Interstate Conflict Modelling

Submitted in partial fulfillment of the requirements
for the course of

Case Study
in
Minor 'Applied Econometrics: A Big Data Experience for All'

Akif Baser 2604372 - m.a.baser@student.vu.nl
Stijn Lijnsvelt 2648687 - s.j.g.lijnsvelt@student.vu.nl
Syb Heringa 2647294 - s.j.j.heringa@student.vu.nl
Julie Gaile 2528374 - j.gaile@student.vu.nl

Under the kind guidance of

Ninoslav Malekovic
Chief Data Scientist at HCSS

Siem Jan Koopman
Francisco Blasques
Michiel Hofman
Quint Wiersema
Teaching professors

SCHOOL OF BUSINESS AND ECONOMICS
VRIJE UNIVERSITEIT AMSTERDAM
January, 2021

Abstract

The Auto Regressive Integrated Moving Average (ARIMA), Logit, Probit and the Poisson regression models were used for interstate conflict modelling to predict the probabilities of conflict for a given year. Militarized Interstate Disputes (MID) data from the Correlates of War project was analysed and model performance evaluated. In-sample probability predictions were made from 1816 until 2014. A 5-year forecast from 2015 until 2019 was made. The ARIMA model was found not suitable for the binary MID data set probability predictions. The Logit and the Probit models produced excellent results with a pseudo-r^2 between 0.2 and 0.4 for most countries. The Poisson model was found to sufficiently forecast the frequency of conflict and was used as input for out-of-sample predictions each year.

1 Introduction

The Hague Center for Strategic Studies (HCSS) showed keen interest in modelling militarized interstate conflict. There are conflicts all over the world and these conflicts influence many facets of everyday life and business. For this reason there is an incredible amount of value in understanding the variables that are correlated with these conflicts. HCSS is interested in the probability of these conflicts. The ultimate goal would be to create two models that can accurately predict whether a country will have a militarized conflict and whether a certain country will have a militarized conflict with a specific other country. This would not only require a lot of data on the conflicts themselves, but also on other explanatory variables. That way a model could utilize both the lagged values of the conflict data as well as other variables, such as GDP. This case study, however, is restricted to a very short timeframe, in which it is simply impossible to achieve a sophisticated model ready to be deployed as a product. For this reason the agreement with HCSS is that the focus of this case study would be on the autoregressive variables of whether there was a conflict and on the frequency of conflict, to create a "skeleton" model that can later be further developed and expanded upon. With the research question in mind: "What is the probability of having a conflict for a given year?". In Section 2 the data collection and preparation process will be discussed. Section 3 will present the models for analysis and in Section 4 the analysis of the results will be presented. In Section 5 the results will be discussed and in Section 6 a conclusion will be drawn with recommendations for future research and development of models for interstate conflict.

2 Data collection

In collaboration with HCSS, several data sets were collected with different variables, sizes and definitions of what a conflict is. The primary data set that was chosen for this case study was MID v5.0 from The Correlates of War project (COW). This data set contains information about militarized interstate disputes from 1816 to 2014. Jones et al. defined MID's in 1996[3]:

"Militarized interstate disputes are united historical cases of conflict in which the threat, display or use of military force short of war by one member state is explicitly directed towards the government, official representatives, official forces, property, or territory of another state. Disputes are composed of incidents that range in intensity from threats to use force to actual combat short of war".

Following this definition, the words "conflict" and "dispute" were used interchangeably, in both cases referring to MID's.

The variables in this data set included, but were not limited to: fatality level, precise fatalities, outcome, hostility level and the number of states on each side. Two different matrices were created. Both matrices were created with all countries on one axis and all observed years on the other axis. The column of the matrix could be treated as a time series. At first, with binary data where the value of the cell (i.e. NTH,1955) was 1 in case the Netherlands had a conflict in that year, and 0 otherwise. The data showed whether a certain country had a militarized dispute in a certain year.

The same process was repeated, but this time the cell value was the frequency of MID's for a certain country in a certain year. Appendix F contains the R code that was used to create these matrices. Excel was used to make some final adjustments to the data set and these are also found in the Appendix F.

2.1 Data preparation for Logit, Probit and Poisson models

New variables were made by shifting the time series 1 year forward, this way the value of 1940 was now in the row of 1941. Repeating this process 5 times, we were able to create a dataframe containing the dependent variable and 5 lags of this variable. An example of this dataframe can be seen in Appendix E (`data_confl`). The same thing was done fore the frequency data. The models needed to have observations in each column in order to be able to run. This lead to the top and bottom rows with NA values being dropped during the process resulting in the loss of observations. The effect was minimized by leaving only the highest lag that was actually in the model, instead of consistently dropping 10 rows (5 on each side).

3 Methodology

Several models were used in this case study to fit the data, to estimate unknown coefficients and to make out-of-sample predictions, namely Autoregressive Integrated Moving Average (ARIMA), Logit, Probit and Poisson regression. The country selection method was based on the Augmented-Dickey-Fuller (ADF) test and variance in the dependent variable, for the reason that it was necessary to know whether the time series was stationary or not in the ARIMA model. In Logit, Probit and Poisson regeression the stationarity was assumed for all variables in the "skeleton" model. By differencing the non-stationary variables, the results would be difficult to interpret. Each of the mentioned models and methods will be discussed in the following sections.

3.1 General

Throughout this case study the significance level alpha (α) is set to 0.05.
To specify the models, the general-to-specific approach is used. Every time a model is run, the variable with the highest p-value gets removed if it is greater than α, and then the model is run again. This process is repeated until there are only variables left with the p-values $< \alpha$.
To judge the quality of the model the following measures of goodness-of-fit are used: r-squared (r^2), pseudo r-squared (pseudo-r^2), accuracy, Akaike Information Criterion (AIC), Bayesian Information Criterion (BIC) and accuracy.
r^2 tells how much of the variance in the dependent variables from the mean is explained by

the model. It ranges between zero and one. High values of r^2 indicate a good fit, low values indicate a bad fit.
For some models it is not possible to calculate the r^2 value, so instead they provide an alternative, McFadden's pseudo-r^2. It ranges from zero to one and can be interpreted as "excellent" if its value is between 0.2 and 0.4 (D. McFadden, 1977).[5]
AIC takes two times the parameters of the model and subtracts two times the log-likelihood (measure of model fit) from it "$2 * K - 2 * log(likelihood)$". The lower the value of AIC, the better the model fits the data.
In the BIC case, instead of multiplying the model parameters two times, they get multiplied by taking the logarithm of the number of observations "$K * log(n) - 2 * log(likelihood)$". Also here, the lowest values of BIC are considered the best (G. Jogesh Babu, 1992).[1]
To compare the models with each other there is another method that can be used, the accuracy method. This method does an in-sample estimation and checks whether or not the estimation is correct according to the data. When the probability of conflict is below 0.5, the estimation will indicate no conflict. This is checked with the real data, to see how many estimates were predicted correctly. The value of accuracy thus indicates the percentage of correctly estimated presence of conflict.

3.2 Country selection procedure

The ADF test was used to check for stationarity (it was automated to check for all the 199 countries). Due to space and time constraints, it would be inefficient to elaborate on all countries in this case study. The focus is therefore on four countries that appear to have stationary data: the United Kingdom ("UKG"), Turkey ("TUR"), the United States of America ("USA") and the Netherlands ("NTH"); and two countries that have a non-stationary time series: Thailand ("THI") and India ("IND"). All these countries have at least 45 years of conflict in the data. To check what happens when there is little conflict, the stationary country Sweden ("SWD") and the non-stationary Chad ("CHA") were also examined. Both have 24 years of conflict. When a time series has almost no or very low variance, the models do not work. What these models do is they explain the variation of the dependent variable using the independent variables. When there is no variation in the dependent variable there is also nothing the model could explain.

3.3 ARIMA

For the first model (ARIMA), the integration order had to be found before implementing the general-to-specific approach. To check whether or not the data was stationary, and whether differencing the non-stationary data made it stationary, the ADF test was implemented. Using this method it could be determined to what order the model should be integrated. The plots of the autocorrelation function (ACF) and the partial autocorrelation function (PACF) were examined. These two plots helped to give an indication of the p and q orders, with which the analysis could be started (E. Heckman, 2016)[2].

An automated version of ARIMA, from the `pmdarima` package was explored for the frequency data. This function first performs the specified test for stationarity (i.e. ADF, KPSS, PP) to decide the integration order and then runs all p and q orders for the ARIMA model and chooses the specification with the lowest AIC value. This means that this is the only model where the general-to-specific approach was not used. However, this different approach is not a very sophisticated procedure to specify a model. Another downside is that it cannot remove specific lags within the order (i.e. when the AR has six lags, the third lag cannot be removed). These two downsides can lead to the removal and inclusion of the wrong variables. Due to the frequency data being discrete and not binary, the ARIMA could work better than in the case of the binary data.

3.4 LOGIT

To conduct an appropriate regression analysis on a binary dependent variable, Logistic regression is often used. The main idea behind it is to predict the relationship between the dependent variable with outcomes one or zero and the explanatory variables that do not need to be binary. A Linear regression or Linear Probability model may lead to results outside the range of zero and one. The Logit model, which is the inverse of the Logistic function, is used for binary data to make in-sample probability predictions. The linear relationship between the dependent and independent variables is not required, however, it is necessary that the independent variables are linearly related to the log odds (D. Schreiber-Gregory, H.M. Jackson, 2018)[8]. Also, it does not make the assumption that the residuals have to be normally distributed.
The Logit model has some disadvantages as well that may often occur when applied to the data set. The data set needs to be large enough and the data observations have to be independent. Another disadvantage is there should be very little or non-existent multicollinearity among the independent variables. Although, this feature has almost no effect on prediction and forecast in regression models (D.J. Mundfrom, M. DePoy Smith, L.W. Kay, 2017)[6]. The last disadvantage is that the coefficients are difficult to interpret. This is not of great importance, as the aim is not to infer causality, but to forecast probabilities. Logit uses the cumulative distribution function (CDF) of the logistic distribution. It is estimated with Maximum Likelihood Estimation, as it is not efficient when using Ordinary Least Squares. The model takes the lagged values of the dependent variable and the lagged values of the conflict frequency data to predict probability of having a conflict or not. When putting the data of variables into the model the outcome is log of the ratio of odds (Stock and Watson, 2019)[9]:

$$log * \left(\frac{p}{1-p}\right) = log * (beta_0 + beta_1 * X_1 + beta_2 * X_2 + \cdots + beta_p * X_p) \tag{1}$$

The probability values can then be extracted by (Stock and Watson, 2019)[9]:

$$P(Y=1|X) = \frac{e^{beta_0 + beta_1 * X_1 + beta_2 * X_2 + \cdots + beta_p * X_p}}{1 + e^{beta_0 + beta_1 * X_1 + beta_2 * X_2 + \cdots + beta_p * X_p}} \tag{2}$$

The Logit model also used the general to specific approach were the least significant value is removed each time the regression is run.

3.5 PROBIT

The Probit model, like the Logit model, is a nonlinear regression model specifically designed to model binary dependent variables with nonlinear data. It also takes the lagged values of the dependent variable and the lagged values of the conflict frequency data to predict the probability of having a conflict or not. The difference between the Logit and Probit model, is that Probit uses the cumulative distribution function (CDF) of the standard normal distribution. This makes it very easy to use, however the assumption of standard normal distribution is usually not fulfilled. This could be a disadvantage using the Probit model. The coefficients of the Probit model are estimated by Maximum Likelihood Estimation as well, which in large samples is consistent, normally distributed and efficient. However, the sample size in this case might not be large enough for some countries, which could be a problem. Just like the Logit model, the coefficients are not directly interpretable, but this is not an issue as was explained before. When putting the data into the model, the following is obtained (Stock and Watson, 2019)[9]:

$$z = beta_0 + beta_1 * X_1 + beta_2 * X_2 + \cdots + beta_p * X_p \tag{3}$$

The probability values can then be extracted by (Stock and Watson, 2019):

$$P(Y = 1|X) = \Phi(beta_0 + beta_1 * X_1 + beta_2 * X_2 + \cdots + beta_p * X_p) \tag{4}$$

Where the set of the independent variables is the quantile z, which is standard normally distributed (Stock and Watson, 2019)[9]:

$$\Phi(z) = P(Z \leq z), \qquad Z \sim \mathcal{N}(0, 1) \tag{5}$$

The general-to-specific approach was implemented for the Probit model as well.

3.6 POISSON

In order to be able to forecast probabilities of conflict 5 years ahead and make out-of-sample predictions, additional values for the frequency of conflict for each year ahead were needed. Poisson regression was used to forecast frequency of conflict for each additional year. The results of the Poisson regression were used as an input for the out-of-sample predictions with the Logit model. The frequency data can be characterised as a count data and the Poisson distribution is suitable to model the logarithm of the mean as a linear function of observed covariates, and estimate a Poisson regression model to forecast the frequency of conflict next year. Furthermore, it is an advantage to be able to use Poisson regression for heteroscedastic count data and obtain a response variable that is a count per unit, while other linear models would not fit well to the data or the response variable would have different characteristics. To make inferences from the Poisson regression, which is a generalized linear model, the following assumptions were made: outcome of the dependent variable is a count per unit that is described by a Poisson distribution, observations must be independent of one another, mean of a Poisson random variable must be equal to its variance and the log of the mean rate log(λ), must be a linear function of X_i (Rodriguez, 2007)[4].

4 Analysis and Results

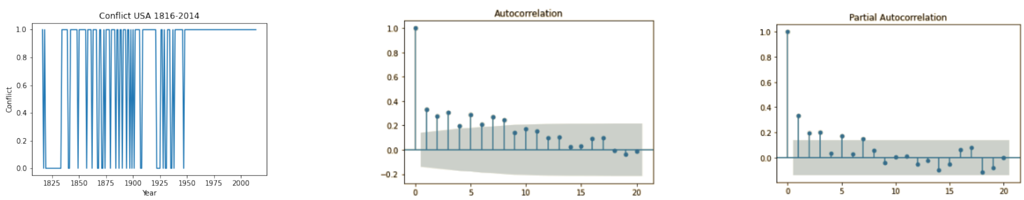
In this section the analysis and results are presented. The steps of the model selection will be reported and explained. The goal of this case study was to create a significant model to forecast the probability of conflict. For the results and the analysis, two countries are selected and thoroughly explained: USA and UKG. The results and forecasts of the six other countries are included in the Appendices A, B, C and D.

4.1 Automation

The data set includes 199 countries. It is time consuming to go through the specification process by hand for each country. In order to make the use of the models for HCSS and further expansion easier and faster, the Poisson, Logit and Probit models are coded in such a way that they are almost fully automated. The user only has to define the country variable as the country of interest and run the code. The variable creation, general-to-specific process, the forecasting and the plotting are then all done automatically.

4.2 ARIMA

The first country that was specified using the ARIMA model was the USA. It is the country with the most observed years of war and the data, as can be observed in figure 1, shows a significant amount of variation before 1950, from which point on it stays in conflict consistently. Overall the data shows strong signs of stochasticity considering the short time span of the data. The PACF and ACF plots can be found in figure 1. According to the ACF there are 8 significant lags and according to the PACF the last significant value is the 7th lag. Using this as an indication, the ARIMA model first started with 8 AR and 8 MA lags. As the USA is stationary the value of integration order is 0.



(a) USA Conflict **(b)** USA ACF **(c)** USA PACF

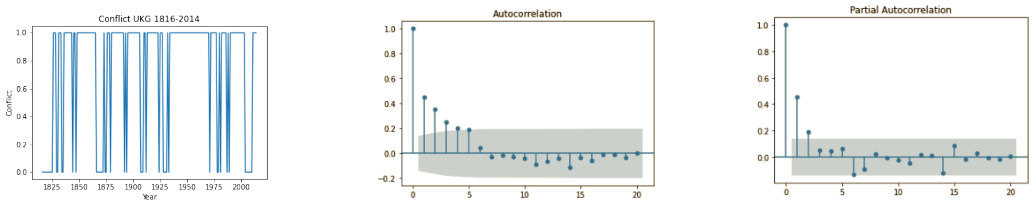
Fig. 1. USA plots

The general-to-specific approach was implemented in the model. After removing the two most insignificant lags the results were invalid. The z values all went to zero and the kurtosis went to almost 200 (a normal distribution has a kurtosis of 3). These results prove that the model is inadequate for this type of data (fig 2).

```
                                        SARIMAX Results
==========================================================================================================
Dep. Variable:                                                USA   No. Observations:                  199
Model:            ARIMA([1, 3, 4, 5, 6, 7, 8], 0, [2, 3, 4, 5, 6, 8])   Log Likelihood                  10.548
Date:                                            Mon, 25 Jan 2021   AIC                             10.903
Time:                                                    16:33:10   BIC                             63.596
Sample:                                                         0   HQIC                            32.230
                                                            - 199
Covariance Type:                                              opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const          1.0000   2.18e-13   4.58e+12      0.000       1.000       1.000
ar.L1         -2.0665    5.6e-19  -3.69e+18      0.000      -2.066      -2.066
ar.L3         -0.0887    1.9e-19  -4.67e+17      0.000      -0.089      -0.089
ar.L4          3.1776   3.52e-20   9.03e+19      0.000       3.178       3.178
ar.L5          3.1290   3.71e-19   8.42e+18      0.000       3.129       3.129
ar.L6         -0.1707   1.04e-18  -1.63e+17      0.000      -0.171      -0.171
ar.L7         -2.0615   2.64e-18    -7.8e+17      0.000      -2.061      -2.061
ar.L8         -0.9613   5.76e-18  -1.67e+17      0.000      -0.961      -0.961
ma.L2          1.6674   2.27e-19   7.33e+18      0.000       1.667       1.667
ma.L3         -0.2551   3.41e-19  -7.49e+17      0.000      -0.255      -0.255
ma.L4         -2.4213   1.38e-19  -1.76e+19      0.000      -2.421      -2.421
ma.L5         -2.4111   3.53e-19  -6.84e+18      0.000      -2.411      -2.411
ma.L6         -0.2269   6.63e-19  -3.42e+17      0.000      -0.227      -0.227
ma.L7          1.6727   1.85e-18   9.05e+17      0.000       1.673       1.673
ma.L8          0.9823   3.85e-18   2.55e+17      0.000       0.982       0.982
sigma2      2.488e-11   1.92e-09      0.013      0.990   -3.75e-09     3.8e-09
===================================================================================
Ljung-Box (L1) (Q):                   0.00   Jarque-Bera (JB):            318582.25
Prob(Q):                              1.00   Prob(JB):                         0.00
Heteroskedasticity (H):               0.00   Skew:                            14.00
Prob(H) (two-sided):                  0.00   Kurtosis:                       197.01
===================================================================================
```

Fig. 2. ARIMA for USA

UKG has many observed conflicts too and the results of its ARIMA model are presented. Although, it has more consistent fluctuation, as shown in figure 3. The ACF and PACF plots can be found in figure 3 and from these figures it can be observed that for the ACF, 5 lags are more or less significant and for the PACF up until the second lag.



(a) UKG Conlict **(b)** UKG ACF **(c)** UKG PACF

Fig. 3. UKG plots

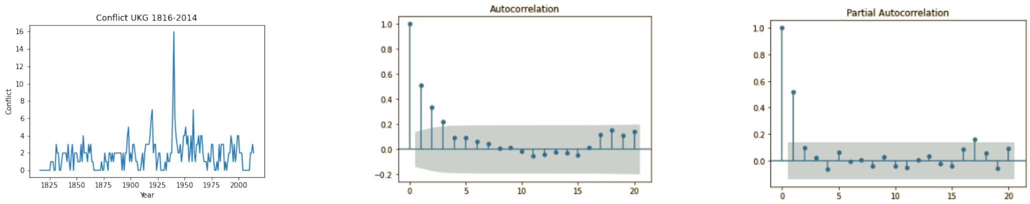
This is why the first model was an ARIMA with 5 autoregressive lags and 5 Moving average lags. The order of integration for the UKG is also 0 since the ADF test has shown it to be stationary. For the UKG, the same issue as for the USA occurred. The results of this model are uninterpretable giving infinite z values, which can be seen in figure 4.

```
                                      SARIMAX Results
=====================================================================================================
Dep. Variable:                                          UKG   No. Observations:                  199
Model:           ARIMA([1, 2, 5, 6, 7, 8], 0, [2, 3, 5, 7, 8])   Log Likelihood                   0.000
Date:                                      Mon, 25 Jan 2021   AIC                             26.000
Time:                                              16:09:58   BIC                             68.813
Sample:                                                   0   HQIC                            43.328
                                                      - 199
Covariance Type:                                        opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.8204         -0       -inf      0.000       0.820       0.820
ar.L1          0.4114         -0       -inf      0.000       0.411       0.411
ar.L2         -0.0585         -0        inf      0.000      -0.058      -0.058
ar.L5          0.8082         -0       -inf      0.000       0.808       0.808
ar.L6         -0.1550         -0        inf      0.000      -0.155      -0.155
ar.L7          0.0575         -0       -inf      0.000       0.058       0.058
ar.L8         -0.4040         -0        inf      0.000      -0.404      -0.404
ma.L2         -0.3534         -0        inf      0.000      -0.353      -0.353
ma.L3          0.1257         -0       -inf      0.000       0.126       0.126
ma.L5         -0.0396         -0        inf      0.000      -0.040      -0.040
ma.L7         -0.3323         -0        inf      0.000      -0.332      -0.332
ma.L8          0.1707         -0       -inf      0.000       0.171       0.171
sigma2         1.2674         -0       -inf      0.000       1.267       1.267
===================================================================================
Ljung-Box (L1) (Q):                   nan   Jarque-Bera (JB):                74.62
Prob(Q):                              nan   Prob(JB):                         0.00
Heteroskedasticity (H):               nan   Skew:                             0.00
Prob(H) (two-sided):                  nan   Kurtosis:                         0.00
===================================================================================
```

Fig. 4. ARIMA for UKG

After having used the model on two countries that seem to have the most variation and stochasticity in their stationary data, it was decided that the ARIMA model was not suited for further analysis of the binary data. As the frequency data was the most accessible explanatory variable to add to the analysis, it was considered the best option to create a framework where more variables can easily be added. It could even have more explanatory power than the lagged dependent variable. To be able to forecast multiple years ahead, the frequency data would have to be forecasted as well.

As this data is different in nature from the binary data it was a good first step to see whether the ARIMA model does work for the frequency data. The frequency data of the USA is not stationary. Therefore, the choice was made to perform analysis on the UKG. The ACF and PACF plots (figure 5) showed few significant lags, however the starting point was set as 6 AR and 6 MA lags to make sure nothing was missed.



(a) UKG Conflict Frequency **(b)** UKG ACF **(c)** UKG PACF

Fig. 5. UKG plots Frequency

```
                               SARIMAX Results
==============================================================================
Dep. Variable:                    UKG   No. Observations:                  199
Model:          ARIMA([1, 3, 4, 5], 0, 6)   Log Likelihood                 5.798
Date:                Mon, 25 Jan 2021   AIC                             12.404
Time:                        17:13:54   BIC                             51.924
Sample:                             0   HQIC                            28.399
                                - 199
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.9355   2.86e-15   3.27e+14      0.000       0.936       0.936
ar.L1          2.3090   1.95e-08   1.19e+08      0.000       2.309       2.309
ar.L3         -2.6826   7.23e-09  -3.71e+08      0.000      -2.683      -2.683
ar.L4          2.2993   9.34e-09   2.46e+08      0.000       2.299       2.299
ar.L5         -0.9793   1.36e-08  -7.21e+07      0.000      -0.979      -0.979
ma.L1         -1.2220   1.05e-08  -1.16e+08      0.000      -1.222      -1.222
ma.L2         -0.4883   8.01e-09   -6.1e+07      0.000      -0.488      -0.488
ma.L3          1.7418   3.43e-09   5.08e+08      0.000       1.742       1.742
ma.L4         -1.2230   9.42e-09   -1.3e+08      0.000      -1.223      -1.223
ma.L5         -0.4867   9.81e-09  -4.96e+07      0.000      -0.487      -0.487
ma.L6          0.7467   1.71e-08   4.37e+07      0.000       0.747       0.747
sigma2         0.8357   4.82e-15   1.73e+14      0.000       0.836       0.836
==============================================================================
Ljung-Box (L1) (Q):                   0.00   Jarque-Bera (JB):         318582.24
Prob(Q):                              1.00   Prob(JB):                      0.00
Heteroskedasticity (H):               0.00   Skew:                        -14.00
Prob(H) (two-sided):                  0.00   Kurtosis:                    197.01
==============================================================================
```

Fig. 6. ARIMA frequency for UKG

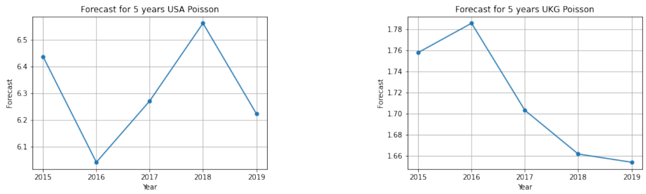
The general-to-specific approach also led to erroneous results for the frequency data as shown in figure 6. The results of the ARIMA models showed that the focus should move to other models. Therefore, the explored auto-ARIMA function was also not used due to that and its flawed specification process.

4.3 POISSON

The Poisson model worked well for the USA data as given in figure 36. The first four out of five lags stayed significant and the model has a pseudo-r^2 of 0.28, meaning it has an excellent fit. The 5-year forecast can be seen in figure 7.

The Poisson model for the UKG, which can be found in figure 37, also did not give errors. The first and third lag were found to be significant. The pseudo-$r2$ is 0.08, unfortunately this means that the fit of this model is not great.

It does, however, enable us to make a forecast that can be used as input for the Logit and Probit models. The 5-year forecast can be seen in figure 7.



(a) Poisson forecast for USA

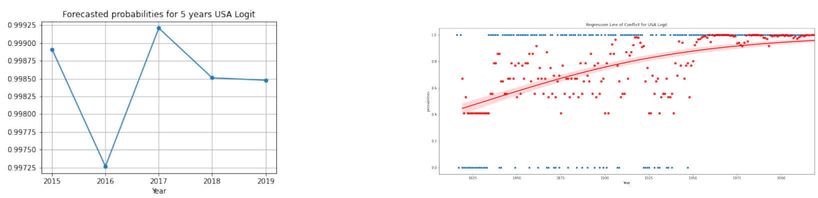
(b) Poisson forecast for UKG

Fig. 7. Poisson forecast

4.4 LOGIT

As it was discussed in the Methodology section, the Logit model is more adequate to use for binary data. The coefficients that are estimated by the Logit model cannot be interpreted as probabilities, instead the latter ones are calculated by the Logit probability formula (2).

The model shows the following goodness-of-fit values for the country USA. The pseudo-r^2 is 0.2440, AIC is 167.488 and BIC is 177.322. The significant coefficients that are observed, are reported in figure 20 in Appendix B. These coefficients are positive and are further used to make a 5-year forecast probabilities of conflict (figure 8). Additionally, the goodness of fit was also measured by looking into accuracy of the model. The amount it correctly predicts whether there is a conflict or not. In the case of Logit model of USA, accuracy is 0.8112. Rounded, this returns an accuracy of 0.81, meaning it predicted 81% times correctly.

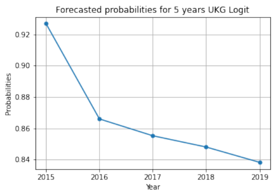


(a) Logit predictions

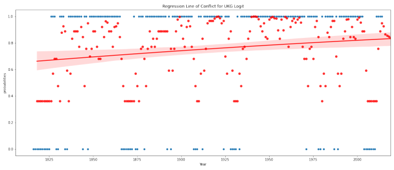
(b) Logit USA

Fig. 8. Logit USA

Figure 21 in Appendix B shows the results of the Logit model and it shows the following measure of goodness-of-fit values for UKG. Pseudo-r^2 of 0.2348, which is sensible since it is between the values 0.2 and 0.4. The AIC and BIC return the values 175.112 and 184.961 respectively. The accuracy of the model is 0.8071, meaning that the model has correctly predicted 81% of the time. The model can now be further used to create a forecast of the probability of UKG having a conflict in the next 5-year period. In figure 9 the forecasts of these probabilities are visually shown. The regression line through the estimated probabilities is plotted in figure 9 as well.



(a) Logit predictions



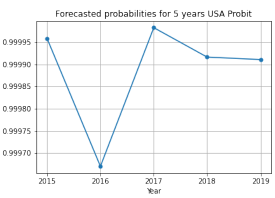
(b) Logit UKG

Fig. 9. Logit UKG

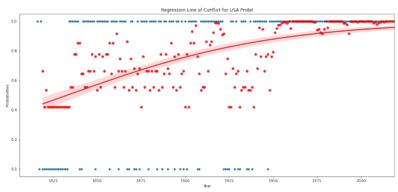
4.5 PROBIT

Next to the Logit model, a Probit model was used and fitted to the data. The Probit model and the Logit model are very much alike and the results of probability are therefore close. The Probit model for the USA returns a pseudo-r^2 of 0.2471, which is a sensible result since it is between the values 0.2 and 0.4, but is slightly higher than the Logit model. AIC measure shows value of 166.812 and BIC gives 176.646. The forecasts of probability of conflict for the Probit model are very close to 1 and are visualised in figure 28 in Appendix C together with the regression line through the estimations (figure 10). The coefficients that are estimated with the Probit model are also not interpretable as probabilities. The quantile function of the standard normal distribution calculates the probability values (4). Nelder-Mead method was used for the countries for which the normal specification methods did not work. This method does not require derivatives (numerical evaluation of the objective function is required only) and works better for the non-stationary countries, which often have a lot of zeros in the data (Nelder and Mead, 1965).[7]

Additionally, the accuracy of the Probit model is calculated the same way as for the Logit, correctly predicting whether there is conflict or not. In case for the USA, accuracy is 0.8112, meaning that it correctly predicted 81% of the time. This is the same as for the Logit model, which could occur since the models are much alike and there are many observations of conflict for USA.



(a) Probit predictions

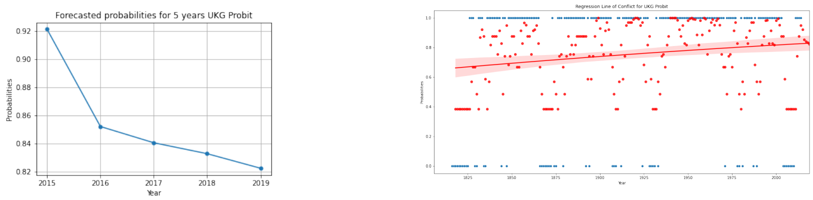


(b) Probit USA

Fig. 10. Probit USA

For the UKG the goodness-of-fit measures are as follows. Pseudo-r^2 of the Probit model is 0.2291, which is slightly lower than the pseudo-r^2 of the Logit model. AIC gives a result of 176.378 and BIC gives 186.228. The accuracy is 0.8071, which is same as of the Logit model.

It correctly predicts 81% of the time. For the UKG only the first and second lagged value of the frequency data are significant. The forecast of probability of conflict is shown in figure 11 below, together with the regression line.



(a) Probit predictions **(b)** Probit UKG

Fig. 11. Probit UKG

5 Discussion

The ARIMA model for binary and frequency data returned many errors. For some countries, when using the general to specific method and with the removal of several specific lags, the model would produce uninterpretable results as shown in figures 17 and 18. This is most likely due to the reason that the ARIMA model is meant for continuous data. The binary and frequency data sets are not very usual for time series data and do not show smooth changes as continuous data would. The ARIMA model is incapable of dealing with this data and thus cannot run the regression properly. The ARIMA model is also not restricted to any boundaries, which is why the probability estimates could return a probability below 0 or above 1, which is not possible.

Although the pseudo-r^2 might not have always been excellent for the Poisson model, the results were sufficient enough to be used as an input in the models. Trying to get a better fit by using more explanatory variables would move the issue. In that case this variable would have to be forecasted as well.

As expected, the AIC and BIC measures presented quite similar results. In the case of the USA, both AIC and BIC differed just by one unit and the lower outcome was given by the Probit model. For the UKG the lowest result was obtained with Logit for both measures.

The Probit model and the Logit model return bounded results between 1 and 0 and are a good fit for nonlinear data, which is why these models are chosen as the main models in this research. The Logit model produces sensible results for the countries of focus, but the Probit model returns an error for Chad. This is likely due to there being too few observations of variation and the resulting non-stationarity. The same error happens for India, which only has variation from 1947 onward as it was founded in that year. This error is fixed by using the Nelder-Mead optimisation in both the Probit and the Logit models. The method also produces better pseudo-r^2. The accuracy of the Logit and Probit models were also relatively good. The lowest accuracy was 0.75 for Turkey, meaning that our model predicted correctly 75% of the time. Most countries scored above 0.80 which is an excellent score for a first skeleton model.

6 Conclusions

The aim of this case study was to create a "skeleton" model which could forecast the probability of conflict for the countries within the data set. During the process of creating this model, a lot of econometric knowledge was tested and applied. The model that was created in the end is a Logit model with lagged values of the dependent variable and lagged values of the frequency of conflict as independent variables. Within the Python code, the estimation of the model for all countries is automated. This means that only one word of the code has to be changed in order to do the prediction for another country. Within the data set are 199 countries. Due to the lack of observations for some countries, the variation can not be explained well enough using the default methods. The time series were created using one matrix. This leads to all countries having a time series starting in 1816.

For the prediction of conflict only two variables and their lagged values are used to do the regression. For this case study the goal was to look at the autoregression and try to elaborate more on the "skeleton" model if the time was available.

The next step would be to include other variables within the model, like GDP, the rate of unemployment or add a dummy variable, whether neighbouring countries are in conflict. Also, the intensity of the conflict in the lagged values could be a very interesting variable to take into account within the model. These new variables could be added very easily into the models once the data is prepared as the entire process was already automated for the frequency variable.

Another possible improvement would be to create a code to start the time series in the year the country was founded instead of the first observed year in the data. This could solve the issues for Chad and India making the Nelder-Mead optimisation unnecessary. The observed errors also pose a potential threat to modelling a conflict between specific countries (a conflict between two or more countries), as these will have even less observations.

The "skeleton" model can also be used for this country specific model as the code is largely automated and only the data and some variable names would have to be changed. To get this data a matrix could be created for each country with all other countries on one axis and all years on the other axis. This way each column would be a time series of conflict between these specific countries which can be used as input.

For further steps more data should be collected on variables that could be important for the prediction of conflict. By including these values, the model could have a larger explanatory power and the forecast could become more accurate.

7 References

[1] G. Jogesh Babu. "Model Selection and Goodness of Fit". In: *Penn State University* (1992). URL: https://astrostatistics.psu.edu/su10/lectures/model_selection_gof.pdf.

[2] Eric Heckman. "Fitting an ARIMA Model". In: *Minitab* (2016). URL: https://blog.minitab.com/blog/starting-out-with-statistical-software/fitting-an-arima-model.

[3] Daniel M. Jones, Stuart A. Bremer, and J. David Singer. "Militarized Interstate Disputes, 1816–1992: Rationale, Coding Rules, and Empirical Patterns". In: *Conflict Management and Peace Science. 1996;15(2):163-213.* (1996). URL: https://doi.org/10.1177/073889429601500203.

[4] Julie Legler and Paul Roback. "Poisson Regression". In: *Broadening Your Statistical Horizons: Generalized Linear Models and Multilevel Models* (2019). URL: https://bookdown.org/roback/bookdown-BeyondMLR/ch-poissonreg.html.

[5] Daniel McFadden. "Quantitative Methods for Analyzing Travel Behaviour of Individuals: Some Recent Developments". In: (1977), p. 35. URL: https://cowles.yale.edu/sites/default/files/files/pub/d04/d0474.pdf.

[6] Daniel J. Mundfrom, Michelle DePoy Smith, and Lisa W. Kay. "The Effect of Multicollinearity on Prediction in Regression Models". In: (2017). URL: file:///Users/julijagaile/Downloads/The_Effect_of_Multicollinearity_on_Prediction_in_R.pdf.

[7] J. A. Nelder and R. Mead. "A simplex method for function minimization". In: *The computer journal, 7(4), 308-313* (1965). URL: https://doi.org/10.1093/comjnl/8.1.27.

[8] Deanna Schreiber-Gregory and Henry M Jackson. "Logistic and Linear Regression Assumptions: Violation Recognition and Control". In: (). URL: https://www.lexjansen.com/wuss/2018/130_Final_Paper_PDF.pdf.

[9] James H. Stock and Mark W. Watson. "Introduction to Econometrics". In: *Boston: Pearson/Addison Wesley* (2019).

Appendices

Appendix A

United States of America			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.99891201 0.99726490 0.99920886 0.99851025 0.99847583	0.99995802 0.99967035 0.99998223 0.99991601 0.99991058	6.43712413 6.04076069 6.27035553 6.56180937 6.22307781
pseudo-r^2	0.2440	0.2471	0.2859
accuracy	0.8112244	0.8112244	

Fig. 12. Forecast for USA

United Kingdom			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.92687445 0.86599096 0.85523631 0.84802646 0.83825407	0.92146624 0.85203769 0.84044207 0.832638 0.82225613	1.758186 1.785984 1.703387 1.661888 1.654080
pseudo-r^2	0.2348	0.2291	0.08933
accuracy	0.8071	0.8071	

Fig. 13. Forecast for UKG

Netherlands			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.50172907 0.25197006 0.19241987 0.18060847 0.17827165	0.50172907 0.25197006 0.19241987 0.18060847 0.17827165	0.444323 0.268630 0.229116 0.221063 0.219456
pseudo-r^2	0.2070	0.2072	0.1276
accuracy	0.8232323	0.8030303	

Fig. 14. Forecast for NTH

Turkey			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.99885769 0.99476963 0.99069337 0.96945492 0.91354330	0.97615248 0.94648231 0.92780745 0.86850245 0.78291853	4.78538563 4.32331486 3.35980600 2.48412079 1.78020874
pseudo-r^2	0.2363	0.2299	0.2370
accuracy	0.7525252	0.7525252	

Fig. 15. Forecast for TUR

Thailand			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.50009082 0.82547418 0.29761276 0.47338697 0.34279861	0.49219379 0.70088849 0.37473691 0.47795230 0.39861532	0.53090482 0.54238849 0.28592774 0.33503111 0.34580676
pseudo-r^2	0.6295	0.6329	0.3549
accuracy	0.896907	0.9072164	

Fig. 16. Forecast THI

India			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.999739740 0.998472487 0.999271390 0.990178096 0.912327484	0.99999582 0.99976376 0.99995112 0.99365465 0.89757488	2.44798015 2.46324379 1.57898328 1.27756900 0.94131993
pseudo-r^2	0.7928	0.7957	0.3800
accuracy	0.9540816	0.9543147	

Fig. 17. Forecast IND

Chad			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.06451900	0.06182478	0.07179663
	0.06451900	0.06182478	0.08075374
	0.06451900	0.06182478	0.08194690
	0.07507895	0.07408691	0.08210717
	0.07650275	0.07573877	0.08212872
pseudo-r^2	0.2513	0.2530	0.2484
accuracy	0.9132653	0.892857	

Fig. 18. Forecast CHA

Sweden			
	LOGIT	PROBIT	POISSON frequency
Forecast	0.34154308	0.35260404	0.24990791
	0.10983956	0.11494627	0.13408660
	0.08996142	0.09215313	0.17269563
	0.09619867	0.09934643	0.13723018
	0.09045532	0.09272399	0.12796788
pseudo-r^2	0.2038	0.2049	0.1548
accuracy	0.9540816	0.898989	

Fig. 19. Forecast SWD

Appendix B

Logit Regression Results

Dep. Variable:	USAConflict	No. Observations:	196
Model:	Logit	Df Residuals:	193
Method:	MLE	Df Model:	2
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.2440
Time:	15:27:08	Log-Likelihood:	-80.744
converged:	True	LL-Null:	-106.80
Covariance Type:	nonrobust	LLR p-value:	4.843e-12

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.3658	0.291	-1.258	0.208	-0.936	0.204
L1_USAFrequency	0.5909	0.197	3.003	0.003	0.205	0.976
L3_USAFrequency	0.4922	0.193	2.544	0.011	0.113	0.871

Fig. 20. Logit for USA

Logit Regression Results

Dep. Variable:	UKGConflict	No. Observations:	197
Model:	Logit	Df Residuals:	194
Method:	MLE	Df Model:	2
Date:	Sat, 23 Jan 2021	Pseudo R-squ.:	0.2348
Time:	15:19:58	Log-Likelihood:	-84.556
converged:	True	LL-Null:	-110.50
Covariance Type:	nonrobust	LLR p-value:	5.380e-12

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.5630	0.296	-1.899	0.058	-1.144	0.018
L1_UKGFrequency	0.8486	0.206	4.125	0.000	0.445	1.252
L2_UKGFrequency	0.4685	0.183	2.554	0.011	0.109	0.828

Fig. 21. Logit for UKG

Logit Regression Results

Dep. Variable:	NTHConflict	No. Observations:	198
Model:	Logit	Df Residuals:	196
Method:	MLE	Df Model:	1
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.2070
Time:	17:48:06	Log-Likelihood:	-84.154
converged:	True	LL-Null:	-106.12
Covariance Type:	nonrobust	LLR p-value:	3.401e-11

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-1.9637	0.240	-8.172	0.000	-2.435	-1.493
L1_NTHFrequency	1.9707	0.350	5.628	0.000	1.284	2.657

Fig. 22. Logit for NTH

```
                           Logit Regression Results
==============================================================================
Dep. Variable:            TURConflict   No. Observations:                  198
Model:                          Logit   Df Residuals:                      196
Method:                           MLE   Df Model:                            1
Date:                Mon, 25 Jan 2021   Pseudo R-squ.:                  0.2363
Time:                        17:48:30   Log-Likelihood:                -102.57
converged:                       True   LL-Null:                       -134.31
Covariance Type:            nonrobust   LLR p-value:                 1.620e-15
===================================================================================
                      coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept          -0.7623      0.219     -3.488      0.000      -1.191      -0.334
L1_TURFrequency     1.2560      0.221      5.691      0.000       0.823       1.689
===================================================================================
```

Fig. 23. Logit for TUR

```
                           Logit Regression Results
==============================================================================
Dep. Variable:            CHAConflict   No. Observations:                  196
Model:                          Logit   Df Residuals:                      194
Method:                           MLE   Df Model:                            1
Date:                Tue, 26 Jan 2021   Pseudo R-squ.:                  0.2513
Time:                        21:24:12   Log-Likelihood:                -54.556
converged:                      False   LL-Null:                       -72.868
Covariance Type:                  HC3   LLR p-value:                 1.432e-09
===================================================================================
                      coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept          -2.6741      0.293     -9.136      0.000      -3.248      -2.100
L3_CHAFrequency     2.2694      0.490      4.630      0.000       1.309       3.230
===================================================================================
```

Fig. 24. Logit for CHA

```
                           Logit Regression Results
==============================================================================
Dep. Variable:            INDConflict   No. Observations:                  197
Model:                          Logit   Df Residuals:                      194
Method:                           MLE   Df Model:                            2
Date:                Tue, 26 Jan 2021   Pseudo R-squ.:                  0.7928
Time:                        21:28:23   Log-Likelihood:                -25.741
converged:                      False   LL-Null:                       -124.21
Covariance Type:                  HC3   LLR p-value:                 1.726e-43
===================================================================================
                      coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept          -3.9150      0.611     -6.413      0.000      -5.112      -2.718
L1_INDFrequency     2.9788      0.849      3.507      0.000       1.314       4.643
L2_INDFrequency     1.5527      0.671      2.313      0.021       0.237       2.868
===================================================================================
```

Fig. 25. Logit for IND

```
                           Logit Regression Results
==============================================================================
Dep. Variable:            THIConflict   No. Observations:                  194
Model:                          Logit   Df Residuals:                      188
Method:                           MLE   Df Model:                            5
Date:                Mon, 25 Jan 2021   Pseudo R-squ.:                  0.6295
Time:                        17:48:58   Log-Likelihood:                -46.558
converged:                       True   LL-Null:                       -125.67
Covariance Type:            nonrobust   LLR p-value:                 2.375e-32
=====================================================================================
                      coef    std err          z      P>|z|      [0.025      0.975]
-------------------------------------------------------------------------------------
Intercept          -2.9679      0.400     -7.427      0.000      -3.751      -2.185
L4_THIConflict      2.2061      1.050      2.101      0.036       0.148       4.264
L1_THIFrequency     1.7209      0.498      3.455      0.001       0.745       2.697
L2_THIFrequency     1.2498      0.495      2.526      0.012       0.280       2.220
L4_THIFrequency    -1.6939      0.592     -2.862      0.004      -2.854      -0.534
L5_THIFrequency     1.1791      0.444      2.655      0.008       0.309       2.050
=====================================================================================
```

Fig. 26. Logit for THI

```
                           Logit Regression Results
==============================================================================
Dep. Variable:            SWDConflict   No. Observations:                  198
Model:                          Logit   Df Residuals:                      196
Method:                           MLE   Df Model:                            1
Date:                Mon, 25 Jan 2021   Pseudo R-squ.:                  0.2038
Time:                        20:20:00   Log-Likelihood:                -58.225
converged:                       True   LL-Null:                       -73.128
Covariance Type:            nonrobust   LLR p-value:                 4.776e-08
===================================================================================
                    coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept        -2.5708      0.288     -8.930      0.000      -3.135      -2.007
L1_SWDFrequency   1.9144      0.401      4.768      0.000       1.127       2.701
===================================================================================
```

Fig. 27. Logit for SWD

Appendix C

Probit Regression Results

Dep. Variable:	USAConflict	No. Observations:	196
Model:	Probit	Df Residuals:	193
Method:	MLE	Df Model:	2
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.2471
Time:	15:42:30	Log-Likelihood:	-80.406
converged:	True	LL-Null:	-106.80
Covariance Type:	nonrobust	LLR p-value:	3.454e-12

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.2013	0.179	-1.125	0.261	-0.552	0.149
L1_USAFrequency	0.3371	0.112	3.009	0.003	0.117	0.557
L3_USAFrequency	0.2875	0.113	2.548	0.011	0.066	0.509

Fig. 28. Probit for USA

Probit Regression Results

Dep. Variable:	UKGConflict	No. Observations:	197
Model:	Probit	Df Residuals:	194
Method:	MLE	Df Model:	2
Date:	Sat, 23 Jan 2021	Pseudo R-squ.:	0.2291
Time:	23:02:54	Log-Likelihood:	-85.189
converged:	True	LL-Null:	-110.50
Covariance Type:	nonrobust	LLR p-value:	1.013e-11

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.2925	0.178	-1.646	0.100	-0.641	0.056
L1_UKGFrequency	0.4693	0.113	4.138	0.000	0.247	0.692
L2_UKGFrequency	0.2563	0.104	2.467	0.014	0.053	0.460

Fig. 29. Probit for UKG

Probit Regression Results

Dep. Variable:	THIConflict	No. Observations:	194
Model:	Probit	Df Residuals:	188
Method:	MLE	Df Model:	5
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.6329
Time:	17:57:15	Log-Likelihood:	-46.133
converged:	True	LL-Null:	-125.67
Covariance Type:	nonrobust	LLR p-value:	1.565e-32

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-1.6968	0.197	-8.634	0.000	-2.082	-1.312
L4_THIConflict	1.2358	0.559	2.209	0.027	0.140	2.332
L1_THIFrequency	0.9425	0.259	3.645	0.000	0.436	1.449
L2_THIFrequency	0.7339	0.283	2.593	0.010	0.179	1.289
L4_THIFrequency	-0.9519	0.303	-3.138	0.002	-1.546	-0.357
L5_THIFrequency	0.6570	0.235	2.792	0.005	0.196	1.118

Fig. 30. Probit for THI

Probit Regression Results

Dep. Variable:	NTHConflict	No. Observations:	198
Model:	Probit	Df Residuals:	196
Method:	MLE	Df Model:	1
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.2072
Time:	17:59:48	Log-Likelihood:	-84.133
converged:	True	LL-Null:	-106.12
Covariance Type:	nonrobust	LLR p-value:	3.327e-11

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-1.1613	0.128	-9.075	0.000	-1.412	-0.910
L1_NTHFrequency	1.1482	0.192	5.982	0.000	0.772	1.524

Fig. 31. Probit for NTH

Probit Regression Results

Dep. Variable:	INDConflict	No. Observations:	197
Model:	Probit	Df Residuals:	194
Method:	MLE	Df Model:	2
Date:	Tue, 26 Jan 2021	Pseudo R-squ.:	0.7957
Time:	20:51:29	Log-Likelihood:	-25.380
converged:	False	LL-Null:	-124.21
Covariance Type:	HC3	LLR p-value:	1.203e-43

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-2.1030	0.250	-8.418	0.000	-2.593	-1.613
L1_INDFrequency	1.6019	0.353	4.541	0.000	0.911	2.293
L2_INDFrequency	0.8387	0.309	2.717	0.007	0.234	1.444

Fig. 32. Probit for IND

Probit Regression Results

Dep. Variable:	SWDConflict	No. Observations:	198
Model:	Probit	Df Residuals:	196
Method:	MLE	Df Model:	1
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.2049
Time:	18:00:12	Log-Likelihood:	-58.143
converged:	True	LL-Null:	-73.128
Covariance Type:	nonrobust	LLR p-value:	4.389e-08

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-1.4746	0.142	-10.410	0.000	-1.752	-1.197
L1_SWDFrequency	1.0963	0.224	4.903	0.000	0.658	1.535

Fig. 33. Probit for SWD

```
                          Probit Regression Results
==============================================================================
Dep. Variable:            TURConflict   No. Observations:                  198
Model:                         Probit   Df Residuals:                      196
Method:                           MLE   Df Model:                            1
Date:                Mon, 25 Jan 2021   Pseudo R-squ.:                  0.2299
Time:                        18:00:37   Log-Likelihood:                -103.43
converged:                       True   LL-Null:                       -134.31
Covariance Type:            nonrobust   LLR p-value:                 3.874e-15
==================================================================================
                     coef    std err          z      P>|z|      [0.025      0.975]
----------------------------------------------------------------------------------
Intercept         -0.4336      0.130     -3.346      0.001      -0.688      -0.180
L1_TURFrequency    0.6909      0.112      6.179      0.000       0.472       0.910
==================================================================================
```

Fig. 34. Probit for TUR

```
                          Probit Regression Results
==============================================================================
Dep. Variable:            CHAConflict   No. Observations:                  196
Model:                         Probit   Df Residuals:                      194
Method:                           MLE   Df Model:                            1
Date:                Tue, 26 Jan 2021   Pseudo R-squ.:                  0.2530
Time:                        21:03:50   Log-Likelihood:                -54.430
converged:                      False   LL-Null:                       -72.868
Covariance Type:                  HC3   LLR p-value:                 1.259e-09
==================================================================================
                     coef    std err          z      P>|z|      [0.025      0.975]
----------------------------------------------------------------------------------
Intercept         -1.5396      0.146    -10.541      0.000      -1.826      -1.253
L3_CHAFrequency    1.3040      0.263      4.965      0.000       0.789       1.819
==================================================================================
```

Fig. 35. Probit for CHA

Appendix D

Poisson Regression Results

Dep. Variable:	USAFrequency	No. Observations:	195
Model:	Poisson	Df Residuals:	190
Method:	MLE	Df Model:	4
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.2859
Time:	17:21:35	Log-Likelihood:	-327.25
converged:	True	LL-Null:	-458.28
Covariance Type:	HC3	LLR p-value:	1.646e-55

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.0445	0.087	-0.512	0.609	-0.215	0.126
L1_USAFrequency	0.1121	0.025	4.492	0.000	0.063	0.161
L2_USAFrequency	0.0698	0.027	2.598	0.009	0.017	0.123
L3_USAFrequency	0.0636	0.029	2.199	0.028	0.007	0.120
L4_USAFrequency	0.0489	0.024	2.062	0.039	0.002	0.095

Fig. 36. Poisson for USA

Poisson Regression Results

Dep. Variable:	UKGFrequency	No. Observations:	196
Model:	Poisson	Df Residuals:	193
Method:	MLE	Df Model:	2
Date:	Mon, 25 Jan 2021	Pseudo R-squ.:	0.08933
Time:	17:18:21	Log-Likelihood:	-325.71
converged:	True	LL-Null:	-357.66
Covariance Type:	HC3	LLR p-value:	1.330e-14

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	0.1655	0.089	1.852	0.064	-0.010	0.341
L1_UKGFrequency	0.1479	0.036	4.109	0.000	0.077	0.219
L3_UKGFrequency	0.0515	0.022	2.333	0.020	0.008	0.095

Fig. 37. Poisson for UKG

Poisson Regression Results

Dep. Variable:	NTHFrequency	No. Observations:	198
Model:	Poisson	Df Residuals:	196
Method:	MLE	Df Model:	1
Date:	Tue, 26 Jan 2021	Pseudo R-squ.:	0.1276
Time:	22:05:45	Log-Likelihood:	-119.61
converged:	True	LL-Null:	-137.11
Covariance Type:	HC3	LLR p-value:	3.318e-09

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-1.7168	0.181	-9.470	0.000	-2.072	-1.361
L1_NTHFrequency	0.9056	0.115	7.874	0.000	0.680	1.131

Fig. 38. Poisson for NTH

Poisson Regression Results

Dep. Variable:	SWDFrequency	No. Observations:	196
Model:	Poisson	Df Residuals:	193
Method:	MLE	Df Model:	2
Date:	Tue, 26 Jan 2021	Pseudo R-squ.:	0.1548
Time:	22:06:08	Log-Likelihood:	-85.862
converged:	True	LL-Null:	-101.59
Covariance Type:	HC3	LLR p-value:	1.478e-07

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-2.2167	0.247	-8.972	0.000	-2.701	-1.732
L1_SWDFrequency	0.8300	0.151	5.493	0.000	0.534	1.126
L3_SWDFrequency	0.3492	0.114	3.056	0.002	0.125	0.573

Fig. 39. Poisson for SWD

Poisson Regression Results

Dep. Variable:	THIFrequency	No. Observations:	194
Model:	Poisson	Df Residuals:	191
Method:	MLE	Df Model:	2
Date:	Tue, 26 Jan 2021	Pseudo R-squ.:	0.3549
Time:	22:08:40	Log-Likelihood:	-152.18
converged:	True	LL-Null:	-235.89
Covariance Type:	HC3	LLR p-value:	4.425e-37

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-1.6017	0.155	-10.308	0.000	-1.906	-1.297
L1_THIFrequency	0.6447	0.076	8.469	0.000	0.495	0.794
L5_THIFrequency	0.3238	0.082	3.967	0.000	0.164	0.484

Fig. 40. Poisson for THI

Poisson Regression Results

Dep. Variable:	TURFrequency	No. Observations:	197
Model:	Poisson	Df Residuals:	194
Method:	MLE	Df Model:	2
Date:	Tue, 26 Jan 2021	Pseudo R-squ.:	0.2370
Time:	22:09:04	Log-Likelihood:	-260.63
converged:	True	LL-Null:	-341.58
Covariance Type:	HC3	LLR p-value:	6.977e-36

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.4362	0.101	-4.311	0.000	-0.634	-0.238
L1_TURFrequency	0.2616	0.047	5.510	0.000	0.169	0.355
L2_TURFrequency	0.1081	0.048	2.256	0.024	0.014	0.202

Fig. 41. Poisson for TUR

```
                     Poisson Regression Results
==============================================================================
Dep. Variable:           INDFrequency   No. Observations:                  196
Model:                        Poisson   Df Residuals:                      192
Method:                           MLE   Df Model:                            3
Date:                Tue, 26 Jan 2021   Pseudo R-squ.:                  0.3800
Time:                        20:51:03   Log-Likelihood:                -159.56
converged:                       True   LL-Null:                       -257.35
Covariance Type:                  HC3   LLR p-value:                 3.775e-42
===================================================================================
                      coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept          -1.5722      0.157    -10.018      0.000      -1.880      -1.265
L1_INDFrequency     0.3824      0.060      6.368      0.000       0.265       0.500
L2_INDFrequency     0.2198      0.074      2.979      0.003       0.075       0.364
L3_INDFrequency     0.2745      0.062      4.420      0.000       0.153       0.396
===================================================================================
```

Fig. 42. Poisson for IND

```
                     Poisson Regression Results
==============================================================================
Dep. Variable:           CHAFrequency   No. Observations:                  198
Model:                        Poisson   Df Residuals:                      196
Method:                           MLE   Df Model:                            1
Date:                Tue, 26 Jan 2021   Pseudo R-squ.:                  0.2484
Time:                        21:03:50   Log-Likelihood:                -66.274
converged:                       True   LL-Null:                       -88.174
Covariance Type:                  HC3   LLR p-value:                 3.639e-11
===================================================================================
                      coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept          -2.6339      0.261    -10.109      0.000      -3.145      -2.123
L1_CHAFrequency     1.6375      0.162     10.124      0.000       1.320       1.955
===================================================================================
```

Fig. 43. Poisson for CHA

Appendix E

	NTHConflict	L1_NTHConflict	L2_NTHConflict	L3_NTHConflict	L4_NTHConflict	L5_NTHConflict
0	0.0	NaN	NaN	NaN	NaN	NaN
1	0.0	0.0	NaN	NaN	NaN	NaN
2	0.0	0.0	0.0	NaN	NaN	NaN
3	0.0	0.0	0.0	0.0	NaN	NaN
4	0.0	0.0	0.0	0.0	0.0	NaN
...	...	...	...	...	...	...
199	NaN	1.0	0.0	0.0	1.0	0.0
200	NaN	NaN	1.0	0.0	0.0	1.0
201	NaN	NaN	NaN	1.0	0.0	0.0
202	NaN	NaN	NaN	NaN	1.0	0.0
203	NaN	NaN	NaN	NaN	NaN	1.0

Fig. 44. Dataframe with created lagged variables

Appendix F

```

MIDB <- read.csv("MIDB 5.0.csv")
\\attach(MIDB)
 \\View(MIDB)
 \\MIDB[,c("ccode", "stday", "stmon", "endday", "endmon", "sidea", "
    revstate", "revtype1", "revtype2", "fatality", "fatalpre", "hiact",
    "hostlev", "orig", "version")] <- list(NULL)
 \\View(MIDB)
 \\library(tidyverse)
 \\MIDB2<- MIDB %>%
  \\ mutate(year = map2(styear, endyear, ':')) %>%
   \\select(-styear, -endyear) %>%
   \\unnest
\\# Other option
\\# MIDB2 =  MIDB %>%
\\#  +     rowwise() %>%
\\#  +     mutate(year = list(seq(styear, endyear, 1))) %>%
\\#  +     ungroup() %>%
\\#  +     select(-styear, -endyear) %>%
\\#  +     unnest()
 \\attach(MIDB2)
 \\MIDBTable <- table(stabb,year)
 \\MIDBTable2 <-xtabs(~stabb+year)
 \\write.csv(MIDBTable2, "MIDBTABLE.csv")
 \\MIDBFREQ<-read.csv("MIDBTABLE.csv")
 \\View(MIDBFREQ)
 \\#add missing years (where nothing happens) in excel by hand

 \\MIDBTable3<-as.data.frame(MIDBTable)
\\ MIDBTable3$Freq<-ifelse(MIDBTable3$Freq>0,1,0)
\\ MIDBDUMMYTABLE<-xtabs(MIDBTable3$Freq~MIDBTable3$stabb+MIDBTable3$
    year)
\\ write.csv(MIDBDUMMYTABLE,"DUMMYTABLE.csv")
 \\#again add missing years by hand in excel
```

Appendix G

```
## ARIMA MODELS

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score as r2_score
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.stattools import adfuller
from statsmodels.tsa.ar_model import AutoReg
from statsmodels.tsa.ar_model import ar_select_order
from sklearn.metrics import r2_score
from statsmodels.tsa.arima.model import ARIMA

df_MIDB = pd.read_csv('BINARYTABLE.csv', sep=";")

# df_MIDB

country = 'USA'

## ACF & PACF

plot_acf(df_MIDB[country], lags=20)
plt.show()

plot_pacf(df_MIDB[country], lags=20)
plt.show()

## Stationarity Test ADF

class StationarityTests:
    def __init__(self, significance=.05):
        self.SignificanceLevel = significance
        self.pValue = None
        self.isStationary = None

    def ADF_Stationarity_Test(self, timeseries, printResults = True):
        #Dickey-Fuller test:
        adfTest = adfuller(timeseries, autolag='AIC')

        self.pValue = adfTest[1]

        if (self.pValue<self.SignificanceLevel):
            self.isStationary = True
        else:
            self.isStationary = False

        if printResults:
            dfResults = pd.Series(adfTest[0:4], index=['ADF Test
    Statistic','P-Value','# Lags Used','# Observations Used'])
            #Add Critical Values
            for key,value in adfTest[4].items():
                dfResults['Critical Value (%s)'%key] = value
            print('Augmented Dickey-Fuller Test Results:')
            print(dfResults)

```

```
sTest = StationarityTests(significance=0.05)
sTest.ADF_Stationarity_Test(df_MIDB[country], printResults = True)
print("Is the {} time series stationary? {}".format(country, sTest.
    isStationary))
print()

## AR

lags = [4]
res = AutoReg(df_MIDB[country], lags=lags, old_names=False).fit()
print(res.summary())


forecast = res.predict(start= len(df_MIDB), end=len(df_MIDB) + 5)
model_estimate = res.predict(start= 0, end=len(df_MIDB))

plt.plot(model_estimate)
plt.plot(df_MIDB[country])
plt.plot(forecast)
plt.legend(["Model estimate", "True Data", "Forecast"])
plt.show()
print(forecast)

## ARIMA with lags from plots ACF, PACF

country_arima = df_MIDB[country]

model_arima = ARIMA(country_arima, order = (4,0,0)).fit()
print(model_arima.summary())


forecast_arima = model_arima.predict(start= len(df_MIDB), end=len(
    df_MIDB) + 5)
arima_estimate = model_arima.predict(start= 1, end=len(df_MIDB))


plt.plot(arima_estimate)
plt.plot(country_arima)
plt.plot(forecast_arima)
plt.legend(["Model estimate", "True Data", "Forecast"])
plt.show()
from sklearn.metrics import r2_score
r2 = r2_score(country_arima,arima_estimate)
print('r2: %f' % r2)
forecast_arima

## ARIMA with arbitrary initial lags = 5

country_arima = df_MIDB[country]

model_arima = ARIMA(country_arima, order = ((1,1,0,1,0),0,(1,1,0,1,0)))
    .fit()
print(model_arima.summary())


forecast_arima = model_arima.predict(start= len(df_MIDB), end=len(
    df_MIDB) + 5)
arima_estimate = model_arima.predict(start= 1, end=len(df_MIDB))
```

```


plt.plot(arima_estimate)
plt.plot(country_arima)
plt.plot(forecast_arima)
plt.legend(["Model estimate", "True Data", "Forecast"])
plt.show()
plt.savefig(country + 'arma_pred.png')

from sklearn.metrics import r2_score
r2 = r2_score(country_arima,arima_estimate)
print('r2: %f' % r2)
forecast_arima

## LOGIT

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score as r2_score
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.stattools import adfuller
import patsy as patsy
from patsy import ModelDesc
from patsy import dmatrices
from patsy import ModelDesc, Term, EvalFactor
import statsmodels.api as sm
from statsmodels.discrete.discrete_model import Logit
from statsmodels.discrete.discrete_model import Probit
import operator
import math
import seaborn as sns
import statsmodels

# Data

df_MIDBFR = pd.read_csv('frequencyMID.csv',sep=";")
df_MIDB = pd.read_csv('BINARYTABLE.csv', sep=";")
Year = df_MIDBFR['YEAR']

# Lagged Variables

country = 'USA'
FirstPredictedYear = 2015
row= FirstPredictedYear - 1816
row2= row + 1
row3= row2 + 1
row4= row3 + 1
row5= row4 + 1
inter_confl = df_MIDB[country]
s3 = pd.Series([np.nan,np.nan,np.nan,np.nan,np.nan])
inter_confl=inter_confl.append(s3,ignore_index=True)
inter_confl=inter_confl.rename(country + "Conflict")

inter_freq = df_MIDBFR[country]
s3 = pd.Series([np.nan,np.nan,np.nan,np.nan,np.nan])
inter_freq=inter_freq.append(s3,ignore_index=True)
inter_freq=inter_freq.rename(country + "Frequency")
```

```
inter_freq1 = inter_freq.shift(1)
inter_freq1 = inter_freq1.rename("L1_"+ country +"Frequency")
inter_freq2 = inter_freq1.shift(1)
inter_freq2 = inter_freq2.rename("L2_"+ country +"Frequency")
inter_freq3 = inter_freq2.shift(1)
inter_freq3 = inter_freq3.rename("L3_"+ country +"Frequency")
inter_freq4 = inter_freq3.shift(1)
inter_freq4 = inter_freq4.rename("L4_"+ country +"Frequency")
inter_freq5 = inter_freq4.shift(1)
inter_freq5 = inter_freq5.rename("L5_"+ country +"Frequency")

inter_lagged = inter_confl.shift(1)
inter_lagged = inter_lagged.rename("L1_" + country + "Conflict")
inter_lagged2 = inter_lagged.shift(1)
inter_lagged2 = inter_lagged2.rename("L2_"+ country +"Conflict")
inter_lagged3 = inter_lagged2.shift(1)
inter_lagged3 = inter_lagged3.rename("L3_"+ country +"Conflict")
inter_lagged4 = inter_lagged3.shift(1)
inter_lagged4 = inter_lagged4.rename("L4_"+ country +"Conflict")
inter_lagged5 = inter_lagged4.shift(1)
inter_lagged5 = inter_lagged5.rename("L5_"+ country +"Conflict")
data_confl = pd.concat([inter_confl, inter_lagged, inter_lagged2,
    inter_lagged3, inter_lagged4, inter_lagged5, inter_freq, inter_freq1
    , inter_freq2, inter_freq3, inter_freq4, inter_freq5],axis=1)


# Poisson Regression for Frequency

#Poisson to predict frequency of unobserved year

Z, K = dmatrices(country + 'Frequency'+ '~' + 'L1_'+country+'Frequency
    + L2_' +country+'Frequency + L3_' +country+ 'Frequency + L4_'
                   +country+ 'Frequency + L5_' +country+ 'Frequency',
    NA_action=patsy.NAAction(NA_types=[]), data=data_confl, return_type=
    'dataframe')

def remove_most_insignificant(df, results):
     max_p_value = max(results.pvalues.iteritems(), key=operator.
    itemgetter(1))[0]
     df.drop(columns = max_p_value, inplace = True)
     return df

insignificant_feature = True
while insignificant_feature:
     modelFR = sm.Poisson(Z, K,missing='drop')
     resultsFR = modelFR.fit(cov_type='HC3')
     significant = [p_value < 0.05 for p_value in resultsFR.pvalues[1:]]
     if all(significant):
         insignificant_feature = False
     else:
         if K.shape[1] == 1:
             print('No significant features found')
             resultsFR = None
             insignificant_feature = False
         else:
             K = remove_most_insignificant(K, resultsFR)

print(resultsFR.summary())
```

```

signif_valuesFR = resultsFR.params.to_frame()
signif_valuesFR = signif_valuesFR.reset_index()
signif_valuesFR.columns = ['sign_variable','coef']


zeta_1 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L1_'+
    country + 'Frequency'] ['coef'].values
zeta_2 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L2_'+
    country + 'Frequency'] ['coef'].values
zeta_3 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L3_'+
    country + 'Frequency'] ['coef'].values
zeta_4 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L4_'+
    country + 'Frequency'] ['coef'].values
zeta_5 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L5_'+
    country + 'Frequency'] ['coef'].values

interceptFR = signif_valuesFR[signif_valuesFR['sign_variable'] == '
    Intercept'] ['coef'].values
if zeta_1.size <= 0:
     zeta_1 = 0

if zeta_2.size <= 0:
     zeta_2 = 0

if zeta_3.size <= 0:
     zeta_3 = 0

if zeta_4.size <= 0:
     zeta_4 = 0

if zeta_5.size <= 0:
     zeta_5 = 0

Frequency1year = interceptFR + (zeta_1 * data_confl['L1_' +country+ '
    Frequency'][row]) + (zeta_2 * data_confl['L2_' +country+ 'Frequency'
    ][row]) + (zeta_3 * data_confl['L3_' +country+ 'Frequency'][row]) +
    (zeta_4 * data_confl['L4_' +country+ 'Frequency'][row]) + (zeta_5 *
    data_confl['L5_' +country+ 'Frequency'][row])
Frequency1year = math.exp(Frequency1year)
Frequency2year = interceptFR + (zeta_1 * Frequency1year) + (zeta_2 *
    data_confl['L2_' +country+ 'Frequency'][row2]) + (zeta_3 *
    data_confl['L3_' +country+ 'Frequency'][row2]) + (zeta_4 *
    data_confl['L4_' +country+ 'Frequency'][row2]) + (zeta_5 *
    data_confl['L5_' +country+ 'Frequency'][row2])
Frequency2year = math.exp(Frequency2year)
Frequency3year = interceptFR + (zeta_1 * Frequency2year) + (zeta_2 *
    Frequency1year) + (zeta_3 * data_confl['L3_' +country+ 'Frequency'][
    row3]) + (zeta_4 * data_confl['L4_' +country+ 'Frequency'][row3]) +
    (zeta_5 * data_confl['L5_' +country+ 'Frequency'][row3])
Frequency3year = math.exp(Frequency3year)
Frequency4year = interceptFR + (zeta_1 * Frequency3year) + (zeta_2 *
    Frequency2year) + (zeta_3 * Frequency1year) + (zeta_4 * data_confl['
    L4_' +country+ 'Frequency'][row4]) + (zeta_5 * data_confl['L5_' +
    country+ 'Frequency'][row4])
Frequency4year = math.exp(Frequency4year)
Frequency5year = interceptFR + (zeta_1 * Frequency4year) + (zeta_2 *
    Frequency3year) + (zeta_3 * Frequency2year) + (zeta_4 *
```

```
    Frequency1year) + (zeta_5 * data_confl['L5_' +country+ 'Frequency'][
    row5])
Frequency5year = math.exp(Frequency5year)
print(Frequency1year)
print(Frequency2year)
print(Frequency3year)
print(Frequency4year)
print(Frequency5year)

## Logit

#Now that we have all data we can use LOGIT to predict
Y, X = dmatrices(country + 'Conflict'+ '~' + ' L1_'+ country + '
    Conflict + L2_'+ country + 'Conflict + L3_'+ country +'Conflict +
    L4_'+ country +
                  'Conflict + L5_'+ country +'Conflict + L1_'+country+'
    Frequency + L2_' +country+'Frequency + L3_' +country+ 'Frequency +
    L4_'
                  +country+ 'Frequency + L5_' +country+ 'Frequency',
    NA_action=patsy.NAAction(NA_types=[]), data=data_confl, return_type=
    'dataframe')



def remove_most_insignificant(df, results):
    max_p_value = max(results.pvalues.iteritems(), key=operator.
    itemgetter(1))[0]
    df.drop(columns = max_p_value, inplace = True)
    return df

insignificant_feature = True
while insignificant_feature:
    model = sm.Logit(Y, X,missing='drop')
    results = model.fit()
    significant = [p_value < 0.05 for p_value in results.pvalues[1:]]
    if all(significant):
        insignificant_feature = False
    else:
        if X.shape[1] == 1:
            print('No significant features found')
            results = None
            insignificant_feature = False
        else:
            X = remove_most_insignificant(X, results)

print(results.summary())

signif_values = results.params.to_frame()
signif_values = signif_values.reset_index()
signif_values.columns = ['sign_variable','coef']
beta1 = signif_values[signif_values['sign_variable'] == 'L1_' + country
     + 'Conflict'] ['coef'].values
beta2 = signif_values[signif_values['sign_variable'] == 'L2_' + country
     + 'Conflict'] ['coef'].values
beta3 = signif_values[signif_values['sign_variable'] == 'L3_' + country
     + 'Conflict'] ['coef'].values
beta4 = signif_values[signif_values['sign_variable'] == 'L4_' + country
     + 'Conflict'] ['coef'].values
```

```
beta5 = signif_values[signif_values['sign_variable'] == 'L5_' + country
    + 'Conflict'] ['coef'].values

theta1 = signif_values[signif_values['sign_variable'] == 'L1_'+ country
    + 'Frequency'] ['coef'].values
theta2 = signif_values[signif_values['sign_variable'] == 'L2_'+ country
    + 'Frequency'] ['coef'].values
theta3 = signif_values[signif_values['sign_variable'] == 'L3_'+ country
    + 'Frequency'] ['coef'].values
theta4 = signif_values[signif_values['sign_variable'] == 'L4_'+ country
    + 'Frequency'] ['coef'].values
theta5 = signif_values[signif_values['sign_variable'] == 'L5_'+ country
    + 'Frequency'] ['coef'].values

intercept = signif_values[signif_values['sign_variable'] == 'Intercept'
    ] ['coef'].values

if theta1.size <= 0:
    theta1 = 0

if theta2.size <= 0:
    theta2 = 0

if theta3.size <= 0:
    theta3 = 0

if theta4.size <= 0:
    theta4 = 0

if theta5.size <= 0:
    theta5 = 0

if beta1.size <= 0:
    beta1 = 0

if beta2.size <= 0:
    beta2 = 0

if beta3.size <= 0:
    beta3 = 0

if beta4.size <= 0:
    beta4 = 0

if beta5.size <= 0:
    beta5 = 0

Y1year = intercept + beta1 * data_confl['L1_' +country+ 'Conflict'][row
    ] + beta2 * data_confl['L2_' +country+ 'Conflict'][row] + beta3 *
    data_confl['L3_' +country+ 'Conflict'][row] + beta4 * data_confl['
    L4_' +country+ 'Conflict'][row] + beta5 * data_confl['L5_' +country+
     'Conflict'][row] + theta1 * data_confl['L1_' +country+ 'Frequency'
    ][row] + theta2 * data_confl['L2_' +country+ 'Frequency'][row] +
    theta3 * data_confl['L3_' +country+ 'Frequency'][row] + theta4 *
    data_confl['L4_' +country+ 'Frequency'][row] + theta5 * data_confl['
    L5_' +country+ 'Frequency'][row]
P1year = math.exp(Y1year)/(1+(math.exp(Y1year)))
Y2year = intercept + beta1 * P1year + beta2 * data_confl['L2_' +country
```

```
    + 'Conflict'][row2] + beta3 * data_confl['L3_' +country+ 'Conflict'
    ][row2] + beta4 * data_confl['L4_' +country+ 'Conflict'][row2] +
    beta5 * data_confl['L5_' +country+ 'Conflict'][row2] + theta1 *
    Frequency1year + theta2 * data_confl['L2_' +country+ 'Frequency'][
    row2] + theta3 * data_confl['L3_' +country+ 'Frequency'][row2] +
    theta4 * data_confl['L4_' +country+ 'Frequency'][row2] + theta5 *
    data_confl['L5_' +country+ 'Frequency'][row2]
P2year = math.exp(Y2year)/(1+(math.exp(Y2year)))
Y3year = intercept + beta1 * P2year + beta2 * P1year + beta3 *
    data_confl['L3_' +country+ 'Conflict'][row3] + beta4 * data_confl['
    L4_' +country+ 'Conflict'][row3] + beta5 * data_confl['L5_' +country
    + 'Conflict'][row3] + theta1 * Frequency2year + theta2 *
    Frequency1year + theta3 * data_confl['L3_' +country+ 'Frequency'][
    row3] + theta4 * data_confl['L4_' +country+ 'Frequency'][row3] +
    theta5 * data_confl['L5_' +country+ 'Frequency'][row3]
P3year = math.exp(Y3year)/(1+(math.exp(Y3year)))
Y4year = intercept + beta1 * P3year + beta2 * P2year + beta3 * P1year +
     beta4 * data_confl['L4_' +country+ 'Conflict'][row4] + beta5 *
    data_confl['L5_' +country+ 'Conflict'][row4] + theta1 *
    Frequency3year + theta2 * Frequency2year + theta3 * Frequency1year +
     theta4 * data_confl['L4_' +country+ 'Frequency'][row4] + theta5 *
    data_confl['L5_' +country+ 'Frequency'][row4]
P4year = math.exp(Y4year)/(1+(math.exp(Y4year)))
Y5year = intercept + beta1 * P4year + beta2 * P3year + beta3 * P2year +
     beta4 * P1year + beta5 * data_confl['L5_' +country+ 'Conflict'][
    row5] + theta1 * Frequency4year + theta2 * Frequency3year + theta3 *
     Frequency2year + theta4 * Frequency1year + theta5 * data_confl['L5_
    ' +country+ 'Frequency'][row5]
P5year = math.exp(Y5year)/(1+(math.exp(Y5year)))
#probabilities
print(P1year)
print(P2year)
print(P3year)
print(P4year)
print(P5year)

## AIC & BIC

print(results.aic)
print(results.bic)


pred=results.predict()
preds=pd.DataFrame(pred)
forecast1=[P1year,P2year,P3year,P4year,P5year]
preds = preds.append(forecast1,ignore_index=True)
startValue=204-len(preds)


realData=df_MIDB[country]
indexYear = pd.read_csv('YearIndex.csv')[startValue:]
index2 = df_MIDB['YEAR'][:]
wide_df2 = pd.DataFrame(realData)
plt.figure(figsize=(20,8))
sns.scatterplot(y=realData,x=index2)
sns.regplot(x=indexYear,y=preds,logistic=True,scatter=True,color='red')
plt.title('Regression Line of Conflict for ' + country + ' Logit')
plt.xlabel('Year')
```

```
plt.ylabel('probabilities')
plt.savefig(country + '_logit_predict.png')


preds2 = [P1year,P2year,P3year,P4year,P5year]
x = ['2015','2016','2017','2018','2019']
plt.title('Forecasted probabilities for 5 years ' + country + ' Logit')
plt.xlabel('Year')
plt.ylabel('Probabilities')
plt.grid()
plt.plot(x,preds2, 'o-', markeredgewidth=0)
plt.savefig(country + '_5years_predict_logit.png')


## PROBIT

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import r2_score as r2_score
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.stattools import adfuller
import patsy as patsy
from patsy import ModelDesc
from patsy import dmatrices
from patsy import ModelDesc, Term, EvalFactor
import statsmodels.api as sm
from statsmodels.discrete.discrete_model import Logit
from statsmodels.discrete.discrete_model import Probit
import operator
import math
import seaborn as sns
import statsmodels

## Data

df_MIDBFR = pd.read_csv('frequencyMID.csv',sep=";")
df_MIDB = pd.read_csv('BINARYTABLE.csv', sep=";")
Year = df_MIDBFR['YEAR']

## Lagged Variables

country = 'USA'
FirstPredictedYear = 2015
row= FirstPredictedYear - 1816
row2= row + 1
row3= row2 + 1
row4= row3 + 1
row5= row4 + 1
inter_confl = df_MIDB[country]
s3 = pd.Series([np.nan,np.nan,np.nan,np.nan,np.nan])
inter_confl=inter_confl.append(s3,ignore_index=True)
inter_confl=inter_confl.rename(country + "Conflict")

inter_freq = df_MIDBFR[country]
s3 = pd.Series([np.nan,np.nan,np.nan,np.nan,np.nan])
inter_freq=inter_freq.append(s3,ignore_index=True)
inter_freq=inter_freq.rename(country + "Frequency")
```

```
inter_freq1 = inter_freq.shift(1)
inter_freq1 = inter_freq1.rename("L1_"+ country +"Frequency")
inter_freq2 = inter_freq1.shift(1)
inter_freq2 = inter_freq2.rename("L2_"+ country +"Frequency")
inter_freq3 = inter_freq2.shift(1)
inter_freq3 = inter_freq3.rename("L3_"+ country +"Frequency")
inter_freq4 = inter_freq3.shift(1)
inter_freq4 = inter_freq4.rename("L4_"+ country +"Frequency")
inter_freq5 = inter_freq4.shift(1)
inter_freq5 = inter_freq5.rename("L5_"+ country +"Frequency")

inter_lagged = inter_confl.shift(1)
inter_lagged = inter_lagged.rename("L1_" + country + "Conflict")
inter_lagged2 = inter_lagged.shift(1)
inter_lagged2 = inter_lagged2.rename("L2_"+ country +"Conflict")
inter_lagged3 = inter_lagged2.shift(1)
inter_lagged3 = inter_lagged3.rename("L3_"+ country +"Conflict")
inter_lagged4 = inter_lagged3.shift(1)
inter_lagged4 = inter_lagged4.rename("L4_"+ country +"Conflict")
inter_lagged5 = inter_lagged4.shift(1)
inter_lagged5 = inter_lagged5.rename("L5_"+ country +"Conflict")
data_confl = pd.concat([inter_confl, inter_lagged, inter_lagged2,
    inter_lagged3, inter_lagged4, inter_lagged5, inter_freq, inter_freq1
    , inter_freq2, inter_freq3, inter_freq4, inter_freq5],axis=1)


## # Poisson Regression for Frequency

#Poisson to predict frequency of unobserved year

Z, K = dmatrices(country + 'Frequency'+ '~' + 'L1_'+country+'Frequency
    + L2_' +country+'Frequency + L3_' +country+ 'Frequency + L4_'
                 +country+ 'Frequency + L5_' +country+ 'Frequency',
    NA_action=patsy.NAAction(NA_types=[]), data=data_confl, return_type=
    'dataframe')

def remove_most_insignificant(df, results):
    max_p_value = max(results.pvalues.iteritems(), key=operator.
    itemgetter(1))[0]
    df.drop(columns = max_p_value, inplace = True)
    return df

insignificant_feature = True
while insignificant_feature:
    modelFR = sm.Poisson(Z, K,missing='drop')
    resultsFR = modelFR.fit(cov_type='HC3')
    significant = [p_value < 0.05 for p_value in resultsFR.pvalues[1:]]
    if all(significant):
        insignificant_feature = False
    else:
        if K.shape[1] == 1:
            print('No significant features found')
            resultsFR = None
            insignificant_feature = False
        else:
            K = remove_most_insignificant(K, resultsFR)

print(resultsFR.summary())
```

```

signif_valuesFR = resultsFR.params.to_frame()
signif_valuesFR = signif_valuesFR.reset_index()
signif_valuesFR.columns = ['sign_variable','coef']


zeta_1 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L1_'+
    country + 'Frequency'] ['coef'].values
zeta_2 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L2_'+
    country + 'Frequency'] ['coef'].values
zeta_3 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L3_'+
    country + 'Frequency'] ['coef'].values
zeta_4 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L4_'+
    country + 'Frequency'] ['coef'].values
zeta_5 = signif_valuesFR[signif_valuesFR['sign_variable'] == 'L5_'+
    country + 'Frequency'] ['coef'].values

interceptFR = signif_valuesFR[signif_valuesFR['sign_variable'] == '
    Intercept'] ['coef'].values
if zeta_1.size <= 0:
     zeta_1 = 0

if zeta_2.size <= 0:
     zeta_2 = 0

if zeta_3.size <= 0:
     zeta_3 = 0

if zeta_4.size <= 0:
     zeta_4 = 0

if zeta_5.size <= 0:
     zeta_5 = 0

Frequency1year = interceptFR + (zeta_1 * data_confl['L1_' +country+ '
    Frequency'][row]) + (zeta_2 * data_confl['L2_' +country+ 'Frequency'
    ][row]) + (zeta_3 * data_confl['L3_' +country+ 'Frequency'][row]) +
    (zeta_4 * data_confl['L4_' +country+ 'Frequency'][row]) + (zeta_5 *
    data_confl['L5_' +country+ 'Frequency'][row])
Frequency1year = math.exp(Frequency1year)
Frequency2year = interceptFR + (zeta_1 * Frequency1year) + (zeta_2 *
    data_confl['L2_' +country+ 'Frequency'][row2]) + (zeta_3 *
    data_confl['L3_' +country+ 'Frequency'][row2]) + (zeta_4 *
    data_confl['L4_' +country+ 'Frequency'][row2]) + (zeta_5 *
    data_confl['L5_' +country+ 'Frequency'][row2])
Frequency2year = math.exp(Frequency2year)
Frequency3year = interceptFR + (zeta_1 * Frequency2year) + (zeta_2 *
    Frequency1year) + (zeta_3 * data_confl['L3_' +country+ 'Frequency'][
    row3]) + (zeta_4 * data_confl['L4_' +country+ 'Frequency'][row3]) +
    (zeta_5 * data_confl['L5_' +country+ 'Frequency'][row3])
Frequency3year = math.exp(Frequency3year)
Frequency4year = interceptFR + (zeta_1 * Frequency3year) + (zeta_2 *
    Frequency2year) + (zeta_3 * Frequency1year) + (zeta_4 * data_confl['
    L4_' +country+ 'Frequency'][row4]) + (zeta_5 * data_confl['L5_' +
    country+ 'Frequency'][row4])
Frequency4year = math.exp(Frequency4year)
Frequency5year = interceptFR + (zeta_1 * Frequency4year) + (zeta_2 *
    Frequency3year) + (zeta_3 * Frequency2year) + (zeta_4 *
```

```
    Frequency1year) + (zeta_5 * data_confl['L5_' +country+ 'Frequency'][
    row5])
Frequency5year = math.exp(Frequency5year)
print(Frequency1year)
print(Frequency2year)
print(Frequency3year)
print(Frequency4year)
print(Frequency5year)

## Probit

#Now that we have all data we can use PROBIT to predict
Y, X = dmatrices(country + 'Conflict'+ '~' + ' L1_'+ country + '
    Conflict + L2_'+ country + 'Conflict + L3_'+ country +'Conflict +
    L4_'+ country +
                  'Conflict + L5_'+ country +'Conflict + L1_'+country+'
    Frequency + L2_' +country+'Frequency + L3_' +country+ 'Frequency +
    L4_'
                  +country+ 'Frequency + L5_' +country+ 'Frequency',
    NA_action=patsy.NAAction(NA_types=[]), data=data_confl, return_type=
    'dataframe')


def remove_most_insignificant(df, results):
    max_p_value = max(results.pvalues.iteritems(), key=operator.
    itemgetter(1))[0]
    df.drop(columns = max_p_value, inplace = True)
    return df

insignificant_feature = True
while insignificant_feature:
    model = sm.Probit(Y, X,missing='drop')
    results = model.fit()
    significant = [p_value < 0.05 for p_value in results.pvalues[1:]]
    if all(significant):
        insignificant_feature = False
    else:
        if X.shape[1] == 1:
            print('No significant features found')
            results = None
            insignificant_feature = False
        else:
            X = remove_most_insignificant(X, results)

print(results.summary())

signif_values = results.params.to_frame()
signif_values = signif_values.reset_index()
signif_values.columns = ['sign_variable','coef']
beta1 = signif_values[signif_values['sign_variable'] == 'L1_' + country
    + 'Conflict'] ['coef'].values
beta2 = signif_values[signif_values['sign_variable'] == 'L2_' + country
    + 'Conflict'] ['coef'].values
beta3 = signif_values[signif_values['sign_variable'] == 'L3_' + country
    + 'Conflict'] ['coef'].values
beta4 = signif_values[signif_values['sign_variable'] == 'L4_' + country
    + 'Conflict'] ['coef'].values
```

```
beta5 = signif_values[signif_values['sign_variable'] == 'L5_' + country
     + 'Conflict'] ['coef'].values

theta1 = signif_values[signif_values['sign_variable'] == 'L1_'+ country
     + 'Frequency'] ['coef'].values
theta2 = signif_values[signif_values['sign_variable'] == 'L2_'+ country
     + 'Frequency'] ['coef'].values
theta3 = signif_values[signif_values['sign_variable'] == 'L3_'+ country
     + 'Frequency'] ['coef'].values
theta4 = signif_values[signif_values['sign_variable'] == 'L4_'+ country
     + 'Frequency'] ['coef'].values
theta5 = signif_values[signif_values['sign_variable'] == 'L5_'+ country
     + 'Frequency'] ['coef'].values

intercept = signif_values[signif_values['sign_variable'] == 'Intercept'
    ] ['coef'].values

if theta1.size <= 0:
    theta1 = 0

if theta2.size <= 0:
    theta2 = 0

if theta3.size <= 0:
    theta3 = 0

if theta4.size <= 0:
    theta4 = 0

if theta5.size <= 0:
    theta5 = 0

if beta1.size <= 0:
    beta1 = 0

if beta2.size <= 0:
    beta2 = 0

if beta3.size <= 0:
    beta3 = 0

if beta4.size <= 0:
    beta4 = 0

if beta5.size <= 0:
    beta5 = 0

import scipy
import scipy.stats as st
Y1year= intercept + beta1 * data_confl['L1_' +country+ 'Conflict'][row]
     + beta2 * data_confl['L2_' +country+ 'Conflict'][row] + beta3 *
    data_confl['L3_' +country+ 'Conflict'][row] + beta4 * data_confl['
    L4_' +country+ 'Conflict'][row] + beta5 * data_confl['L5_' +country+
     'Conflict'][row] + theta1 * data_confl['L1_' +country+ 'Frequency'
    ][row] + theta2 * data_confl['L2_' +country+ 'Frequency'][row] +
    theta3 * data_confl['L3_' +country+ 'Frequency'][row] + theta4 *
    data_confl['L4_' +country+ 'Frequency'][row] + theta5 * data_confl['
    L5_' +country+ 'Frequency'][row]
```

```python
P1year = st.norm.cdf(Y1year)
Y2year = intercept + beta1 * P1year + beta2 * data_confl['L2_' +country + 'Conflict'][row2] + beta3 * data_confl['L3_' +country+ 'Conflict'][row2] + beta4 * data_confl['L4_' +country+ 'Conflict'][row2] + beta5 * data_confl['L5_' +country+ 'Conflict'][row2] + theta1 * Frequency1year + theta2 * data_confl['L2_' +country+ 'Frequency'][row2] + theta3 * data_confl['L3_' +country+ 'Frequency'][row2] + theta4 * data_confl['L4_' +country+ 'Frequency'][row2] + theta5 * data_confl['L5_' +country+ 'Frequency'][row2]
P2year = st.norm.cdf(Y2year)
Y3year = intercept + beta1 * P2year + beta2 * P1year + beta3 * data_confl['L3_' +country+ 'Conflict'][row3] + beta4 * data_confl['L4_' +country+ 'Conflict'][row3] + beta5 * data_confl['L5_' +country + 'Conflict'][row3] + theta1 * Frequency2year + theta2 * Frequency1year + theta3 * data_confl['L3_' +country+ 'Frequency'][row3] + theta4 * data_confl['L4_' +country+ 'Frequency'][row3] + theta5 * data_confl['L5_' +country+ 'Frequency'][row3]
P3year = st.norm.cdf(Y3year)
Y4year = intercept + beta1 * P3year + beta2 * P2year + beta3 * P1year + beta4 * data_confl['L4_' +country+ 'Conflict'][row4] + beta5 * data_confl['L5_' +country+ 'Conflict'][row4] + theta1 * Frequency3year + theta2 * Frequency2year + theta3 * Frequency1year + theta4 * data_confl['L4_' +country+ 'Frequency'][row4] + theta5 * data_confl['L5_' +country+ 'Frequency'][row4]
P4year = st.norm.cdf(Y4year)
Y5year = intercept + beta1 * P4year + beta2 * P3year + beta3 * P2year + beta4 * P1year + beta5 * data_confl['L5_' +country+ 'Conflict'][row5] + theta1 * Frequency4year + theta2 * Frequency3year + theta3 * Frequency2year + theta4 * Frequency1year + theta5 * data_confl['L5_' +country+ 'Frequency'][row5]
P5year = st.norm.cdf(Y5year)
print(P1year)
print(P2year)
print(P3year)
print(P4year)
print(P5year)

## AIC & BIC

print(results.aic)
print(results.bic)

pred=results.predict()
preds=pd.DataFrame(pred)
forecast1=[P1year,P2year,P3year,P4year,P5year]
preds = preds.append(forecast1,ignore_index=True)
startValue=204-len(preds)

realData=df_MIDB[country]
indexYear = pd.read_csv('YearIndex.csv')[startValue:]
index2 = df_MIDB['YEAR'][:]
wide_df2 = pd.DataFrame(realData)
plt.figure(figsize=(18,8))
sns.scatterplot(y=realData,x=index2)
sns.regplot(x=indexYear,y=preds,logistic=True,scatter=True,color='red')
plt.title('Regression Line of Conflict for ' + country + ' Probit')
plt.xlabel('Year')
plt.ylabel('Probabilities')
```

```
plt.savefig(country + '_probit_predict.png')

preds2 = [P1year,P2year,P3year,P4year,P5year]
x = ['2015','2016','2017','2018','2019']
plt.title('Forecasted probabilities for 5 years ' + country + ' Probit'
    )
plt.xlabel('Year')
plt.ylabel('Probabilities')
plt.grid()
plt.plot(x,preds2, 'o-', markeredgewidth=0)
plt.savefig(country + '_5years_predict_probit.png')


from sklearn import metrics

diff = len(data_confl)-5 -len(pred)

metrics1 = metrics.accuracy_score(realData[diff:],pred.round(),
    normalize=True)
metrics2 = metrics.accuracy_score(realData[diff:],pred.round(),
    normalize=False)

print(metrics1)
print(metrics2)
```