4.2 presents the average probability of correctly predicted attacks achieved on the 259 attacks of the test set for 100 Monte Carlo simulations. This number is quite small due to the computational limitations. 'Exact Location and Day' provides 6561×10348 possible time-space possibilities for an attack or 2538×10348 if the ocean regions get neglected. The 259 attacks in the test set are distributed on 223 different days. The terror attacks are coded by day. 'Exact Day' aggregates these attacks per day. It counts the days with a simulated attack and an attack in the test set. We suspected a strong spatio-temporal relationship within one day, because these attack often occur within hours, as seen in Paris[1] or in Madrid[2], and can therefore not be captured by daily coded data. A specific region might be safer than usual on the day after an attack due to increased police presence there. Hence we also look at 'Exact Day $\pm$ 1 Day', which counts a predicted attack within one day of an actual attack. We observe that around 13% of the predicted days with terror attacks are within one day of actual days with attacks.

[1] Suicide bombers from the islamic state attack multiple places in Paris on the 13. November 2015 [Global Terrorism Database, 2020]

[2] AL Qaida bombs Railway Station in Madrid on 11. March 2004. Attacks happen within minutes in different places of the city. [Global Terrorism Database, 2020]

Table 4.2: Average probability to correctly predict a terror attack through spatio-temporal model with spatial Gaussian kernel and an AR(1) time process.

	Exact Location and Day	**Exact Day**	**Exact Day $\pm$ 1 Day**
$\hat{\bar{\pi}}$	0.00096	0.04834	0.13152
SE[$\hat{\bar{\pi}}$ **]**	(0.000002)	(0.000151)	(0.000223)

Note: $\hat{\bar{\pi}} \in (0,1)$

Chapter 5

Threat Level Classification

5.1 Model Estimates

The implementation of the hurdle models with overdispersion, introduced in section 2.2, is done in two steps. First, we model the probability of a value higher than zero through a logistic regression, implemented in R through the glm function. Then we use iteratively weighted least squares (IWLS) to model the positive count data loosely following the approach of Ver Hoef & Boveng [2007]. The IWLS r-code is attached in 'Appendix C: R-code'. We wrote the code because we didn't find a R implementation of the negative binomial model with two overdispersion parameters. The choice of covariates is limited due to a rather small amount of events with wounded or killed people, shown in figure 3.1a, and exclusively factor variables in the dataset. We compare different models with at most one covariate per model, because multilinearity is otherwise a big concern.

5.1.1 Hurdle Model

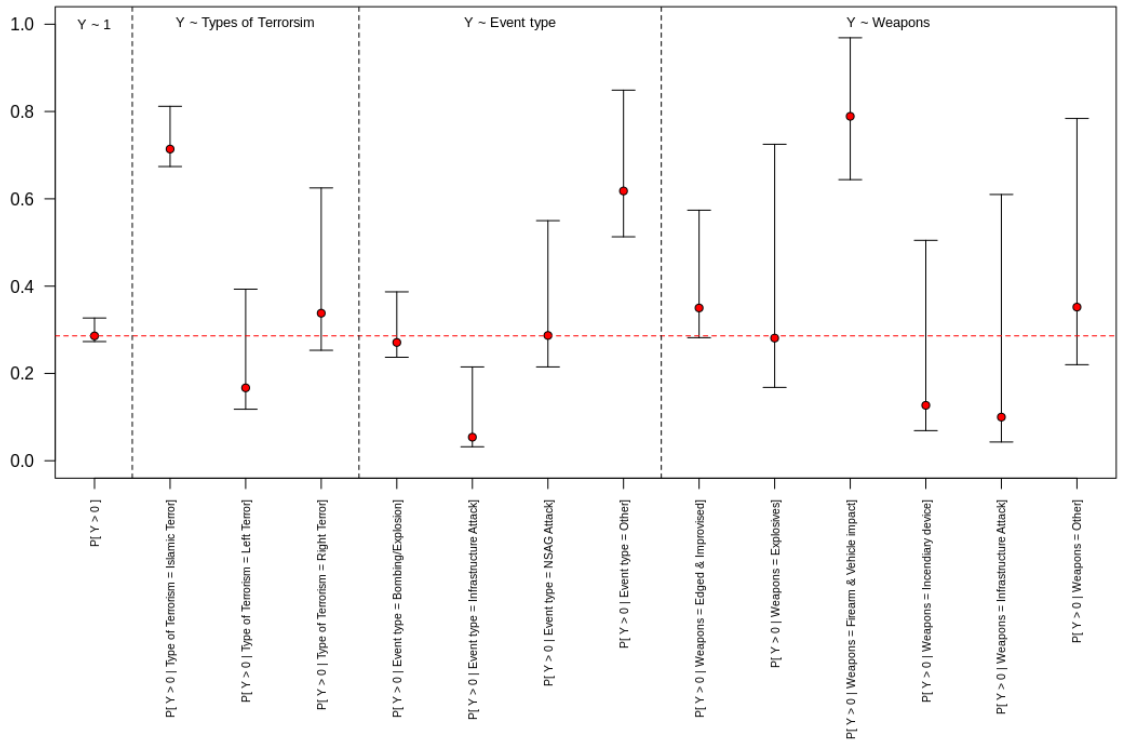
The hurdle step applies to all threat classification models. The total amount of events is 776. We split again two third of the data into a training set used for the modelling. We use logistic regression to estimate the coefficients of four different models. Figure 5.1 shows the marginal probability for each characteristic to surpass the hurdle and get a value higher than zero. $Y \sim X$ corresponds to the regression model with the covariate X. The dotted red line shows the relative amount of non zero values in the training set. Each probability has its 95 % confidence interval attached. The vertical dotted lines divide the probabilities into the four logistic models shown at the top of the figure. Categories of the variables 'Event type' [1] and 'Weapons' [2] were merged into

[1] Categories of variable 'Event type': 'Armed Assault', 'Unarmed Assault', 'Unknown' and 'Assassination' were merged into category 'Other'.

[2] Categories of variable 'Weapons': 'Small caliber', Firearm', 'Vehicle impact' were merged into category 'Firearm & Vehicle impact'; categories 'Virus/Malware', 'Arson' and 'Rocks' were merged into category 'Infrastructure Attack'; categories 'Knife & Axe' and 'Unknown' were merged into category 'Other'.

fewer categories to avoid multicollinearity in the IWLS estimate. We state from figure 5.1 that Islamic terrorism has a significantly higher probability to cause victims in their attacks then the other types of terrorism. The event type category 'Infrastructure Attack' has the biggest chance to not injure or kill someone in an attack. The newly formed event type category 'Other' has a high probability to surpass the hurdle as does the newly formed weapon category 'Firearm & Vehicle impact'. Y is the amount of wounded or killed victims per attack.

Figure 5.1: 95 % confidence intervals of the marginal probabilities to exceed zero for covariate characteristics in the hurdle step



5.1.2 Count Models

We model the overdispersion through a linear, a quadratic and a linear and quadratic mean-variance relationship. These three distributions were modelled using iteratively weighted least squares. There are 152 observations in the training set with atleast one wounded or killed victim. The relative small sample size restricts the covariate choice. Hence we are forced to pass on modelling interaction terms of the covariates nor can we use different categorical variables at once. Further we had to recode the categorical variables 'Event type' and 'Weapons' as explained in the 'Hurdle Model' subsection. The estimates of the overdispersion parameters are displaced in table 5.1. $\hat{\omega}$ covers the linear-, while $\hat{\theta}$ covers the quadratic mean-variance relationship. $\hat{\phi}$ is the dispersion parameter showing how many times the variance is bigger than the mean. It is derived in equation (5.1).

$$\hat{\phi} = \frac{\hat{\sigma}^2}{\hat{\mu}} = \frac{\hat{\mu}\left(\hat{\omega}+\hat{\theta}\,\hat{\mu}\right)}{\hat{\mu}} = \hat{\omega}+\hat{\theta}\,\hat{\mu}\,, \quad \hat{\mu} = \frac{1}{152}\sum_{i=1}^{152}\hat{\mu}_i = 14.36 \tag{5.1}$$

The first model, the intercept only model, shows the same overdispersion estimates for the negative binomial distributions. It also shows the identical dispersion value for all three distributions, which follows from the fact that the mean is a constant, hence $E[\hat{\mu}^2] = E[\hat{\mu}]^2$. The other three models contain covariates, therefore the mean per covariate category varies, which leads to a different (over)dispersion between the quasi-Poisson and the negative binomial models. The dispersion for the two negative binomial models is identical by construction. We follow that for categories with a high estimated mean, the negative binomial2 will give a high variance, but for categories with a small amount of wounded and killed victims we expect it to estimate a smaller variance than the other two methods. We expect the quasi-Poisson model, which has only a linear mean-variance relationship, to have a smaller variance for high mean values in comparison to the two negative binomial models with their quadratic mean-variance relationship.

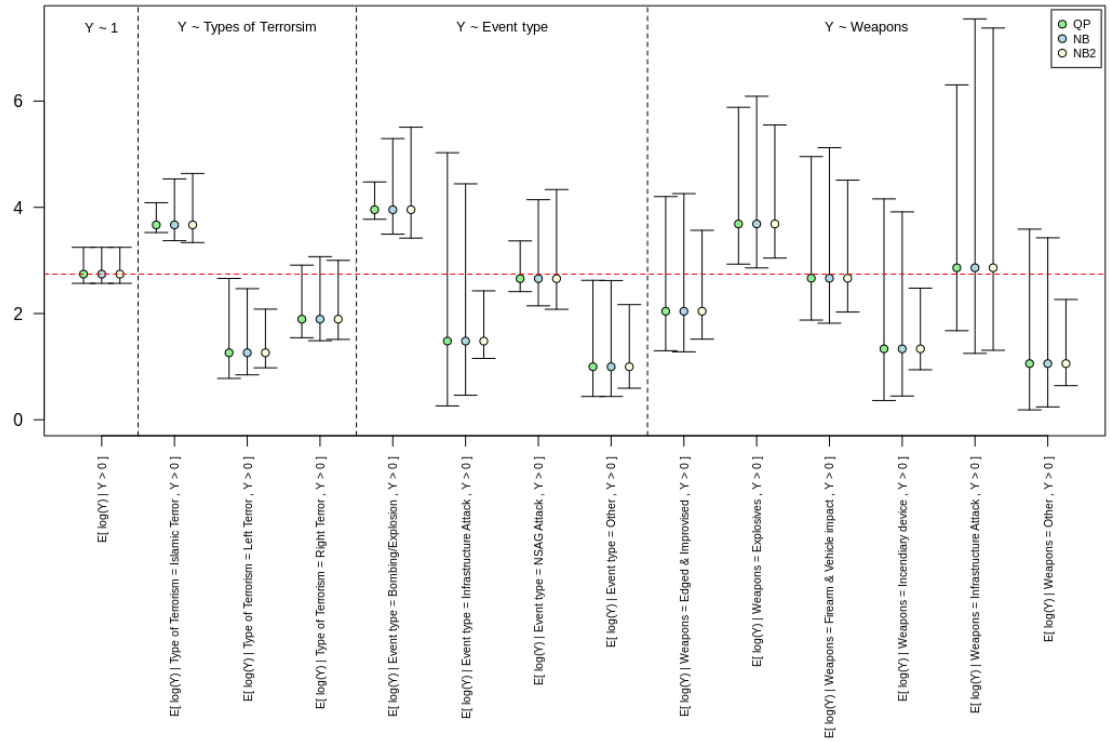
Table 5.1: Estimates of the overdispersion parameters ω and θ and the dispersion parameter ϕ for the quasi-Poisson, the negative binomial with one and with two overdispersion parameters, respectively, for four different models

		Y ~ 1	**Y ~ T**	**Y ~ E**	**Y ~ W**
Quasi-Poisson	$\hat{\omega}$	152.8	80.3	70.5	65.5
	$\hat{\theta}$	0	0	0	0
	$\hat{\phi}$	152.8	80.3	70.5	65.5
Negative binomial	$\hat{\omega}$	1	1	1	1
	$\hat{\theta}$	9.8	8.8	8.8	8.8
	$\hat{\phi}$	152.8	136.9	138.3	137.8
Negative binomial2	$\hat{\omega}$	1	-55.5	-67.3	-71.3
	$\hat{\theta}$	9.8	12.4	13.2	13.3
	$\hat{\phi}$	152.8	136.9	138.3	137.8

Notes: $\hat{\phi} = \frac{\hat{\sigma}^2}{\hat{\mu}} = \hat{\omega} + \hat{\theta} E[\ \hat{\mu}\ ]$, T = Types of Terrorism, E = Event type, W = Weapons

Figure 5.2 shows the 95 % confidence intervals for the expected value of 'Wounded & Killed' victims per attack given that atleast one victim has been wounded or killed on the natural log scale. The vertical dotted lines divide the different models. The dotted red line corresponds to $E[Y|Y > 0]$, where Y is the amount of wounded or killed victims per attack. The green dots represent the quasi-Poisson, the blue dots and the yellow dots represent the negative binomial with one and two overdispersion parameter, respectively. We observe that the estimated mean per categories are equal for the three distributions, but they differ in terms of variance. We observe that 'Types of Terrorism' has a huge influence on the expected amount of Y. As expected does the quasi-Poisson model have a narrower confidence interval for 'Islamic Terror', but a bigger confidence interval for 'Left Terror', compared to the other two methods. We observe that with a smaller expected value for Y, the confidence intervals for the quasi-Poisson get larger in relation to the other two methods. The intervals of the negative binomial2 model get larger with a bigger expected value in relation to the other two methods, as we expected from table 5.1. We also observe that the categories from 'Event type' and 'Weapons' do not overlap much, even though some of them share the same name, since we estimate different mean and variances for them. Generally we expect samples from the negative binomial2 to perform well in comparison to the other two methods, since a high variance for high mean values increases the chance of getting these very large values in the training sample.

Figure 5.2: 95 % confidence intervals for the expected value of 'Wounded & Killed', given that atleast one has been wounded or killed. The intervals cover different covariate characteristics using the quasi-Poisson (QP), the negative binomial with one (NB) and the negative binomial with two overdispersion parameter (NB2).



5.2 Model Comparison through Simulations

5.2.1 Implementation of Sampling

We estimated the parameters for the three distributions for four models each. Now we take samples from these distributions. We simulate these samples through the R-function rnbinom() from the stats package, which only allows θ_{NB} as a parameter. Hence we derive the following equations using the variance decomposition in equation (5.2), which is derived from equation (2.12).

$$\sigma^2_{QP} = \mu\left(\omega_{QP} + \mu 0\right) \qquad \sigma^2_{NB} = \mu\left(1 + \mu\theta_{NB}\right) \qquad \sigma^2_{NB2} = \mu\left(\omega_{NB2} + \mu\theta_{NB2}\right) \tag{5.2}$$

Equation (5.2) shows again the variance decomposition of the three methods. Lets set σ^2_{QP} equal to σ^2_{NB} and solve for θ_{NB}, the overdispersion parameter implemented in rnbinom. We then get

$$\theta_{NB} = \frac{(\omega_{QP} - 1)}{\mu} \tag{5.3}$$

which is the transformation of ω_{QP} used to model a linear mean-variance relationship inside the rnbinom framework. To get the transformation for the double parametrized negative binomial model we set $\sigma^2_{NB2} = \sigma^2_{NB}$ and solve again for θ_{NB}. We get

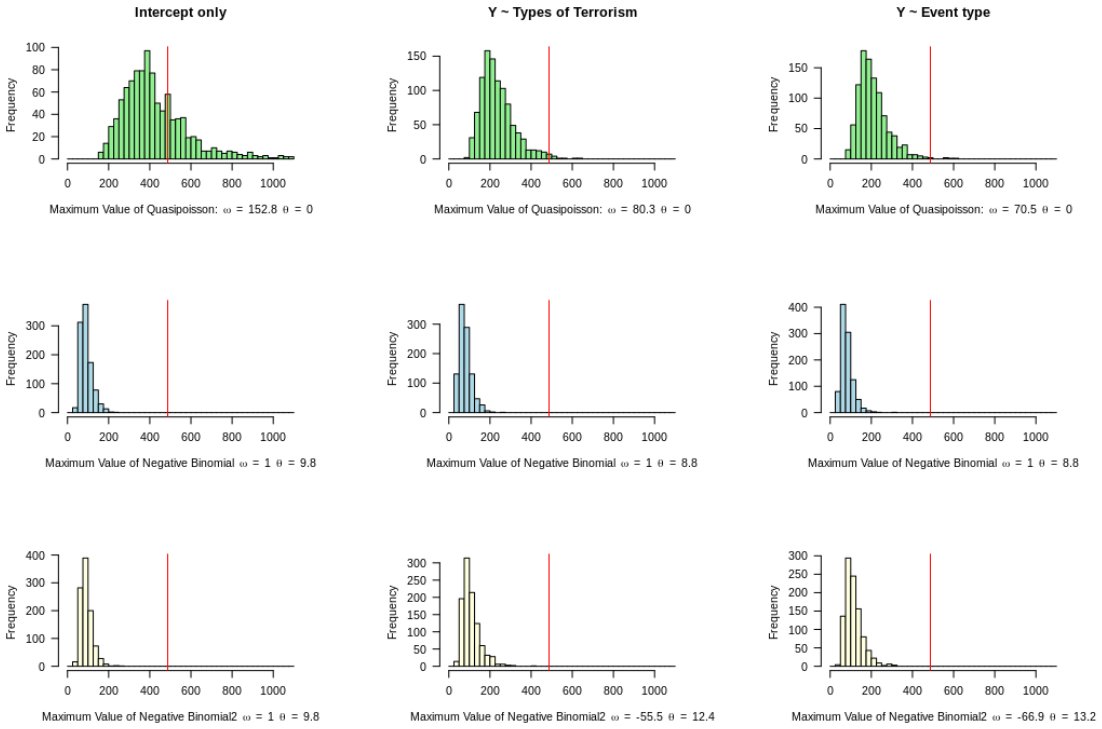
$$\theta_{NB} = \frac{(\omega_{NB2} - 1)}{\mu} + \theta_{NB2} \tag{5.4}$$

which corresponds to the transformation used in rnbinom to sample from a model with a linear and a quadratic mean-variance relationship. The r-code of the sampling function for overdispersed hurdle models is provided in 'Appendix C: R-code'

5.2.2 Comparison on the training set

Coverage probabilities of chosen values

Figure 5.3: Histograms of the maximum value of 1000 simulated samples, containing 517 observations each, based on the training set characteristics, for three different models using the quasi-Poisson, the negative binomial and the double parametrized negative binomial



We simulate a sample from the three estimated distributions based on the characteristics of the training set. We sample with an only intercept model, a model with the three types of terrorism and a model with four categories of event types. We then observe the maximum value per model and sampling distribution. This procedure was repeated 1000 times and is shown in figure 5.3. The left column shows the maximum values for a model with only the intercept, the middle column samples from a model with different expectations per types of terrorism, where the right column has different expectations per event type. The red line shows the true maximum value observed in the training set. In green we sampled the quasi-Poisson with θ fixed to zero, in blue we sampled from the regular negative binomial with ω fixed to 1 and in yellow we allow both parameters to vary. Noteable that the double parametrized negative binomial and the

regular negative binomial are identical for the intercept only model. We observe that the linear mean-variance relationship model covers the maximum value best.

Figure 5.4: Histograms of the percentage of zeros of 1000 simulated samples, based on the training set characteristics, for three different models using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

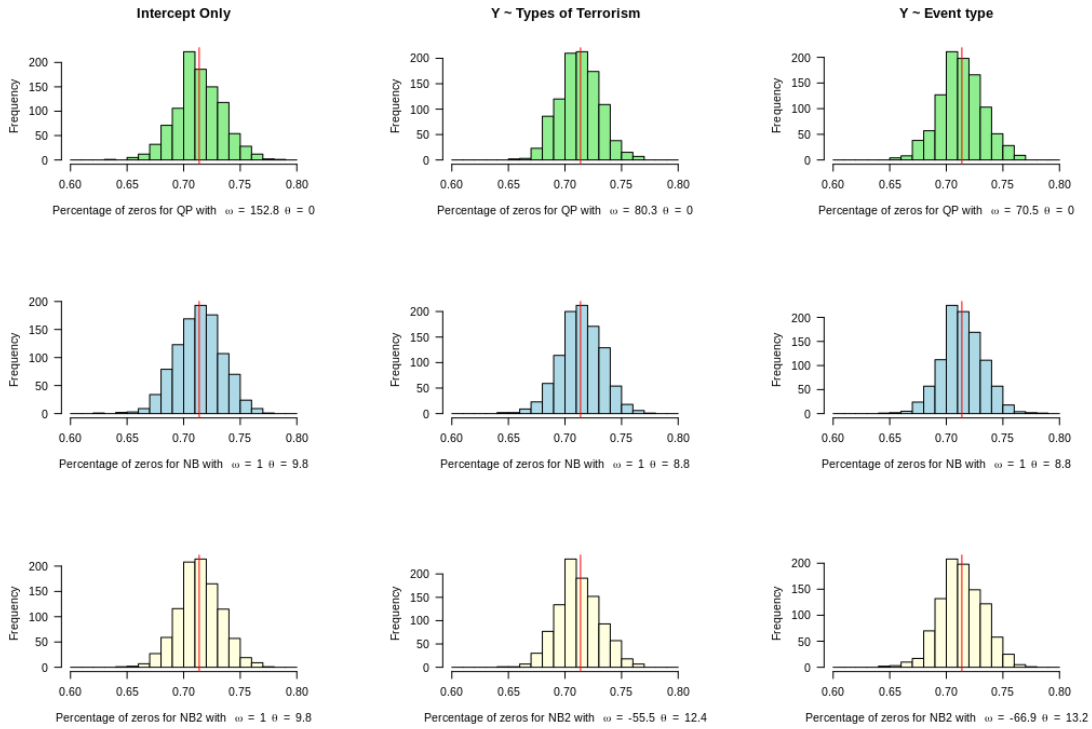


Figure 5.4 shows the relative amount of zeros per simulated sample. We cover the true value of 71.3% in the training set, shown as the red line, well for every distribution and model. This is no surprise since we force the amount of zeros to be in that range through a separate hurdle process. The barplot, 3.1a, indicated that the data might contain an excess of ones as well. We see support for that claim in figure 5.5, which shows the relative amount of ones per sample. Every distribution and model undersamples the amount of one wounded & killed victim per attack. It seems that there is an other seperate process that divides the threat for a single person with the threat for multiple people. However we would require more datapoints to implement a second hurdle step.

Figure 5.5: Histograms of the percentage of ones per of 1000 simulated samples, based on the training set characteristics, for three different models using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

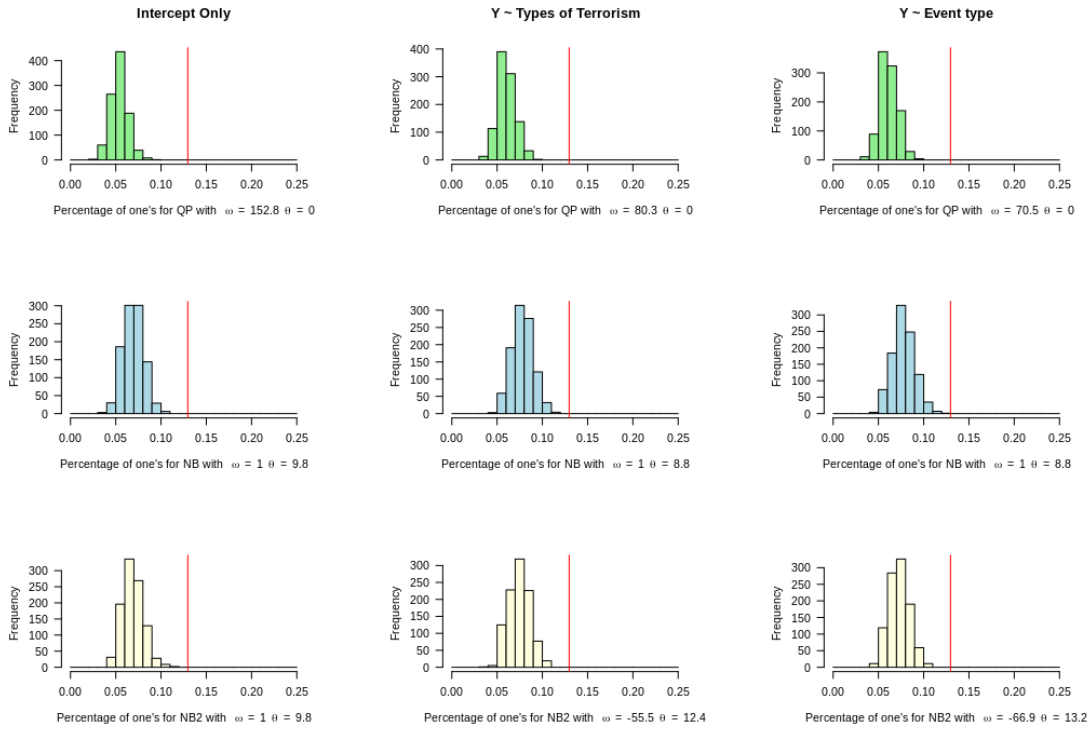
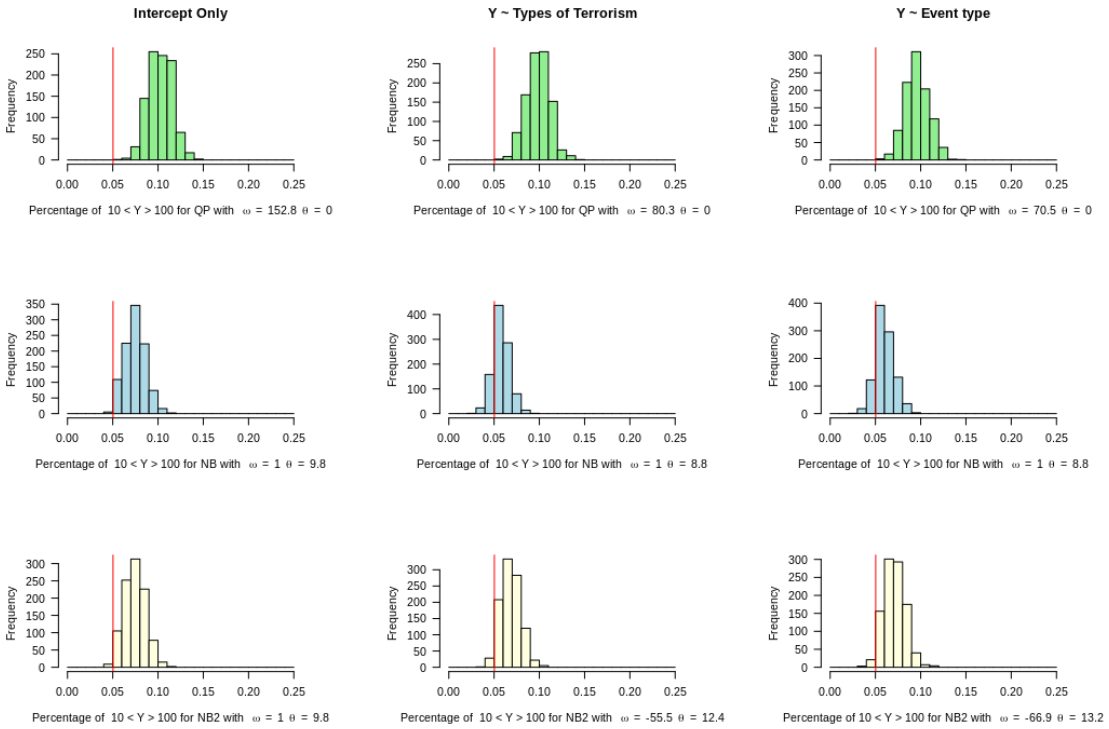


Figure 5.6 shows the same histograms for values higher then 10, but smaller then 100. We see that the all three distributions tend to oversample the relative amount of these values. Specially the quasi-Poisson, which was able to cover the extreme values in figure 5.3 well, performs badly here. Generally we observe that the threat variable 'Wounded & Killed' follows a unique distribution with a strong excess of zeros, a light excess of ones, few medium sized values and some very high values. Histograms of the relative amount of other interesting values and of the model, **Y ~ Weapons**, are attached in the 'Appendix B Tables & Pictures'.

Figure 5.6: Histograms of the relative amount of values between 10 and 100 of 1000 simulated samples, based on the training set characteristics, for three different models using the quasi-Poisson, the negative binomial and the double parametrized negative binomial



Covariate Categories

It is important to get a similar coverage for the simulated and the observed data, but in order to distinguish between the threat of different attacks, it is more important to achieve variance through the covariate categories. The next four figures show the quasi-Poisson and the two negative binomials for the intercept only model in figure 5.7, the 'Type of Terrorism' model in figure 5.8, the 'Event type' model in figure 5.9 and the 'Weapons' model in figure 5.10. These overlapping histograms show the $\log_{10}$ of the count for the same simulated values then before, split for the covariate categories. We compare them to the histograms for the observed values in 'Histograms of the Data' in the appendix. We observe again that the quasi-Poisson generates more high values for all models than the other two methods. We also observe that the quasi-Poisson struggles to create variance between different categories of the variables. We further discover that the histograms of the observed values contain less values in the middle section, than the simulated values. Figure 5.7 shows the simulated data without covariates present. Using the quasi-Poisson leads to high values suggesting up to 1000 'Wounded & Killed' victims in an attack. The negative binomials are identical if no covariates are present. They underestimate the threat by not simulating values above 250.

Figure 5.7: Histograms of the simulated values for the quasi-Poisson, the negative binomial with one and the negative binomial with two overdispersion parameters.

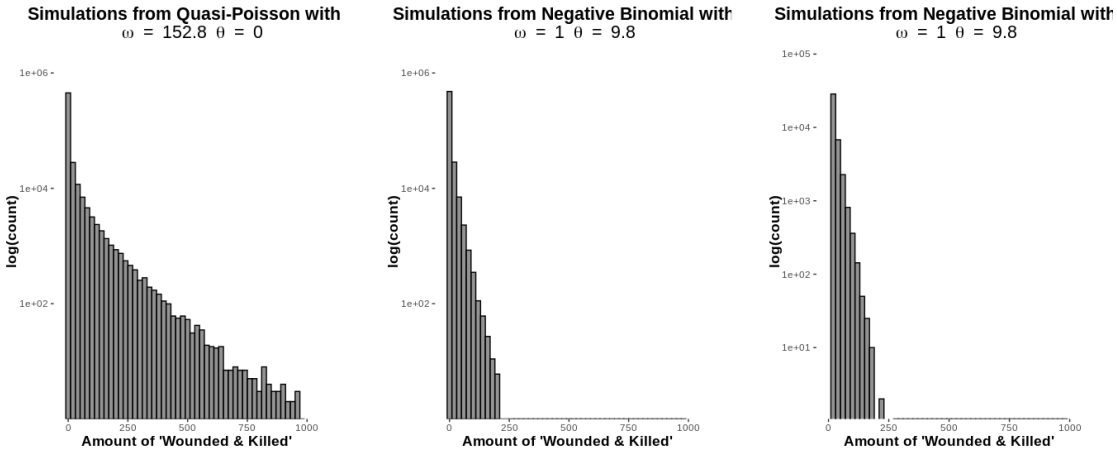
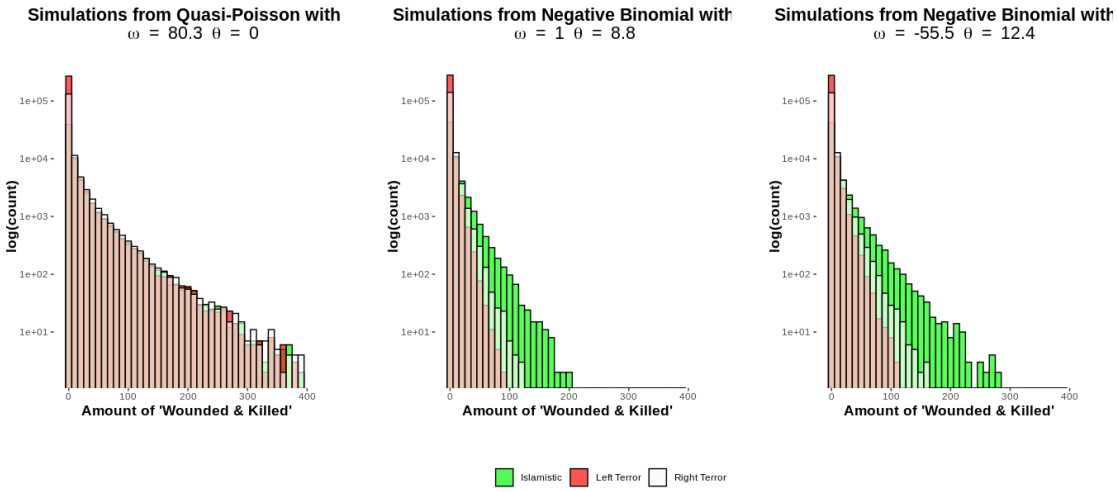


Figure 5.8 shows that the quasi-Poisson does not distinguish well between the three types of terrorism. The negative binomial with a single overdispersion parameter creates variance between the categories but fails to cover the extreme high values in the data. The distribution on the right represents the negative binomial distribution with two overdispersion parameters. It also cre-

ates a nice variance between the categories and comes closer to simulate big values than the other negative binomial distribution. The light brown color stands for all three types of terrorism. We immediately see that the creation of zeros follows a different process, since they have an equally high $\log_{10}$ count and are equally split across categories for the three distributions.

Figure 5.8: Histograms of the simulated values for the covariate **'Types of Terrorism'** for the quasi-Poisson, the negative binomial with one and the negative binomial with two overdispersion parameters.



We observed from the model estimates in figure 5.2 that the categories in 'Event type' and 'Weapons' are not well seperated, hence we expect both models to produce worse results than the 'Types of Terrorism' model. Figure 5.9 shows the histograms split into the four event types. Again we see that the variance between the categories increases from the left graph to the right graph. The negative binomial models categorise 'NSAG Attack' and 'Bombing/Explosion' as the two most dangerous types of attacks. They never suggest a 'Facility/Infrastructure Attack' to cause more then 100 victims, while the quasi-Poisson model can suggest up to 300 victims. The highest observed amount of victims for that category reaches 12. Figure 5.10 reveals the simulated amount of 'Wounded & Killed' per 'Weapons' category. Both negative binomial models identify 'Explosives' as the most dangerous 'Weapons' category. The negative binomial with two overdispersion parameters predicts higher values than the other negative binomial here. The quasi-Poisson again fails to create variance between the weapon types.

Figure 5.9: Histograms of the simulated values for the covariate **'Event type'** for the quasi-Poisson, the negative binomial with one and the negative binomial with two overdispersion parameters.

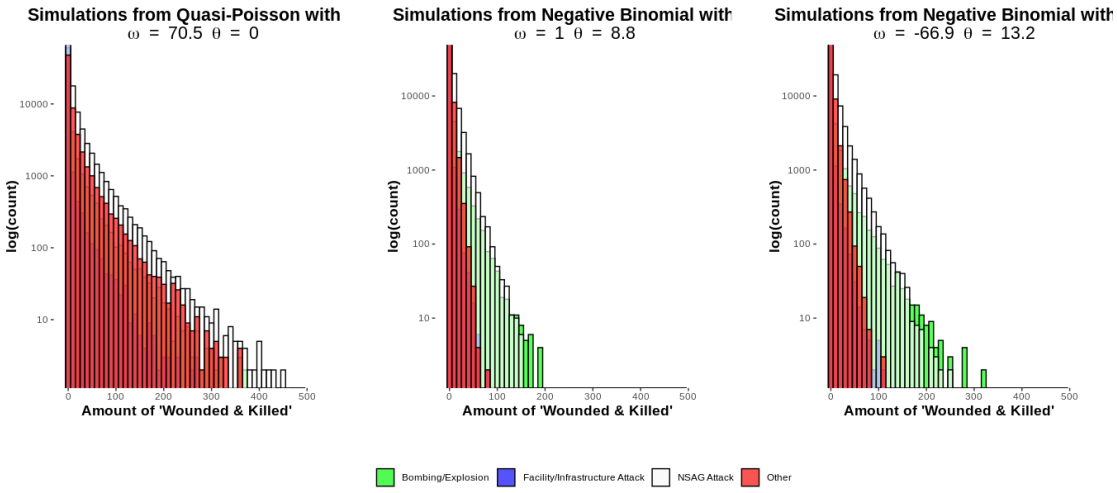
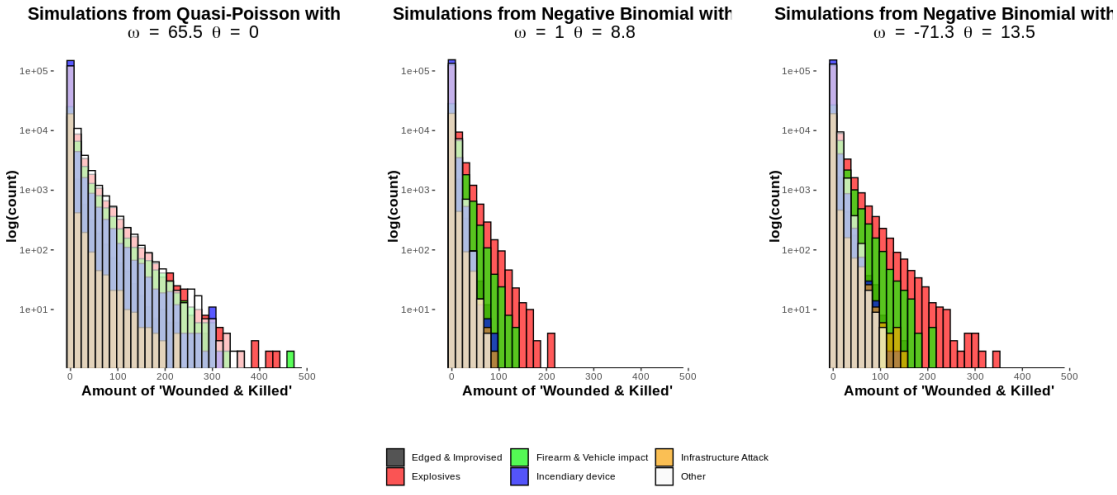


Figure 5.10: Histograms of the simulated values for the covariate **'Weapons'** for the quasi-Poisson, the negative binomial with one and the negative binomial with two overdispersion parameters.



5.2.3 Comparison on the test set

Goodness of Fit tests

We simulate again 1000 times the quasi-Poisson, the regular negative binomial and the negative binomial with two overdispersion parameters for each of the four covariate models. We now use a model matrix based on the test set. Hence we get 12 simulated samples with the size of 1000 times the length of the test set. How to access the goodness of fit of the models is a non-trivial question. First we use the nonparametric Wilcoxon signed-rank test invented by Wilcoxon [1945]. We use it on each of the 1000 samples with the test set shown in equation (5.5). This test statistic allows the data points to be matched and compared in order to determine whether they follow the same underlying distribution. The Wilcoxon signed-rank test does not require the data to be continuous and does value a clear differentiation between covariate categories since it looks at the ranks of the paired data. The disadvantage of this test is that it only looks at the ranks of the data and therefore does not control for overdispersion. It is implemented in wilcox.test() in the R Core Team [2020] package.

Paired Wilcoxon Signed-Rank Test

$$
\begin{aligned}
&\texttt{Step 1: Rank the paired differences according to their absolute value}\\
&\texttt{Step 2: Compute the sum of the positive ranks } Sp \texttt{ and the sum}\\
&\qquad\texttt{of the negative ranks } Sn\\
&\texttt{Step 3: Compute the test statisic: } W_R := \min(Sp, Sn)\\
&\texttt{Step 4: Compute the mean and standard deviation of } W_R\texttt{:}\\
&\qquad \mu_{W_R} = \frac{n(n+1)}{4}\\
&\qquad \sigma_{W_R} = \frac{n(n+1)(2n+1)}{24} - \frac{\sum t^3 - \sum t}{48}\\
&\texttt{Step 5: Compute the z-value } z_W = \frac{W_R - \mu_{W_R}}{\sigma_{W_R}},\ z_W \overset{H_0}{\sim} N(0,1)
\end{aligned}
\tag{5.5}
$$

where n is the length of the test set and t represents the number of times the i^{th} value occurs. For large n are the p-values computed through the standard normal distribution.

[NCSS, n.d.]

We will introduce the quantile method which is mindful considering the actual simulated values and therefore takes overdispersion into account. It is described in equation (5.6). The idea of this method is to reward the sampling distributions for having a small variance while punishing

a big shift in location. The result of this method should be analysed with caution since there is no underlying literature for it. We removed the zero values since they follow a different underlying process, hence their actual number is fixed to the correct number of zeros in the data of around 70 % on average. An other advantage of removing the zeros is that the computed value t becomes more alert to the 25 % quantile and the median which otherwise would always show zero. We will further look at some basic summary statistics.

Method based on Quantiles

$$
\begin{aligned}
&\texttt{Step 1: Remove zero values, } Y = Y > 0 \\
&\qquad \texttt{a) Only keep simulations for test set value bigger 0} \\
&\qquad\quad \texttt{if } Y_i^{Test} > 0\,, \ \texttt{keep } Y_{i,1:1000}^{SIM} \\
&\qquad \texttt{b) Remove zero values for the simulated values } Y_{i,j}^{SIM} > 0 \\
&\texttt{Step 2: Compute quantile values, } \left(X_{(1)}, X_{(n/4)}, X_{(n/2)}, X_{(3n/4)}, X_{(n)}\right), \texttt{ for } Y_i^{SIM} \\
&\texttt{Step 3: Compute value t:} \\
&\qquad \texttt{a) } X_i = \left(X_{i,(1)}, X_{i,(n/4)}, X_{i,(n/2)}, X_{i,(3n/4)}, X_{i,(n)} \right) \\
&\qquad \texttt{b) } t := \sum_{i=1}^{5} \sum_{j=1} |X_{i,j} - Y_i^{Test}|
\end{aligned}
\tag{5.6}
$$

Y_i^{Test} represents the i^{th} value in the test set. $Y_{i,1:1000}^{SIM}$ represents 1000 simulated values based on the covariate characteristics of Y_i^{Test}. $\left(X_{(1)}, X_{(n/4)}, X_{(n/2)}, X_{(3n/4)}, X_{(n)}\right)$ stand for the order statistics, where $X_{(1)}$ is the smallest and $X_{(n)}$ the largest value. The computed value t is the summed up difference between the observed value in the test set and the quantiles of the simulated values, hence a smaller t indicates a better fit.

Results of the tests

We constructed for each of the three distributions and four covariates 1000 times the paired Wilcoxon signed-rank test between one simulation on the test set and the observed test set values. For each of the $12 \cdot 1000$ tests we compute a p-value. In table 5.2 under 'Confidence intervals of p-values' we constructed the 95% confidence intervals of these p-values and performed twelve χ^2 tests for uniformity under 'P-value of χ^2 test' since the p-values follow a uniform distribution under H_0 [Murdoch et al., 2008, p. 243].

We observe that all the uniformity tests give tiny values and therefore indicate that the simulated values don't follow the same underlying distribution then the observed ones. We also observe that the confidence intervals of the quasi-Poisson contain the most tiny values indicating the

worst fit. The regular negative binomial seems to fit the data the best here. The histograms of the p-values are attached in the appendix as figure 12.

Table 5.2: 95 % confidence intervals (up) and the p-value of the χ^2 test for uniformity (bottom) of 1000 p-values from the paired Wilcoxon signed-rank test done through simulated values based on the test set.

Confidence intervals of p-values	Quasi-Poisson $\omega = 1,\ \theta = 0$	Negative Binomial $\omega = 1,\ \theta = \hat{\theta}$	Negative Binomial $\omega = \hat{\omega},\ \theta = \hat{\theta}$
Y ~ 1	[0.000, 0.505]	[0.015, 0.675]	[0.014, 0.661]
Y ~ Type of Terrorism	[0.001, 0.419]	[0.026, 0.620]	[0.014, 0.581]
Y ~ Event type	[0.002, 0.473]	[0.034, 0.650]	[0.016, 0.637]
Y ~ Weapons	[0.003, 0.560]	[0.049, 0.641]	[0.019, 0.662]

P-value of χ^2 test	Quasi-Poisson $\omega = 1,\ \theta = 0$	Negative Binomial $\omega = 1,\ \theta = \hat{\theta}$	Negative Binomial $\omega = \hat{\omega},\ \theta = \hat{\theta}$
Y ~ 1	0	0	0
Y ~ Type of Terrorism	0	0	0
Y ~ Event type	0	0	0
Y ~ Weapons	0	0	0

Table 5.3 shows the computed values for the quantile method. The method has been performed on the zero-truncated simulated and observed data. The test statistic represents the sum of the absolute difference between the observed values and the quantiles of the simulated values. Since small values indicate a better fit, we state that the quasi-Poisson performs the worst of the models. The regular negative binomial model performs best, despite not being able to cover the extreme tails of the underlying true distribution.

Table 5.3: Values of the Quantile method done through simulated values based on the test set.

	Quasi-Poisson $\omega = \hat{\omega},\ \theta = 0$	Negative Binomial $\omega = 1,\ \theta = \hat{\theta}$	Negative Binomial $\omega = \hat{\omega},\ \theta = \hat{\theta}$
Y ~ 1	44'177	13'884	13'842
Y ~ Type of Terrorism	28'758	12'945	15'560
Y ~ Event type	24'902	13'449	15'931
Y ~ Weapons	23'613	12'886	16'496

Chapter 6

Discussion

6.1 Difficulties with the implementation

Data

First we needed to filter the three types of terrorism used from our large data set (over 600'000 attacks worldwide). We did it through a key words search applied in the headline, the description or in the actor variable directly. This approach can lead to inaccuracies specially since many terror attacks have only a suspected perpetrator. Further are the boundaries of the terrorism types fluid, since many organisations are separatist but have also an extreme left or an extreme right mindset. Many terror attacks used multiple weapons, where we decided to categorise an attack based on the most threatening weapon used. Further do attacks included into the Global Terrorism Database [2020] and Jane's Information Group [2020] database not necessary contain the same information in terms of 'Weapons' 'Event type' or even 'Wounded & Killed'.

Space-Time Models

The main restriction for space-time models was its dimension of 10348:6561. Hence we were not able to investigate exciting methods such as Bayesian or Statistical Learning methods. An other issue was the rather small amount of actual attacks in the training and specially the test set compared to the gigantic space of possible regions and days to attack. The third problem was connected to the coding of the data. Terrorism data has a strong spatio-temporal correlation with many attacks happening on the same day and in the same region. Unfortunately we can not utilize that feature since we have daily data and the risk of an attack on the next day is tiny in comparison to the risk right after an attack on the same day, probably due to increased security forces. Hence the estimated parameter $\hat{\phi}$ is very close to one which indicates no temporal correlation in the model.

Hurdle Models with Overdispersion

The main restriction of the count models were the rather small amount of attacks with wounded or killed victims. The test set contained only 79 attacks with atleast one victim and 41 attacks with more than one. We further only had categorical covariates which in combination with the discrete nature of the target variable quickly led to multicollinearity. We had to recode the variable 'Event type' and 'Weapons' and were forced to estimate the models with only one covariate at the time. From a statistical point of view create categories such as 'Vehicle impact' as a weapon type a beautiful variance to the other categories with an extremely high expected amount of 'Wounded & Killed' (see table 3.3). Unfortunately are no attacks with a vehicle recorded that led to no wounded or killed victims, hence this 'Weapon' type creates no variance for the hurdle part and can therefore not be used as a category.

6.2 Conclusion

We found evidence that terror attacks are strongly spatially clustered, since the spatial neighbourhood matrix with a single parameter performed better than the gradient boosting method with spatial covariates. Therefore we also suspected a strong spatio-temporal cluster which we only found to some extent. The performance issues of the spatio-temporal model are most likely because of the coding of the time dimension, the large amount of possible attack spots over space and time, the relatively small amount of data used and restrictions in the model estimation process due to computational burdens. There further might be a change in the behaviour of terrorists away from spatio-temporal clustered organized attacks towards less organized single perpetrators as stated in Jenkins et al. [2016, p. 6]. We compared the amount of wounded and killed per attack with three hurdle count models with overdispersion. We found strong evidence that the linear mean-variance relationship from the quasi-Poisson does not model the variable well. The negative binomial model with a linear and a quadratic mean-variance relationship (NB2) performed slightly worse than the regular negative binomial (NB) regarding the goodness of fit tests which however struggles modelling the extreme values. A nice property of the NB2 is that it estimates the largest variances for categories with large estimated means and the smallest variances for categories with small estimated means. We can not recommend one of these two models but advise against the use of only a linear mean-variance relationship. We strongly recommend to further investigate how to model the threat of a terror attack. It requires more data points in order to construct more complex models with multiple covariates. We think that this can be achieved by expanding the types of terrorism or relaxing the geographical boundaries. An other interesting idea would be to combine these models with a simple spatial model.

References

Anselin, L. (2020). *Contiguity-Based Spatial Weights*. Retrieved 2020-11-03, from https://geodacenter.github.io/workbook/4a_contig_weights/lab4a.html

Bhaktha, N. (2018). *Properties of Hurdle Negative Binomial Models for Zero-Inflated and Overdispersed Count data* (Unpublished doctoral dissertation). The Ohio State University.

Center for International Earth Science Information Network. (2000). *Gridded Population of the World*. Columbia University, and Centro Internacional de Agricultura Tropical. Retrieved from https://doi.org/10.7927/H4XK8CG2 doi: 10.5860/choice.45-2621

Computational complexity of machine learning algorithms, T. (2018). *Computational complexity of machine learning algorithms*. Retrieved 2020-11-05, from https://www.thekerneltrip.com/machine/learning/computational-complexity-learning-algorithms/

Craig, C. M., Overbeek, R. W., & Niedbala, E. M. (2019). A Global Analysis of Temperature, Terrorist Attacks, and Fatalities. *Studies in Conflict and Terrorism*. Retrieved from https://www.tandfonline.com/doi/abs/10.1080/1057610X.2019.1606992 doi: 10.1080/1057610X.2019.1606992

Deléamont, P.-Y. (2020). Statistical Learning:Tree-based methods [PowerPoint presentation]. Université de Neuchâtel.

Europol. (2019). *European Union Terrorism Situation and Trend Report 2019* (Tech. Rep.). European Union Agency for Law Enforcement Cooperation.

Europol. (2020). *European Union Terrorism Situation and Trend Report 2020* (Tech. Rep.). European Union Agency for Law Enforcement Cooperation.

Global Terrorism Database. (2020). *GTD | Global Terrorism Database*. Retrieved from https://start.umd.edu/gtd/

Greenwell, B., Boehmke, B., Cunningham, J., & Developers, G. (2020). gbm: Generalized boosted regression models [Computer software manual]. Retrieved from https://CRAN.R-project.org/package=gbm (R package version 2.1.8)

Guo, W. (2019). Common statistical patterns in urban terrorism. *Royal Society Open Science*, *6*(9), 190645. doi: 10.1098/rsos.190645

Hao, M., Jiang, D., Ding, F., Fu, J., & Chen, S. (2019). Simulating Spatio-Temporal Patterns of Terrorism Incidents on the Indochina Peninsula with GIS and the Random Forest Method. *International Journal of Geo-Information*, *8*(133). doi: 10.3390/ijgi8030133

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2000). *An introduction to Statistical Learning* (Vol. 7) (No. 10). doi: 10.1007/978-1-4614-7138-7

Jane's Information Group. (2020). *Janes*. Retrieved from `www.janes.com`

Jenkins, M. B., Willis, H. H., & Han, B. (2016). *Do Significant Terrorist Attacks Increase the Risk of Further Attacks? Initial Observations from a Statistical Analysis of Terrorist Attacks in the United States and Europe from 1970 to 2013*.

Lindén, A., & Mäntyniemi, S. (2011). Using the negative binomial distribution to model overdispersion in ecological count data. *Ecology*, *92*(7), 1414–1421. doi: 10.1890/10-1831.1

Lumen. (n.d.). *The Germanic Tribes | Boundless World History*. Retrieved 2020-11-30, from `https://courses.lumenlearning.com/boundless-worldhistory/chapter/the-germanic-tribes/`

Martin, E., & Hine, R. (2008). *A Dictionary of Biology*. Oxford University Press. doi: 10.1093/acref/9780199204625.001.0001

Morris, S. A. (2016). *Spatial Methods for Modeling Extreme and Rare Events* (Unpublished doctoral dissertation). North Carolina State University.

Murdoch, D. J., Tsai, Y. L., & Adcock, J. (2008). P-values are Random Variables. *American Statistician*, *62*(3), 242–245. doi: 10.1198/000313008X332421

Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, *7*(DEC). doi: 10.3389/fnbot.2013.00021

NCSS. (n.d.). *PASS Sample Size Software Documentation Chapter 493, Paired Wilcoxon Signed-Rank Tests* (Tech. Rep.). Author. Retrieved from `https://www.ncss.com/software/pass/pass-documentation/`

Porter, M. D., & White, G. (2010). Self-exciting hurdle models for terrorist activity. *Annals of Applied Statistics*, *4*(1), 106–124. doi: 10.1214/11-AOAS513

R Core Team. (2020). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `https://www.R-project.org/`

Rodriguez, G. (2013). Models for Count Data With Overdispersion. *Princeton Statistics*(September 2007), 1–7.

Smith, T. E. (2020). *Spatial weight matrices* (Tech. Rep.). Retrieved from https://www.seas.upenn.edu/$\sim$ese502/lab-content/extra_materials/SPATIALWEIGHTMATRICES.pdf

Tolan, G. M., & Soliman, O. S. (2015). An experimental study of classification algorithms for terrorism prediction. *International Journal of Knowledge Engineering*, *1*(2), 107–112.

Uddin, M. I., Zada, N., Aziz, F., Saeed, Y., Zeb, A., & Ali Shah, S. A. (2020). Prediction of Future Terrorist Activities Using Deep Neural Networks. *Complexity*, *2020*. doi: 10.1155/2020/1373087

Ver Hoef, J. M., & Boveng, P. L. (2007). Quasi-Poisson vs. negative binomial regression: How should we model overdispersed count data? *Ecology*, *88*(11), 2766–2772.

Weisstein, E. W. (1984). Poisson Distribution – from Wolfram MathWorld Poisson Distribution – from Wolfram MathWorld. (21), 21–23. Retrieved from https://mathworld.wolfram.com/PoissonDistribution.html

Welsh, A. H., Cunningham, R. B., Donnelly, C. F., & Lindenmayer, D. B. (1996). Modelling the abundance of rare species: Statistical models for counts with extra zeros. *Ecological Modelling*, *88*(1-3), 297–308. doi: 10.1016/0304-3800(95)00113-1

Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, *1*(6), 80. doi: 10.2307/3001968

WorldClim.org. (2000). *Global climate and weather data — WorldClim 1 documentation*. Retrieved 2020-11-04, from https://worldclim.org/data/index.html

Appendices

Appendix A: Derivations

The Poisson distribution is a special case of the binomial distribution when the number of trials N goes to infinity while the expected number of successes n remains fixed. The probability to obtain exactly n successes in N trials, modelled through a binomial distribution is:

$$P(n|N) = \frac{N!}{n!(N-n)!}\,\pi^n(1-\pi)^{(N-n)} \tag{1}$$

Viewing the distribution as a function of the expected number of successes instead of the sample size N:

$$\lambda \equiv N\,\pi \tag{2}$$

For fixed π equation (1) becomes:

$$P(n|N) = \frac{N!}{n!(N-n)!}\,\frac{\lambda}{N}^{n}\,(1-\frac{\lambda}{N})^{(N-n)} \tag{3}$$

Now let the sample size N go to ∞:

$$P(n) = \lim_{N\to\infty} \frac{N!}{n!(N-n)!}\,\frac{\lambda}{N}^{n}\,(1-\frac{\lambda}{N})^{(N-n)} \tag{4}$$

$$= \lim_{N\to\infty} \frac{N!}{n!(N-n)!}\,\frac{\lambda^n}{N^n}\,(1-\frac{\lambda}{N})^{N}\,(1-\frac{\lambda}{N})^{-n} \tag{5}$$

$$= \lim_{N\to\infty} \frac{N!}{(N-n)!\,N^n}\,\frac{\lambda^n}{n!}\,(1-\frac{\lambda}{N})^{N}\,(1-\frac{\lambda}{N})^{-n} \tag{6}$$

$$= 1 \cdot \frac{\lambda^n}{n!} \cdot e^{-\lambda} \cdot 1 \tag{7}$$

$$= \frac{\lambda^n\,e^{-\lambda}}{n!} \tag{8}$$

[Weisstein, 1984]

Appendix B: Tables & Pictures

Analysis Data

Table 1: Cross-validation (CV) value to minimise the parameter optimization procedure for the Spatio-Temporal Model

$\hat{\rho}$	$\hat{\delta}$	$\hat{\phi}$	CV-Value	$\hat{\rho}$	$\hat{\delta}$	$\hat{\phi}$	CV-Value
0.1	0.1	0.99	0.9231	0.01	0.01	0.70	1.4983
0.01	0.1	0.99	0.9231	0.01	0.01	0.75	1.4913
0.01	1	0.99	0.9231	0.01	0.01	0.65	1.5043
0.01	0.01	0.99	0.9231	0.01	0.01	0.55	1.5141
0.08	0.1	0.99	0.9231	0.01	0.10	0.45	1.5214
0.1	10	0.99	0.9231	0.01	0.10	0.35	1.5268
0.001	0.1	0.99	0.9231	0.0001	0.1	0.25	1.5307
0.2	0.1	0.99	0.9236	0.01	0.1	0.25	1.5307
0.2	0.1	0.99	0.9236	0.01	0.1	0.15	1.5336
0.01	0.01	0.95	1.4037	0.5	10	0.99	11.5467
0.01	0.01	0.85	1.4725	0.5	10	0.75	11.9418

Exploratory Tables

Table 2: Terror Organisations in the Analysis Dataset

Terror organisation	Amount of Attacks	Type of Terrorism	Organisation Country
Al-Qaeda	9	Islamic	International
Ansar al-Islam	1	Islamic	Iraq
Armed Islamic Group (GIA)	10	Islamic	International
French Armed Islamic Front	1	Islamic	Indonesia
Hizbullah	1	Islamic	Lebanon
Hofstad Network	1	Islamic	Netherlands
Islamic Golden Army	1	Islamic	Algeria
Islamic State	24	Islamic	Iraq
Anarchist Faction	2	Left Terror	Italy & Jordan
Anarchist Liberation Brigade	1	Left Terror	Greece
Anarchist Squad	1	Left Terror	Greece
Anarchist Struggle	1	Left Terror	Serbia-Montenegro
Anarchists	20	Left Terror	International
Angry Brigade (Italy)	2	Left Terror	Italy
Animal Liberation Front (ALF)	10	Left Terror	International
Anonymous	3	Left Terror	International
Anti-State Proletarian Nuclei	1	Left Terror	Albania
Basque Fatherland and Freedom (ETA)	1	Left Terror	Spain
Black Star	6	Left Terror	International
CCCCC	4	Left Terror	Italy & Spain
Earth Liberation Front (ELF)	2	Left Terror	International
Epanastatikos Agonas (EA)	12	Left Terror	Greece
Erotic Anti-authority Cells	1	Left Terror	United States
Federazione Anarchica Informale (FAI)	26	Left Terror	Italy
Greek Anarchists Union	2	Left Terror	Greece
Grey Wolves	4	Left Terror	Italy
Hekla Reception Committee Initiative	8	Left Terror	Germany
Individualidades Tendiendo a lo Salvaje	4	Left Terror	Mexico
Informal Anarchist Federation	5	Left Terror	International
International Solidarity	1	Left Terror	Italy
Irish National Liberation Army (INLA)	31	Left Terror	International
Last Generation	1	Left Terror	Greece
Left-Wing	8	Left Terror	International
May 98	2	Left Terror	Greece
Omáda Laïkón Agonistón (OLA)	4	Left Terror	Greece
Popular Will	1	Left Terror	Greece
Proletarian Nuclei for Communism	1	Left Terror	Italy
Provisional IRA (PIRA)	1	Left Terror	United Kingdom
Republican Action Against Drugs (RAAD)	9	Left Terror	United Kingdom
Revolted Persons of the Polytech School	1	Left Terror	Greece
Revolutionary Nuclei	1	Left Terror	Greece
Revolutionary Perspective	1	Left Terror	Switzerland
Revolutionary Popular Left	3	Left Terror	Greece & Netherlands
Revolutionary Proletarian Initiative Nuclei	2	Left Terror	Italy
Revolutionary Violence Units	1	Left Terror	Israel
Rouvikonas	15	Left Terror	Greece
Sekta Epanastaton	3	Left Terror	Greece
Synomosia Pyrinon tis Fotias (SPF)	28	Left Terror	Greece
Turkish Leftists	8	Left Terror	Turkey
Actiefront Nationalistisch Nederland	1	Right Terror	Netherlands
Animal Rights extremists	13	Right Terror	International
Ariska Brodraskapet (Aryan Brotherhood)	1	Right Terror	Sweden
Danish Neo-Nazi Group	1	Right Terror	Great Britain
Neo-Nazi extremists	183	Right Terror	International
Right-wing extremists	12	Right Terror	International
Right-Wing Youths	2	Right Terror	Spain
White supremacists/nationalists	1	Right Terror	Afghanistan
Unknown	51	Islamic	
Unknown	194	Left Terror	
Unknown	31	Right Terror	

Histograms of the Data

Figure 1: Histogram of 'Wounded & Killed' per terror attack split for 'Types of Terrorism'

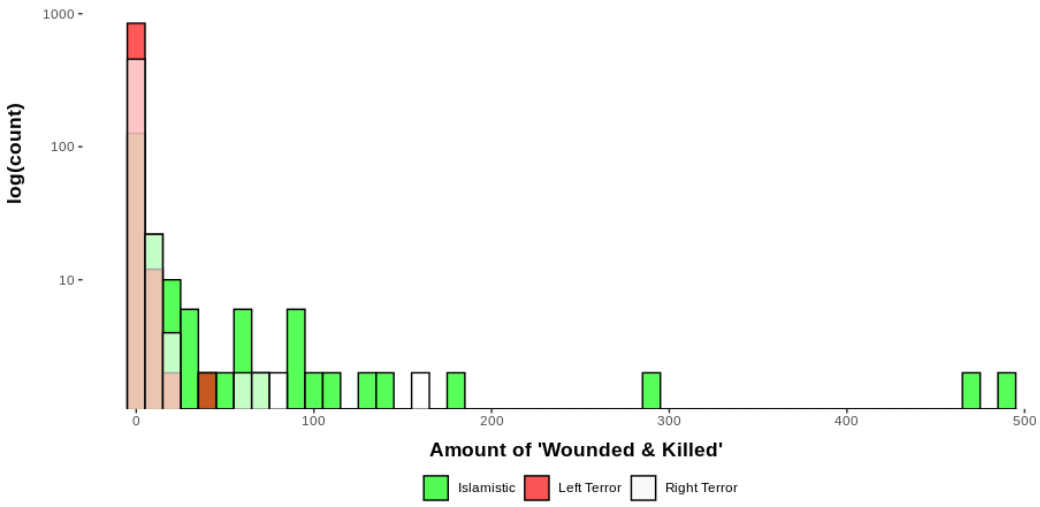


Figure 2: Histogram of 'Wounded & Killed' per terror attack split for recoded 'Event type'

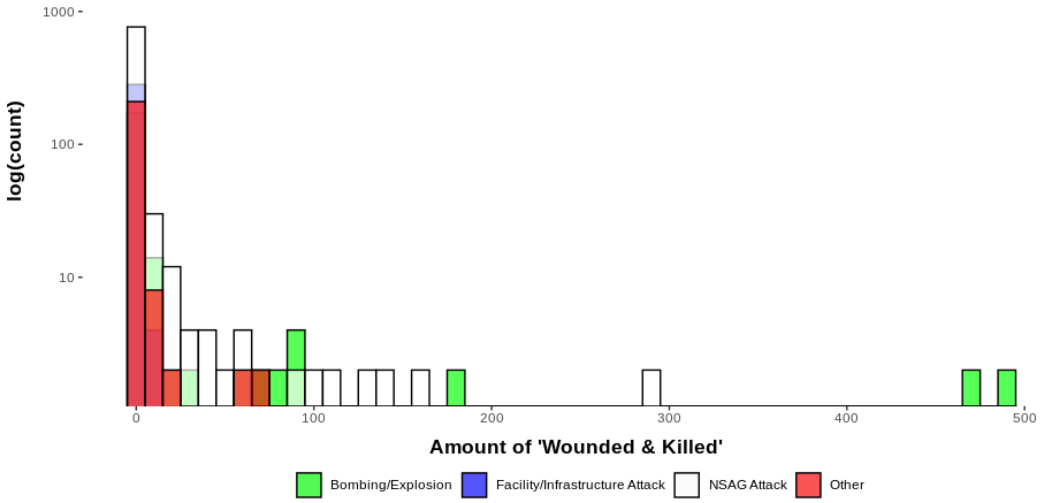
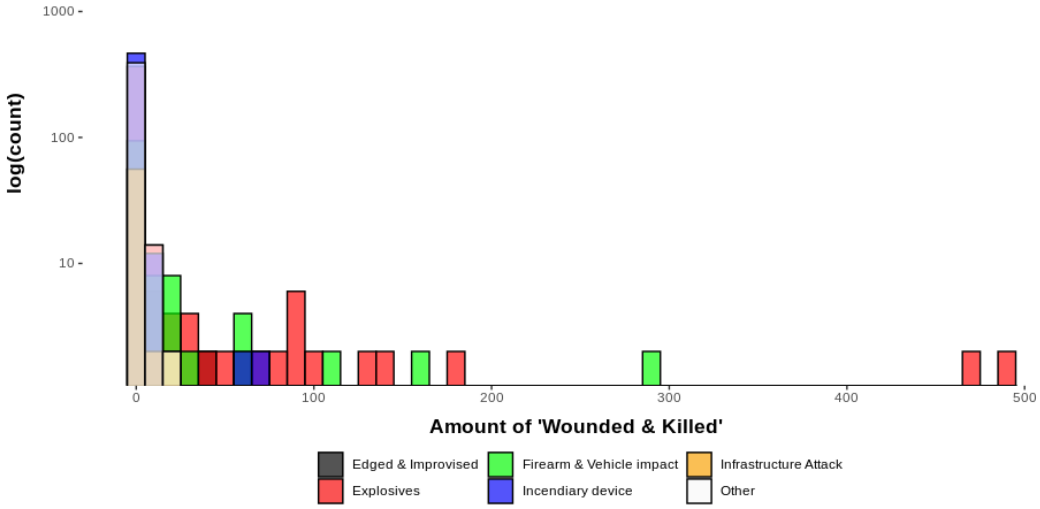


Figure 3: Histogram of 'Wounded & Killed' per terror attack split for recoded 'Weapons'



Histograms of the Simulated Values

Figure 4: Histograms of the relative amount of two's per training sample for three different models using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

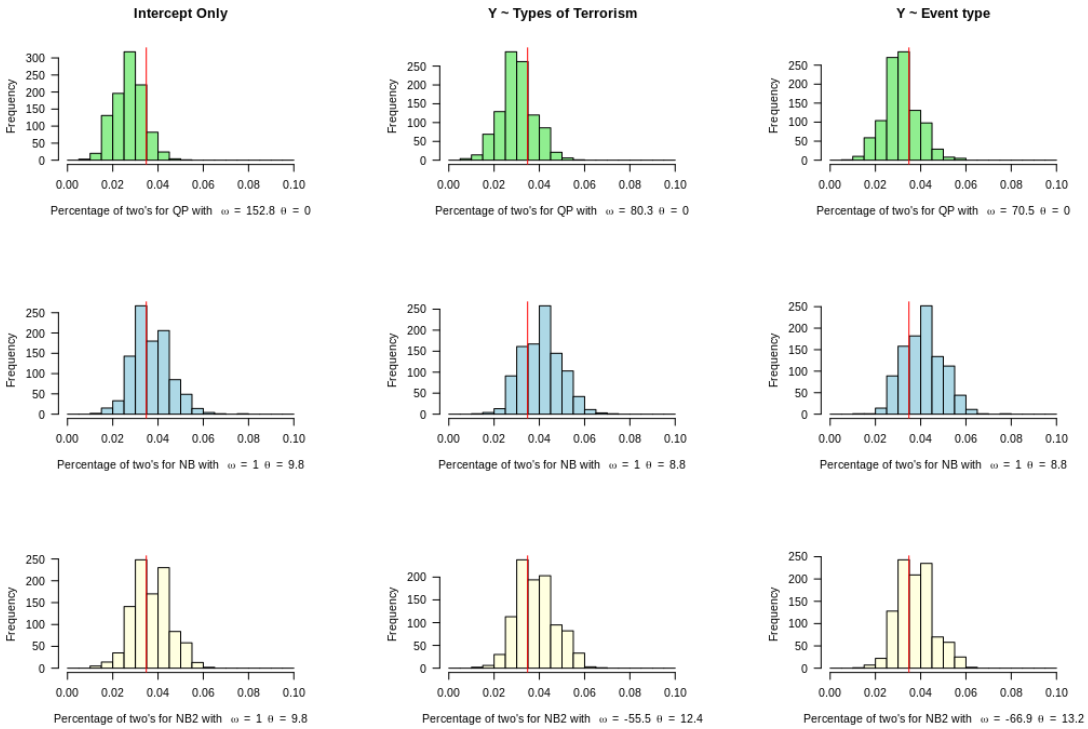


Figure 5: Histograms of the relative amount of three's per training sample for three different models using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

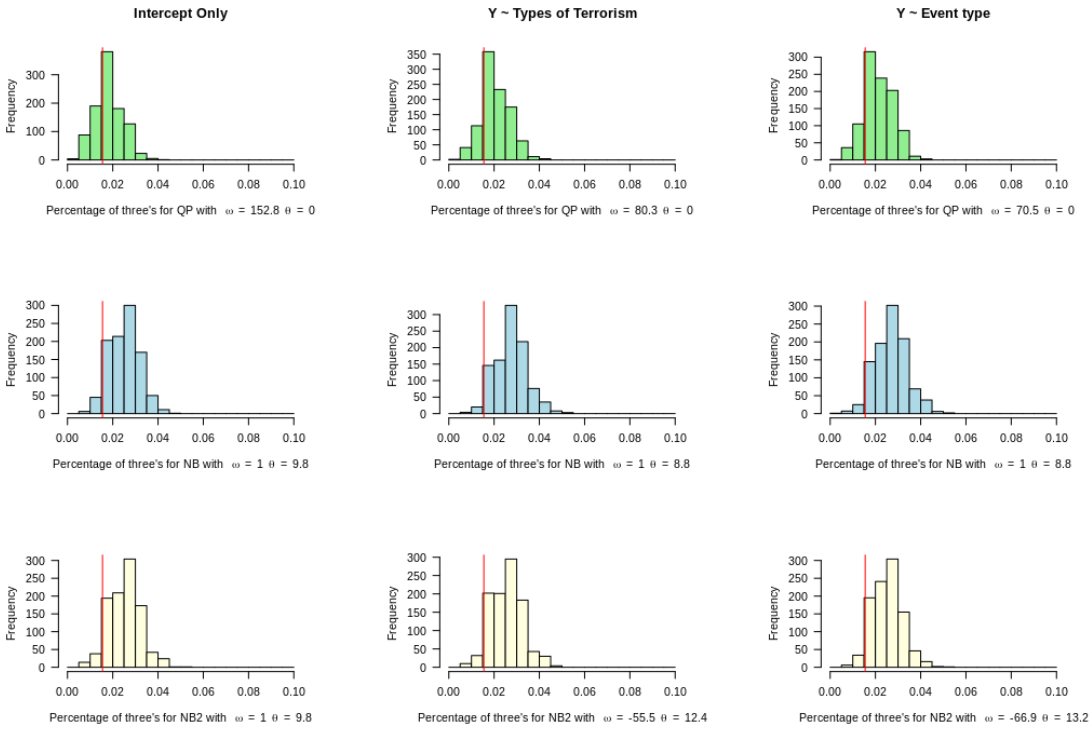


Figure 6: Histograms of the relative amount of four's per training sample for three different models using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

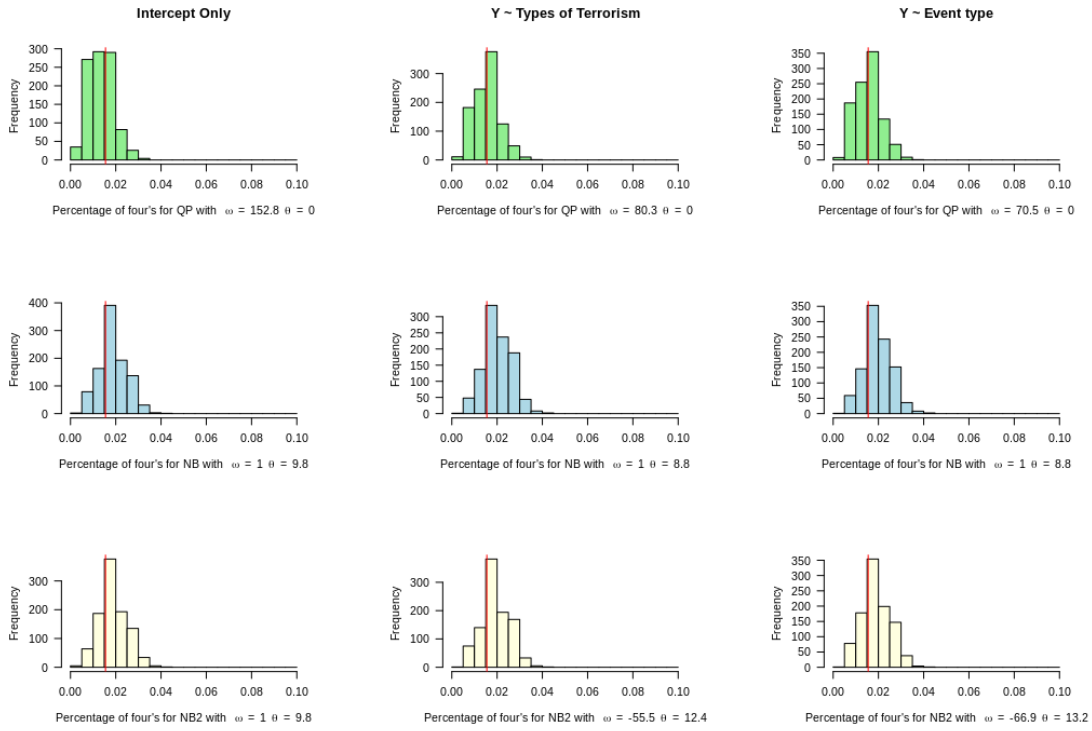


Figure 7: Histograms of maximum value per training sample for the model **Y** ~ **Weapons** using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

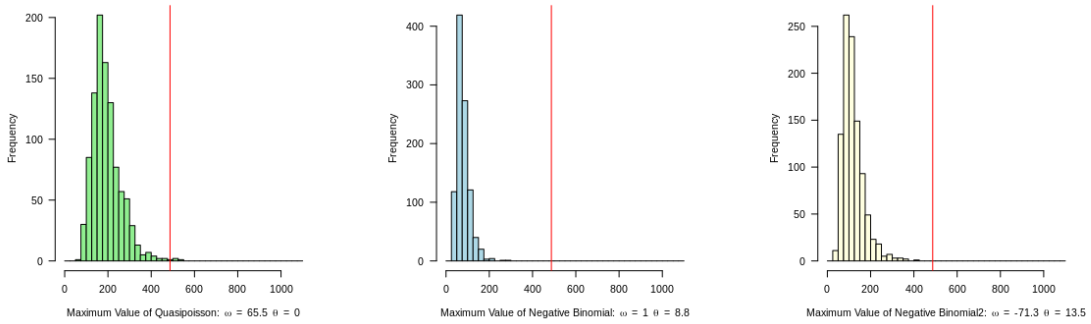


Figure 8: Histograms of the relative amount of of one's per training sample for the model **Y** ~ **Weapons** using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

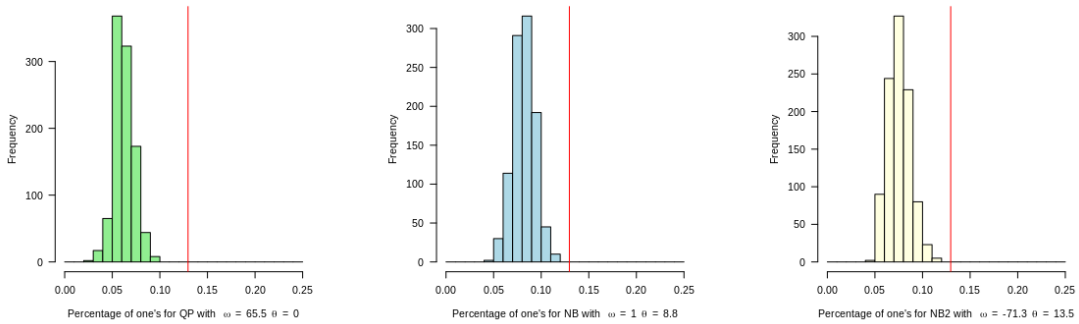


Figure 9: Histograms of the relative amount of values between 10 and 100 per training sample for the model **Y ~ Weapons** using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

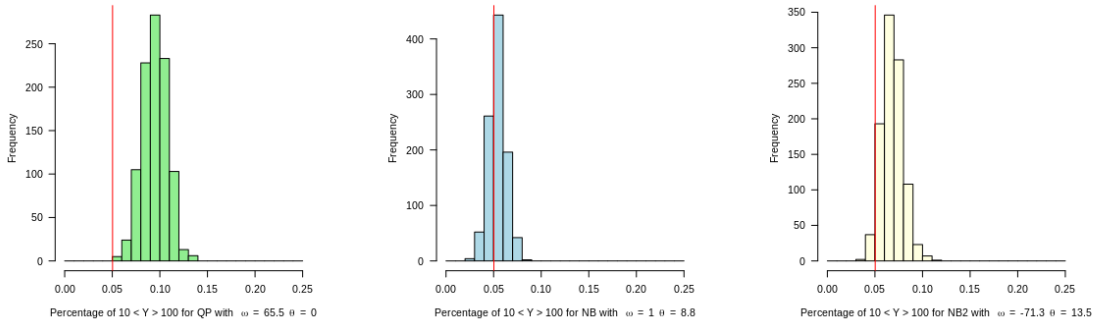


Figure 10: Histograms of the relative amount of of two's per training sample for the model **Y ~ Weapons** using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

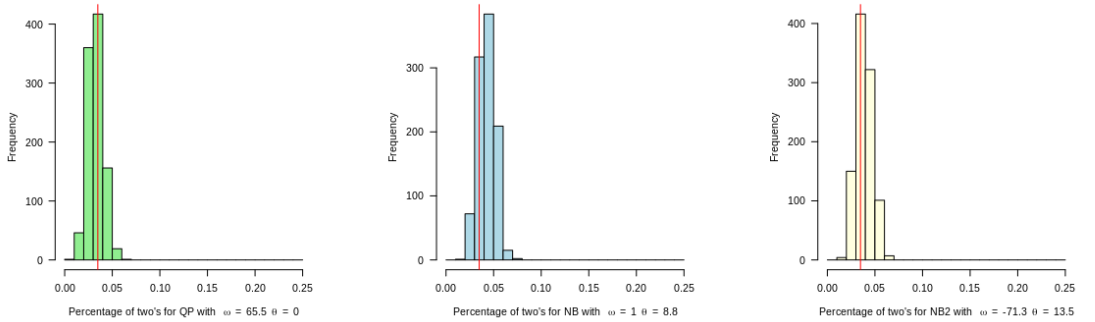


Figure 11: Histograms of the relative amount of of three's per training sample for the model **Y** ~ **Weapons** using the quasi-Poisson, the negative binomial and the double parametrized negative binomial

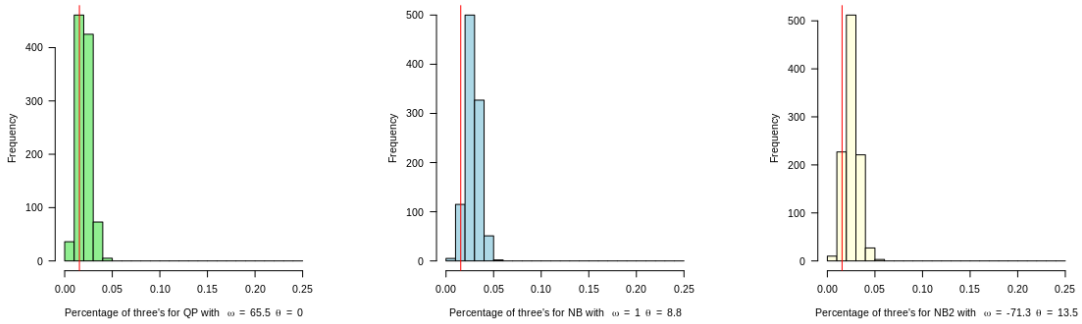
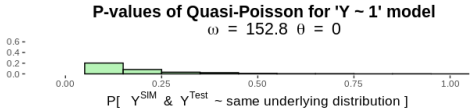
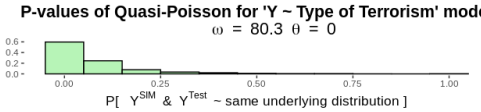
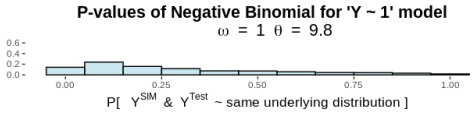
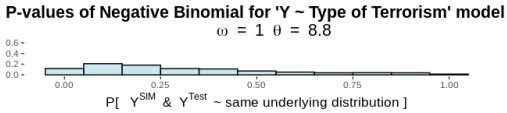


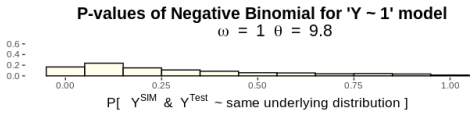
Figure 12: Histograms of the p-values of the paired Wilcoxon signed-rank test

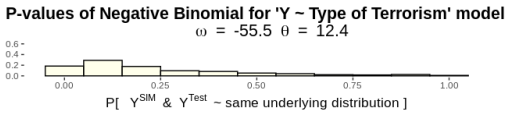






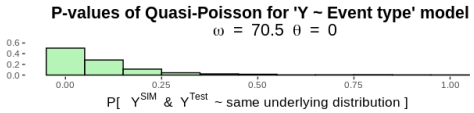


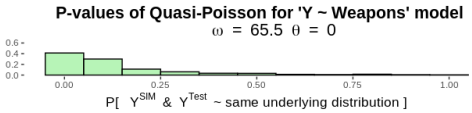


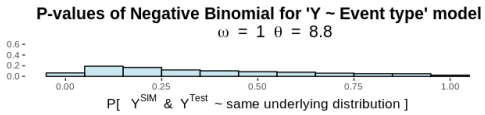


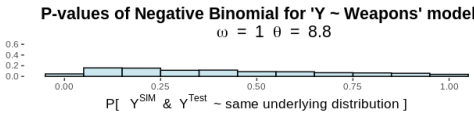
(a) **'Intercept only'** models

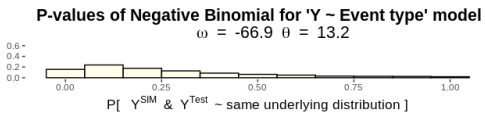
(b) **'Type of Terrorism'** models

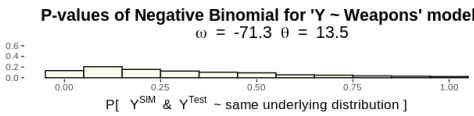












(c) **'Event type'** models

(d) **'Weapons'** models

Whole Data

Table 3: Summary Statistics of the Variable Wounded & Killed for the three types of terrorism for the whole world and whole time line (1970-2020).

Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	n
Islamic Terror	5.04	620.2	0-2461	99.6	76523
Right Terror	3.45	218.9	0-228	63.5	634
Left Terror	1.89	45.7	0-347	24.1	10856

Table 4: Summary Statistics of the Variable Wounded & Killed per country for the whole world and the whole time line (1970-2020), for the three types of terrorism.

Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	n
Afghanistan	5.46	190.3	0:412	34.9	18260
Albania	0.00	-	0:0.0	-	1
Algeria	6.52	405.8	0:329	62.3	1364
Argentina	10.04	2396.4	0:250	238.7	26
Australia	0.44	2.6	0:9	5.9	45
Austria	0.41	0.9	0:4	2.2	22
Azerbaijan	0.39	2.0	0:6	5.2	18
Bahrain	1.67	6.3	0:7	3.8	9
Bangladesh	5.22	323.8	0:260	62.0	1092
Belarus	0.00	0.0	0:0	-	5
Belgium	11.14	1112.7	0:131	99.9	22
Bhutan	1.00	2.0	0:2	2	2
Bolivia	0.33	0.3	0:1	0.8	9
Bosnia and Herzegovina	1.83	4.2	0:4	2.3	6
Brazil	0.13	0.3	0:2	2	15
Bulgaria	41.00	-	41:41	-	1
Burkina Faso	10.55	329.9	0:70	31.3	20
Burundi	0.00	-	0:0	-	1
Cameroon	6.32	315.9	0:164	50.0	499
Canada	0.90	7.7	0:14	8.6	29
Chad	20.58	1125.3	0:152	54.7	62
Chile	0.42	2.3	0:14	5.5	345
China	9.19	604.1	0:133	65.7	36

Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	n
Colombia	2.66	30.6	0:90	11.5	2270
Congo, The Democratic Republic of	8.00	50.0	3:13	6.2	2
Côte d'Ivoire	27.50	1512.5	0:55	55	2
Croatia	3.00	-	3:3	-	1
Cuba	0.00	0.0	0:0	-	14
Cyprus	2.00	8.0	0:4	4	2
Czech Republic	0.20	0.2	0:1	0.9	9
Czechoslovakia	1.00	-	1:1	-	1
Democratic Republic of Congo	0.0	-	0:0	-	1
Denmark	0.95	3.8	0:7	4	19
Djibouti	0.00	0.0	0:0	-	3
East Germany (GDR)	0.5	0.5	0:1	1	2
Ecuador	4.50	73.6	0:28	16.4	10
Egypt	6.30	570.1	0:463	90.5	701
El Salvador	2.98	37.2	0:23	12.5	50
Eritrea	0.00	0.0	0:0	-	6
Ethiopia	1.71	12.9	0:13	7.5	17
Finland	3.33	33.3	0:10	10.0	3
France	7.51	848.6	0:288	113.0	134
Gaza and the West Bank	0.5	10.63	0:85	20.5	1915
Georgia	2.69	23.8	0:17	8.9	16
Germany	1.47	48.4	0:74	32.9	317
Greece	0.57	31.3	0:90	55.1	266
Guatemala	8.35	328.9	0:88	39.4	26
Guinea-Bissau	0.0	-	0:0	-	1
Honduras	0.00	-	0:0	-	1
Hungary	0.00	-	0:0	-	1
India	1.80	281.1	0:906	155.8	7239
Indonesia	6.67	807.9	0:251	121.1	254
Iran	4.90	542.8	0:197	110.8	88
Iran, Islamic Republic of	7.6	529.48	0:199	69.7	183
Iraq	8.04	1831.4	0:2000	227.9	9965
Ireland	0.58	0.3	0:1	0.5	12
Israel	5.45	391.3	0:185	71.7	814
Italy	0.29	0.4	0:3	1.2	189
Jamaica	1.00	0.0	1:1	0.0	2
Japan	0.23	0.4	0:2	1.6	13
Jordan	2.09	60.8	0:41	29.1	34
Kazakhstan	0.00	0.0	0:0	-	2

Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	n
Kenya	0.56	2.7	0:11	4.8	100
Democratic People's Republic of Korea	0.0	-	0:0	-	1
Southkorea	0.00	0.0	0:0	-	3
Kosovo	0.00	0.0	0:0	-	2
Kuwait	27.55	5728.7	0:255	208	11
Kyrgyzstan	0.58	1.9	0:6	3.3	33
Lao, People's Democratic Republic of	0.0	-	0:0	-	1
Lebanon	6.22	1397.9	0:547	224.6	504
Libya	9.94	602.5	0:215	60.6	379
Madagascar	0.00	-	0:0	-	1
Malaysia	0.26	2.1	0:8	8	31
Maldives	10.29	263.6	0:34	25.6	7
Mali	5.14	243.4	0:193	47.3	256
Mauritania	0.28	1.8	0:12	6.6	101
Mauritiania	4.09	53.5	0:24	13.1	11
Mexico	0.68	4.3	0:9	6.3	19
Montenegro	0.00	-	0:0	-	1
Morocco	1.72	58.7	0:36	34.1	43
Mozambique	13.19	585.2	0:72	44.4	16
Myanmar	1.60	12.8	0:8	8	5
Nepal	2.67	31.3	0:60	11.7	366
Netherlands	0.27	0.6	0:3	2.4	15
Netherlands Antilles	0.0	-	0:0	-	1
New Zealand	22.50	2025.0	0:90	90	4
Nicaragua	1.75	12.3	0:7	7	4
Niger	13.03	952.6	0:241	73.1	146
Nigeria	12.99	696.5	0:300	53.6	1536
Norway	17.31	1979.4	0:164	114.3	16
Oman	0.00	-	0:0	-	1
Pakistan	5.03	468.6	0:432	93.2	8605
Palestinian Territory, Occupied	37.25	88994.3	0:2461	2389.1	68
Panama	2.00	2	1:3	1	2
Paraguay	1.38	4.35	0:9	3.2	74
Peru	2.60	10.16	0:16	3.9	93
Philippines	2.99	61.65	0:190	20.6	3094
Poland	1.25	6.25	0:5	5	4
Portugal	3.00	19	0:8	6.3	3
Qatar	0.00	0.0	0:0	-	4
Romania	0.00	-	0:0	-	1

Variable: Wounded & **Killed**	μ	σ^2	Range	$\frac{\sigma^2}{\mu}$	n
Russia	34.56	2683.56	0:170	77.7	18
Russian Federation	2.01	120.27	0:217	59.8	3985
Saint Vincent and the Grenadines	0.00	-	0:0	-	1
Saudi Arabia	3.23	167.5	0:102	51.9	136
Senegal	0.00	0.0	0:0	-	3
Serbia	0.00	0.0	0:0	-	2
Sierra Leone	0.00	-	0:0	-	1
Singapore	0.00	0.0	0:0	-	6
Slovak Republic	1.00	2.0	0:2	2.0	2
Somalia	4.43	221.67	0:231	50.0	2079
South Africa	5.31	361.15	0:99	68.0	29
Spain	18.92	6536.02	0:487	345.5	72
Sri Lanka	109.00	83167	0:763	763.0	7
Sudan	1.12	3.6	0:5	3.2	8
Sweden	1.53	19.4	0:19	12.7	19
Switzerland	0.00	0.0	0:0	-	10
Syria	4.19	203.6	0:65	48.6	21
Syrian Arab Republic	3.1	343.58	0:1785	109.3	15913
Tajikistan	4.35	113.2	0:40	26	92
Tanzania, United Republic of	0.0	-	0:0	-	1
Thailand	2.94	40.7	0:45	13.9	112
Tunisia	11.12	1044.2	0:200	93.9	49
Turkey	6.41	978.2	0:346	152.7	311
Turkmenistan	3.00	-	3:3	-	1
Uganda	33.21	3573.2	0:143	107.6	29
Ukraine	0.00	0.0	0:0	-	10
United Arab Emirates	0.0	0.0	0:0	-	13
United Kingdom	2.6	235.24	0:179	89.5	294
United States	0.5	24.57	0:103	46.6	690
Uruguay	2.00	-	2:2	-	1
Uzbekistan	5.92	114.9	0:43	19.4	24
Venezuela	2.25	20.3	0:9	9.0	4
Venezuela, Bolivarian Republic of	4.17	37.4	0:13	9	6
West Bank and Gaza Strip	4.4	59.35	0:48	13.6	165
West Germany (FRG)	17.5	3998.77	0:228	228.0	13
Yemen	4.91	329.9	0:486	67.2	1737
Yugoslavia	0.50	0.5	0:1	1.0	2
Zimbabwe	1.00	-	1:1	-	1

Figure 13: Amount of 'Wounded & Killed' for the three types of terrorism for the whole data set on the $\log_{10}$ scale.

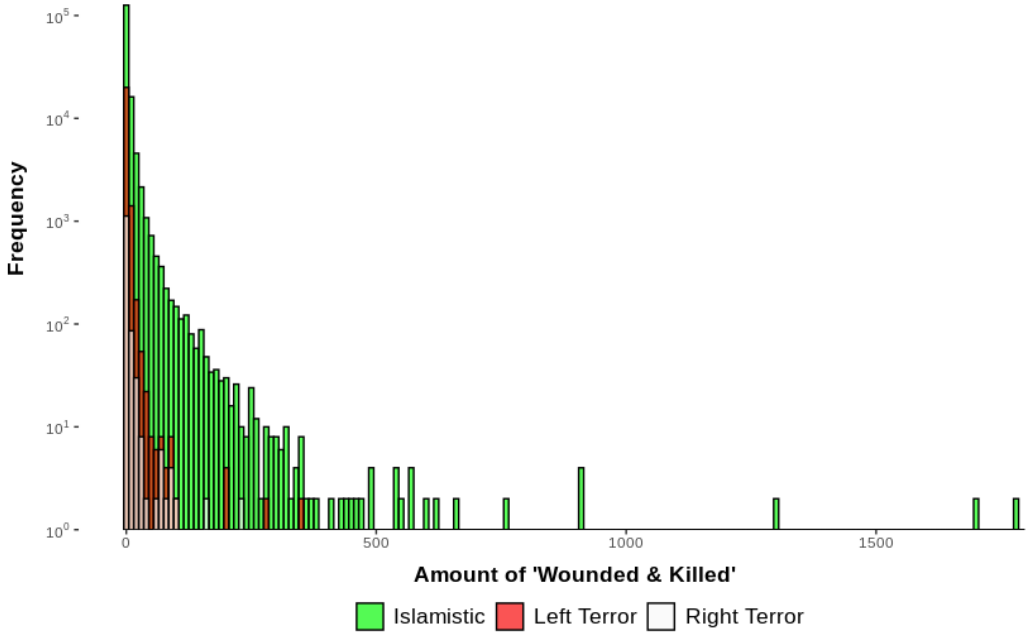
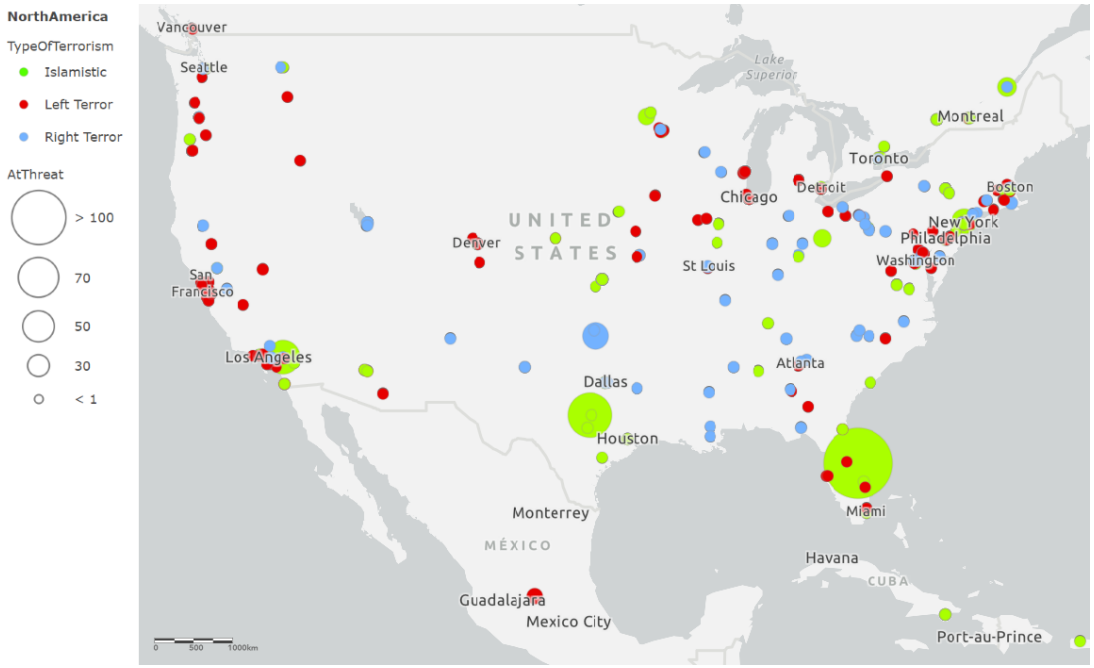


Figure 14: ArcGIS Map of North America for the variable 'Wounded & Killed' and the three types of terrorism



Source: Esri, HERE, Garmin, FAO, NOAA, USGS, EPA, and the GIS User Community

Appendix C: R-code

Iteratively weighted least squares

```
IWLS <- function(Omega1,Theta1,tollvl=1,X=X){
X <- as.matrix(X)
mu <- rep(mean(Y),length(X[,1]))
Omega <- Omega1 # Initial Omega
Theta <- Theta1 # Initial Theta
Y.head<-Y
tol <- 10 # Initial tolerance level
i <- 1 # counter

while(tol>tollvl){ # Iterative
W<-diag(c(mu/(Omega[i] + Theta[i]*mu)) )
# weighted matrix, dispersion parameter decide the distribution used
# Theta = 0 -> QuasiPoisson, Omega = 1 -> Negbin,
# both unfixed -> Negbin2 (reason wrote this code)
Beta.head<-solve(t(X)%*%W%*%X,t(X), tol=9^-45)%*%W%*%Y.head # blue
nu <- X%*%Beta.head
mu <- exp(nu)
Y.head <- nu + (Y-mu)/mu # derivative of log(nu) -> 1/nu
if(Theta1==0){ # Quasi-Poisson Case
    Theta <- c(Theta,0)
    Omega <- c(Omega,mean( (Y-mu)^2/mu) )
} else{
    if(Omega1==1){ # Negative binomial Case
    Theta<- c(Theta,mean( ((Y-mu)^2 - mu )/mu^2 ) )
    Omega <- c(Omega,1)
    } else{ # Negative binomial with 2 overdispersion parameters
    ThetaF  <- mean( (Y-mu)^2 - mu )/(mean(mu)^2)
    Omega <- c(Omega,mean( ((Y-mu)^2)/mu - mu*ThetaF) )
    Theta  <- c(Theta,mean( ((Y-mu)^2 - mu*Omega[i+1] )/(mean(mu)^2 )) )
    }
}
tol <- abs(Omega[i+1]-Omega[i])+abs(Theta[i+1]-Theta[i])
i <- i+1 # counter +1
}
list(Omega=Omega,
     Theta=Theta,
     Beta.head=Beta.head,
     SE = sqrt(diag(solve(t(X)%*%W%*%X))),
     FittedValues = mu)
}
```

Sampling function for overdispersed hurdle models

```
# Theta in rnbinom() is 1/theta in the paper
Sample.Hurdle.Count <- function(SIM=1000, HurdleCoef, X, Beta, Theta=0, Omega=1){
# Total amount of simulated values is length(Data)*SIM
X <- as.matrix(X) # transform logodds into probabilities
logit2prob<-function(logit){
    odds <- exp(logit)
    prob <- odds / (1 + odds)
    return(prob)
}
H<-rbinom(length(X)*SIM,1,prob=logit2prob(X%*%HurdleCoef))
mu<-X%*%Beta
if(Omega>1&Theta==0){ # Quasi-Poisson
    Theta <-mu/(Omega-1) # inverse of function in paper
} else{
    if(Omega>1 & Theta>0){ # NB2
    Theta <- mu/(Omega-1+mu*Theta)
    # inverse of function in paper
} else{
    Theta <- rep(Theta,length(mu))
 }
}

# Sample fron rnbinom
mu <- rep(mu,SIM)
Theta <- rep(Theta,SIM)
Y.pred <-rnbinom(n = SIM*length(X[,1]), mu = mu, size =Theta )

while(sum(Y.pred==0)>0){ # rejection sampling
    Y.pred.new<-rnbinom(n = length(which(Y.pred==0)),
    mu = mu[which(Y.pred==0)], size =Theta[which(Y.pred==0)] )
    Y.pred[which(Y.pred==0)] <- Y.pred.new
}

Y.pred*H
}
```

Create Probability Matrix for Spatio-Temporal Model

```
SpatialTemporalProbabilityMatrix <- function(opt,nTime,
    nEvents,Event,smat,long.grid,lat.grid){ # opt = c(rho,delta,phi)
    Weighted.Prob.Mat2 <- array(0,c(6561,nTime)) # array dim=6561:10348

# Scaled Gaussian Kernel
W <- function(s,v,rho) exp( ( sum(abs(s-v)/rho))^2 /(-2) )

# Look at every attack
for(i in 1:nEvents){
    # Create Spatial Weight Matrix W per attack
    Space<- apply(smat,1,FUN = W, v=c(long.grid[i],lat.grid[i]),rho=opt[1])

    # Assign W to the day after the attack
    Weighted.Prob.Mat2[,(Event[i]+1)] <- c(exp(-1/opt[2])*t(Space))
}

# AR(1) part
for(j in 1:nTime) Weighted.Prob.Mat2[,j] <-  Weighted.Prob.Mat2[,j]+
    Weighted.Prob.Mat2[,max(1,(j-1))] * opt[3]

# normalise
Weighted.Prob.Mat2<-Weighted.Prob.Mat2/sum(c(Weighted.Prob.Mat2),na.rm = T)
}
```

Monte Carlo Simulation for Spatio-Temporal Model

```
SpatioTempMC<-function(SIM, latlong, Events.Date, Train.ProbMat.Vec){
## Create event/attack matrix from test set
# time = 1:10348
SpatioTemporalEvent.Test.Matrix<-matrix(0,nrow=6561,ncol=length(time))

for(i in 1:length(Test)){ # 259 iterations
   # latlong and Events.Date = grids and date of attacks
   SpatioTemporalEvent.Test.Matrix[latlong[i],Events.Date[i]]
       <- SpatioTemporalEvent.Test.Matrix[latlong[i],Events.Date[i]]+1
}

SpatioTemporalEvent.Test.Vector<-c(SpatioTemporalEvent.Test.Matrix)
Test.Event.SpatioTemp<-which(!SpatioTemporalEvent.Test.Vector==0)

# Only time dimension
Obs_Attacks_PerDay <- colSums(SpatioTemporalEvent.Test.Matrix)
Test.Number <- which(Obs_Attacks_PerDay>0)
```

```
n.days<-length(which(Obs_Attacks_PerDay>0))

## Manipulate probability matrix (remove NAs and zero probabilities)
MC<-numeric(length(Train.ProbMat.Vec)) # length = 67'000'000
Train.ProbMat.Vec[which(Train.ProbMat.Vec==0)]<-NA
Train.ProbMat.Vec.NAOMIT <- na.omit(Train.ProbMat.Vec)
n.SIM<-length(Train.ProbMat.Vec.NAOMIT)    # 14'000

### Simulate
R_Time_and_Space <- numeric(SIM)
R_Space_fixed <- numeric(SIM)
R_Space_fixed_time_relaxed_1 <- numeric(SIM)

for(i in 1:SIM){ # simulate the 259 obs from the test set SIM times
   MC.NAOMIT<-rep(0,length(Train.ProbMat.Vec.NAOMIT))
   MCMC <- sample(n.SIM,length(Test),p= c(Train.ProbMat.Vec.NAOMIT))
   MC.NAOMIT[MCMC] <- 1
   MC[!is.na(Train.ProbMat.Vec)] <- MC.NAOMIT # dim is 6561:10'348
   Sim_Attacks_PerDay <- colSums(matrix(MC,ncol=10348)) # sum up per day

   ### Compare
   DaysAttackSim<- which(Sim_Attacks_PerDay>0)

   # Time and Space
   R_Time_and_Space[i]<-
   sum(duplicated(c(which(MC==1),Test.Event.SpatioTemp)))/length(Test)

   # Time
   R_Space_fixed[i] <-
   length(unique(which(abs(outer(Test.Number, DaysAttackSim, `-`))
        <= 0, arr.ind = TRUE)[,1]))/min(n.days,length(DaysAttackSim))

   # Time with +/- one day
   R_Space_fixed_time_relaxed_1[i] <-
   length(unique(which(abs(outer(Test.Number,DaysAttackSim, `-`))
       <= 1, arr.ind = TRUE)[,1]))/min(n.days,length(DaysAttackSim))
}

list(Result_TS = R_Time_and_Space,
     Result_T  = R_Space_fixed,
     Result_T2 = R_Space_fixed_time_relaxed_1)
}
```

Optimization Procedure of the Spatio-Temporal Model

```
Optimise1<-function(opt){
nn<-length(Train)
CVn<-round(c(1,nn*0.25,nn*0.5,nn*0.75,nn),0) # 5 fold CV
Fit<-numeric(length(CVn)-1) # length 4
DateStart<-as.numeric(todayCoded)-min(as.numeric(todayCoded))+1

for(i in 1:(length(CVn)-1)){ # Loop 4 times
    Train.CV<-Train[-(CVn[i]:CVn[i+1])] # 3/4 of the train data
    LeftOut<-Train[(CVn[i]:CVn[i+1])] # 1/4 of the train data

 # Create Probability Matrix (see 'Create Probability Matrix for Spatio-Temporal
                              Model')

    ProbMat<-SpatialTemporalProbabilityMatrix(opt,
        nEvents =  length(Train.CV), nTime =  length(todayCoded),
        Event = DateNumeric[-(CVn[i]:CVn[i+1])], smat = s ,
        long.grid = long.grid[Train.CV], lat.grid = lat.grid[Train.CV])

    ProbMat.Vec<-c(ProbMat)

     # Build comparison matrix on left out values
     ControlMat.CV<-matrix(0,nrow=6561,ncol=length(todayCoded))
     latlong.Leftout<-(lat.grid[LeftOut]*81+long.grid[LeftOut]-81)
     Events.CV.Date <- DateNumeric[(CVn[i]:CVn[i+1])]-
         min(as.numeric(todayCoded))+1

     for(j in 1:length(LeftOut)){
         ControlMat.CV[latlong.Leftout[j],Events.CV.Date[j]] <-
             ControlMat.CV[latlong.Leftout[j],Events.CV.Date[j]]+1
     }
     ControlVec.CV<-c(ControlMat.CV)

     # Compare
     Fit[i]<-sum((ProbMat.Vec-ControlVec.CV)^2) # squared loss function
}
sum(Fit)
}
```