Department of Economics and Management
Institute of Economics (ECON)
Statistical Methods and Econometrics
Prof. Dr. Melanie Schienle

Bachelor's Thesis

Development of an Automated Conflict Prediction System by State Space ARIMA Methods

Written by **Adrian Leon Scholl**

Matr. No. **2219278**
B. Sc. Industrial Engineering and Management

Supervisor: **Prof. Dr. Melanie Schienle**

November 11, 2022

I hereby confirm truthfully that I have authored this bachelor's thesis independently and without the use of source material and aids other than those stated, that I have marked all passages literally or textually adapted from other sources as such and that I have respected the statutes of the Karlsruhe Institute of Technology (KIT) for ensuring good scientific practice.

Karlsruhe, 11/11/2022

.....

Adrian Leon Scholl

Abstract

While the urgency for early detection of crises is increasing, truly reliable conflict prediction systems are still not in place, despite the emergence of better data sources and the use of state-of-the-art machine learning algorithms in recent years. Researchers still face the rarity of conflict onset events, which makes it difficult for machine learning-based systems to detect crucial escalation or de-escalation signals. As a result, prediction models can be outperformed by naïve heuristics, such as the no-change model, which leads to a lack of confidence and thus limited practical usability. In this thesis, I address this *heuristic crisis* with the development of a fully-automated machine learning framework capable of optimizing arbitrary econometric state space ARIMA methods in a completely data-driven manner. With this framework, I compare the predictions of a model portfolio consisting of all 8 possible combinations of a standard ARIMA, a seasonal SARIMA, an ARIMAX model with socio-economic variables, and an ARIMAX model with conflict indicators of neighboring countries as exogenous predictors. In addition, each model is examined on a monthly and quarterly periodicity. By comparing the out-of-sample prediction errors, I find that this approach can beat the no-change heuristic in the country-level one-year ahead prediction of the log change of conflict fatalities in all metrics used, including the TADDA score.

Contents

List of Figures

List of Tables

Lists of Abbreviations and Symbols

Abbreviations

ACLED	Armed Conflict Location and Event Dataset
ARIMA	Autoregressive Integrated Moving Average
ARIMAX	Regression with ARIMA errors
CV	Cross-Validation
FOCB	First Observation Carried Backward
IMF	International Monetary Fund
LGSSM	Linear Gaussian State Space Model
LOCB	Last Observation Carried Forward
MAE	Mean Absolute Error
MSE	Mean Square Error
PCA	Principal Component Analysis
RMSE	Root Mean Square Error
SARIMA	Seasonal ARIMA
TADDA	Targeted Absolute Distance with Direction Augmentation
UN	United Nations
UNHCR	United Nations High Commissioner for Refugees
WB	World Bank

Symbols

$\overline{x}$	Sample mean
σ_x	Sample standard deviation
y	Vector of the endogenous variable
X	Matrix of the exogenous variables
f	Forecast vector of the endogenous variable y
y_Δ	Logarithmic change of endogenous variable
f_Δ	Logarithmic change of forecast of the endogenous variable y

1 Introduction

New crises emerge around the world every year, posing an ever-growing challenge to government and international organizations. Recognizing the high social and economic damage caused by crises every year, the urgency of anticipating crises before they occur has gained more and more importance. UN Secretary-General António Guterres even addressed crisis prevention as "not merely a priority, but the priority" in a speech to the Security Council in 2017 (Guterres 2017). This urgency has led to efforts in the development of crisis early warning systems, intended to initiate a paradigm shift from responding to conflict to anticipatory actions for crisis prevention.

An important component of crisis early warning is the prediction of the emergence and development of violent conflicts. Violent conflict is of particular interest because its consequences extend far beyond the direct effects of armed confrontation. For instance, armed conflict is responsible for an all-time high of 89.3 million displaced persons in 2021 according to UNHCR's annual Global Trends Report 2021, leading to significant migration flows (UNHCR 2022). Due to its severe socioeconomic impacts, violent conflict can also be understood as "development in reverse" (Collier et al. 2003). In addition, violent conflict can have global implications and can threaten food security, even in countries not directly affected as, for example, most recently after the start of the Ukraine war in February 2022.

Instead of a random scattering of resources, in theory, a conflict prediction system allows a much more targeted allocation of resources and actions. In reality, however, despite recent advances in research, the prediction of conflicts is still at an early stage. For a long time, conflict modelling focused on the identification of statistically significant variables, instead of out-of-sample prediction or relied on very simple forecasting models, such as low-resolution logit regression models that aimed at classifying countries into conflict or no-conflict. With the emergence of better data sources and technologies, state-of-the-art machine learning methods and out-of-sample validation have become the standard in modern conflict prediction. (Hegre et al. 2013, 2022, Ward et al. 2010)

Recently, there has been a shift from the previous classification approach to predicting escalation and de-escalation based on the log change in the number of conflict fatalities. This fatality log change was also the target variable of the 2021 ViEWS prediction competition organized by Uppsala University in Sweden. In the ViEWS prediction competition, 9 different models for country-month conflict prediction were submitted for the African continent. In the subsequent evaluation of the competition by Vesco et al. (2022), their prediction results were compared with each other, as well as with a naïve no-change heuristic that always predicts a log change of 0. Surprisingly, none of the 9 submitted country-month models could beat the no-change heuristic, in the TADDA metric developed for measuring conflict predictions. With all the state-of-the-art models from conflict research being outperformed by a very simple heuristic, the credibility and thus practicality of current conflict prediction systems in a policy context has to be questioned. (Vesco et al. 2022)

Also, within the German Federal Government, efforts to develop practical crisis early warning systems have also been ongoing for several years. In the PREVIEW Division of the German Federal Foreign Office, which collaborated on this work, a quarterly conflict prediction system is already in use and is undergoing continuous development in close exchange with the research community. As of July 2022, machine learning algorithms such as Random Forest, XGBoost, but also time series models such as Auto ARIMA and Prophet are in use. The machine learning models developed at PREVIEW come very close to the performance of the no-change heuristic, which is used as a baseline, in the TADDA score, as shown in Figure 1. The time series models Auto ARIMA and Prophet, in contrast, perform very poorly with TADDA scores far above that of the no-change heuristic.

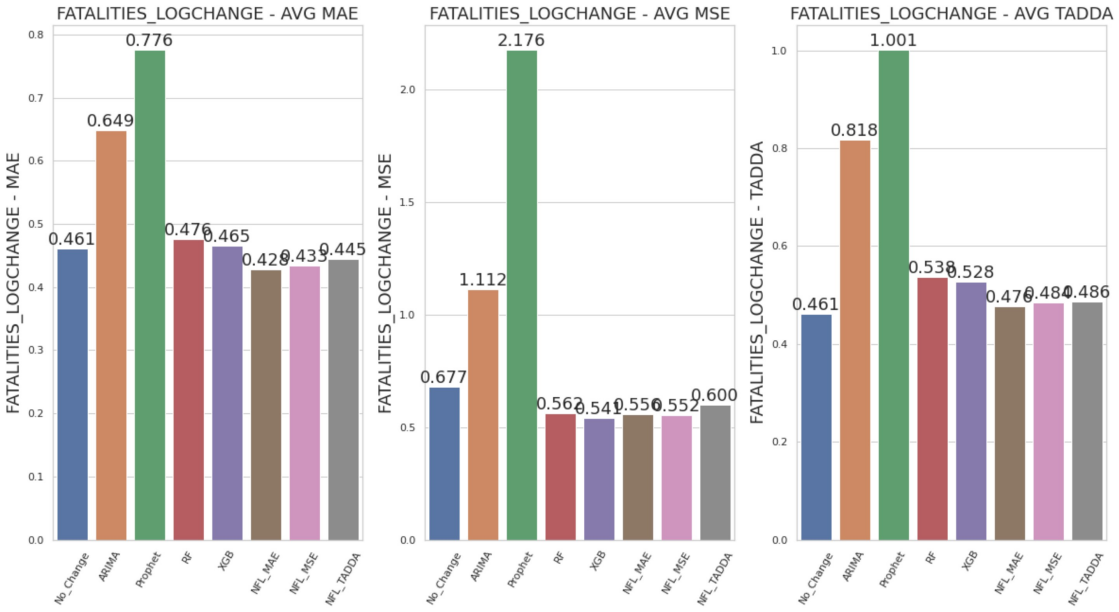

Figure 1: Global out-of-sample MAE, MSE and TADDA error scores of the PREVIEW models for the forecast of the log change in conflict fatalities as of July 2022.

This heuristic crisis provides the motivation for this thesis, in which I have investigated whether and how an approach of econometric state space ARIMA methods can be used as a proposed alternative model to generate successful conflict predictions at country level. To this end, I develop an automated machine learning framework that optimizes the hyperparameters of state space ARIMA methods in a data-driven manner using a pipeline approach and then evaluates them out-of-sample using the metrics TADDA, MAE, and RMSE. With this framework, I test 8 different state space ARIMA variants. These include standard ARIMA, seasonal ARIMA models, ARIMAX models with socio-economic variables and neighbor conflict indicators as exogenous predictors. Since PREVIEW, unlike ViEWS, uses quarterly, rather than monthly, models, I test each of the 8 model variants at both temporal aggregation levels.

It turns out that the automated conflict prediction system developed in this thesis is able to beat the no-change baseline in all metrics, including the TADDA metric. Thus, this approach outperforms both the PREVIEW models and all models submitted to the ViEWS competition in the TADDA score, demonstrating that the no-change heuristic can indeed be beaten in the TADDA score, albeit just marginally.

In particular, these results prove that even less data-hungry time series models like the state space ARIMA methods can at least keep up with state-of-the-art machine learning models with proper hyperparameter tuning. Furthermore, by testing a total of 16 state space ARIMA variants, I identify the model components that are crucial for success. As a first result, it can be shown that the monthly models significantly outperform the quarterly models. While the implementation of seasonality seems to disturb the forecasts, the inclusion of socio-economic exogenous variables or conflict indicators of neighboring countries leads to the best forecasting results. Moreover, in the subsequent country comparison, I find that in addition to geographical patterns, the prediction errors also exhibit a certain dependence on the past country conflict incidence.

This thesis is organized as follows. In the next chapter, I describe the motivation for the models examined, which is derived from conflict theory, and discuss in detail how they are modeled and computed in state space form. Chapter 3 introduces the three datasets used, ACLED, the IMF World Economic Outlook, and the WB World Development Indicators, discussing their properties and their use in the models. Chapter 4 then deals with the development of the automatic machine learning framework for the implementation of the state space ARIMA approach in Python. Chap-

ter 5 then presents the out-of-sample prediction errors of the models, which are then evaluated first at the global level and then at the country level. Finally, in chapter 6, the model results are critically reviewed and compared to the current state-of-the-art, after which the chapter concludes with an outlook for further research in the field of conflict prediction.

2 Method

In this thesis, I develop an automated framework for the global prediction of violent conflict that optimizes 16 different configurations of state space ARIMA methods in a data-driven approach for each country. Each of these ARIMA state space methods is then used to generate individual country-level 1-year ahead conflict forecasts. Therefore, I first specify a conflict indicator to act as the endogenous target variable y of the models which will be subject to the forecasts. For this purpose, I choose the (log) absolute number of conflict fatalities, which is a good indicator for the occurrence of violent conflict for several reasons. First, the mere existence of conflict fatalities is a clear signal that a conflict has escalated into violence. Second, the number of conflict fatalities provides a good quantifiable indication of the severity of the escalation of such conflict. Overall, the choice of this outcome variable ensures that magnitude only of escalated violent conflict is modeled. Furthermore, with sources such as the ACLED dataset, described in Chapter 3.1, a comprehensive data basis on the number of conflict fatalities is available.

For a successful modeling of conflict fatalities for forecasting, I rely on well-known conflict patterns and findings from conflict research. It is, for instance, well known in conflict theory that the probability for the emergence of a conflict is not equally distributed across all countries Collier et al. (2003). Mueller & Rauh (2022) show that, in fact, there is a strong dependence of future conflicts on conflict history. In general, there is very little empirical risk for the outbreak of a violent conflict and thus also for an increased number of conflict-related deaths. However, once a conflict erupts in a country, this country often remains stuck in a repeated cycle of violence, and an increased probability for the onset of new conflicts persists for several years after the start of the conflict. Collier & Sambanis (2002) refer to this phenomenon as the conflict trap.

Mueller & Rauh (2022) find that for countries with recurrent violence, conflict history plays a crucial role in predicting future conflicts. For these countries, the use of an autoregressive model that predicts future conflict fatalities using the number of conflict fatalities from previous periods seems to be a plausible straightforward approach. Hence, I choose to use the popular ARIMA time series model as the foundation for the statistical modelling in this work. The detailed structure of the `ARIMA` model with all its components is described in section 2.1.2 of this chapter.

However, the influence of conflict history on future conflict might exhibit more complex patterns than the simple dependence from the previous periods covered by the ARIMA model. Guardado & Pennings (2020), for example, found a substantial decrease in conflict intensity in the months of the harvest season in Iraq, Pakistan and Afghanistan. To cover such potentially also to other countries applicable seasonal patterns in the conflict history, I extend the basic ARIMA model with annual multiplicative effects to the seasonal ARIMA model `SARIMA`. In addition to the harvest effect, this generically modeled annual seasonality also allows for picking up other possible seasonal effects, such as annually fluctuating climate conditions for example. The mathematical implementation of the addition of multiplicative annual seasonality to the ARIMA model is discussed in Section 2.1.3.

While according to findings by Mueller & Rauh (2022) conflicts in conflict-rich countries can be predicted very well with their conflict history, which I cover with the (S)ARIMA models, Mueller & Rauh (2022) also show that for the prediction of the hard cases, namely the prediction of so far largely conflict-free countries, additional predictors are needed. Unlike Mueller & Rauh (2022), who solve this problem using text indicators from news articles, I pursue an approach with two other types of indicator variables.

In one attempt, I additionaly incorporate socio-economic country indicators X_{SE} into the (S)ARIMA models. In addition to conflict history, Collier et al. (2003) and Collier & Sambanis (2002) also consider the economic conditions of a country as a driver for higher or lower conflict risk. They point out that slower economic growth substantially increases a country's conflict risk.

As with the conflict trap, this is a vicious circle mechanism, because in the same way that a weaker economy leads to an increased risk of conflict, an economic slowdown is a possible consequence of conflict. I attempt to exploit this relationship for predicting conflict fatalities in the models I refer to as `(S)ARIMAX_SE`.

In a second attempt, I extend the (S)ARIMA models by integrating conflict indicators of all surrounding neighboring countries X_N, resulting in the models labeled `(S)ARIMAX_N`. In doing so, I make use of the phenomenon of the spillover effect pointed out by Murdoch & Sandler (2002), Collier & Sambanis (2002). The spillover property of violent conflicts describes that the negative effects of a conflict can spill over to neighboring countries and destabilize them as well. For a country's conflict forecast, this means that the existence of a conflict in a neighboring country also increases its own conflict risk. The inclusion of conflict indicators from neighboring countries can therefore potentially help in forecasting the hard predictable cases, where there are no signs of erupting conflict in the country's conflict history.

Finally, in the `(S)ARIMAX_SE+N` models, I combine both types of exogenous variables just presented into a set X_{SE+N} of exogenous predictors to see how useful these additional variables are in combination for the prediction of conflict fatalities. In all three cases of ARIMA models with additional predictor variables, I assume that all predictor variables are exogenous to conflict fatalities. This, however, is neither for the socio-economic indicators X_{SE+N} nor for the neighbor conflict indicators X_N very likely to be true. As stated above, Collier & Sambanis (2002) point out that conflicts not only have a negative impact on the socio-economic state of a country, but at the same time the negative socio-economic impacts of conflict are a driver for further conflicts. Likewise, the perspective of the spill-over effect can be reversed. The increased risk of conflict in a country caused by conflicts in neighboring countries can also be reflected back to all neighboring countries, where it in turn can contribute to further destabilization. Thus, these effects have a two-way working mechanism. How the indicators, which are assumed to be exogenous, can be elegantly combined with the (S)ARIMA approach to form a regression model with (S)ARIMA errors, is described in detail in Section 2.1.4 of this chapter.

Since it is a priori unknown, which of the model components are actually successful in predicting conflict fatalities and which ones rather disturbing the model, I will test all combinations of the above described model specifications. An overview of these ARIMA variants is provided in Table 1. These models can all be formulated and computed using the state space modeling approach. I discuss the motivation for using this approach, which differs from classical ARIMA modeling, in Section 2.1. I explain the components of the Linear Guassian state space model underlying all state space ARIMA variants in Section 2.1.1. To compute this general state space model, an efficient algorithm, the Kalman filter, can be formulated. With the one-time definition of the Kalman filter for the linear gaussian state space model in Section 2.1.5, the computation algorithm for all derived ARIMA state space models is specified simultaneously.

To be able to compare the predictive power of the models not only to relative to each other, I additionally generate naïve baseline predictions for each country. For this purpose, I choose the no-change model used in the ViEWS Predictions Competition, which extrapolates the log absolute number of fatalities for each country unchanged into the future, see Hegre et al. (2022). The absolute forecast quality of the State Space ARIMA methods can thus be measured by whether and by how much they can beat these baseline predictions. The no-change baseline model is presented in detail in Section 2.2.

Moreover, I examine the influence of the periodicity of the data used in the modelling. In the German Federal Foreign Office, PREVIEW currently uses quarterly models for quantitative crisis early warning. In the ViEWS prediction competition, on the other hand, a monthly periodicity of input data and forecasts was used, see Vesco et al. (2022). In my work, I therefore test the forecast performance of all mentioned models, which are summarized in Table 1, once with monthly data (suffix: `_MONTHLY`) and once with quarterly data (suffix: `_QUARTERLY`). In total, this yields 16 different state space ARIMA model configurations and two no-change baseline models, whose model performances I will analyze and compare in chapter 5.

Complexity Level	Model Name	Description	Exogenous Variables
Baseline	`NO_CHANGE`	Last observation carried forward	None
Tier 0	`ARIMA`	ARIMA	None
Tier 1	`SARIMA`	SARIMA	None
Tier 1	`ARIMAX_SE`	Regression with ARIMA errors	Socio-economic indicators
Tier 1	`ARIMAX_N`	Regression with ARIMA errors	Conflict indicators of neighboring countries
Tier 2	`SARIMAX_SE`	Regression with SARIMA errors	Socio-economic indicators
Tier 2	`SARIMAX_N`	Regression with SARIMA errors	Socio-economic country indicators
Tier 2	`ARIMAX_SE+N`	Regression with ARIMA errors	Socio-economic country indicators and conflict indicators of neighboring countries
Tier 3	`SARIMAX_SE+N`	Regression with SARIMA errors	Socio-economic country indicators and conflict indicators of neighboring countries

Table 1: Overview of all models whose conflict fatality forecasts are evaluated in this work ordered by complexity. The `NO_CHANGE` model provides the baseline for the evaluation of ARIMA state space methods.

2.1 State Space Modelling Approach

In contrast to the classical Box-Jenkins modelling of ARIMA processes, I rely on the state space approach, which allows the formulation of equivalent state space ARIMA equations that can be converted back to the classical ARIMA equations. The State Space approach is kept very general and allows the modeling of any kind of system, which is why it has a wide range of applications in the natural sciences as well as in engineering. For time series forecasting, state space modeling allows a reinterpretation of the underlying problem as the evolution of a system over time: Instead of directly modeling the time series of an observed variable y_t, y_t is linked to the unknown internal states α_t of the system and then the temporal state evolution of the system is modeled. The training of a state space model corresponds to learning the unknown state components based on the observation time series y_t. During forecasting, an estimate for a future state can be generated based on all previous states, which in turn can be used to derive the corresponding future value of the observed variable.

This approach offers the advantage that all ARIMA variants tested in this work can be constructed based on the general linear Gaussian state space model, which is introduced in the next Section 2.1.1. This is a convenient advantage in comparison to the Box-Jenkins approach, where, for example, the integration of exogenous variables is much more difficult to handle. For the computation of the linear Gaussian state space model, there exists a general formulation of the very efficient recursive Kalman filter algorithm, whose properties and mathematical properties are discussed in more detail in section 2.1.5. This property makes it possible to set up the Kalman filter equations only once in a general form for the linear Gaussian state space model and then apply them to all derived state space ARIMA models in the same manner. For the computation of the ARIMA state space models by the Kalman filter, only the state space matrices imposed by the general model have to be specified, which is the subject of the Sections 2.1.2 to 2.1.4 (Commandeur & Koopman 2007).

2.1.1 Linear Gaussian State Space Model

The linear Gaussian state space model (LGSSM) is the general framework for linear models with a Gaussian error term and is therefore capable of handling all ARIMA variants used in this work.

It is described by the following two equations, the observation equation and the state equation.

$$y_t = Z_t \alpha_t + \varepsilon_t, \varepsilon_t \sim \mathcal{N}(0, H_t) \tag{2.1}$$

$$\alpha_{t+1} = T_t \alpha_t + R_t \eta_t, \eta_t \sim \mathcal{N}(0, Q_t), t = 1, ..., n \tag{2.2}$$

$$\alpha_1 \sim \mathcal{N}(a_1, P_1) \tag{2.3}$$

The **observation equation** (2.1) establishes a relation between the observations y_t and the state vector α_t via the matrix Z_t and the error term ε_t. y_t thereby is a $p \times 1$ vector containing the observations of the p endogenous variables at a time t. In my univariate fatality prediction problem, $p = 1$ holds and the y_t correspond to the logarithmic conflict fatalities of a country at time t. The $n_{states} \times 1$ state vector α_t is unobservable and determines the current state of the system at a point of time t. The matrix Z_t is called design matrix. Depending on the model specification, it establishes a relation between the observation vector y_t and the state vector α_t and thus has to be of dimension $p \times n_{states}$. The $p \times 1$ vector ε_t is a Gaussian error term characterized by an expected value of 0 and the diagonal $p \times p$ covariance matrix H_t.

The **state equation** (2.2) describes the transition of the system from one state α_t to the next state α_{t+1}. The new state consists of an autoregressive part, determined by the $n_{states} \times n_{states}$ transition matrix T_t, and a part that depends on the $r \times 1$ error term η_t and is controlled by the $n_{states} \times r$ selection matrix R_t. This property will be used in the following state space definition of the ARIMA models, which also consist of an autoregressive and an error dependent part. The error term η_t follows an r-dimensional normal distribution with expected value 0 and the diagonal $r \times r$ covariance matrix Q_t. The two equations hold for all n time points $t = 1, ..., n$ for which observations y_t are available.

The initial state α_1 (2.3) is assumed to follow a normal distribution whose expected value a_1 and covariance matrix P_1 are assumed to be known.

Theoretically, the design, the transition and the selection matrices Z_t, T_t, R_t as well as the two covariance matrices H_t, Q_t could contain the model parameters to be estimated. In the following sections, however, it becomes clear that in (S)ARIMA modeling T_t will contain all (seasonal) autoregressive parameters and R_t will contain all (seasonal) moving average parameters. The covariance matrix Q_t contains the variance σ^2 of the error term η_t as parameter. The remaining matrices Z_t and H_t will remain unparameterized. It is also worth mentioning that all model components in the Equations (2.1) and (2.2), as implied by the t in the index, are theoretically time-variable in the general model. For the ARIMA formulations in the Sections 2.1.2 to 2.1.4, however, this optional generalization opportunity will be omitted for the matrices Z_t, T_t, R_t, H_t and Q_t.

The linear Gaussian state space model makes the following **assumptions** for the two error terms:

1. Firstly, both errors are each assumed to be serially independent with $\mathbb{E}[\varepsilon_i, \varepsilon_j] = 0$ and $\mathbb{E}[\eta_i, \eta_j] = 0$ for all non-identical time points $i \neq j, i, j = 1, ..., n$.

2. Secondly, the two error terms are additionally assumed to be independent of each other at all time points $i, j = 1, ..., n$, so that $\mathbb{E}[\varepsilon_i, \eta_j], \forall i, j = 1, ..., n$ holds.

2.1.2 ARIMA State Space Model

The ARIMA state space model is the foundation of the state space ARIMA methods discussed in this thesis. The mathematical formulation of the ARIMA state space model presented in this section will serve as a basis for the following Section 2.1.3 on the seasonal ARIMA and Section 2.1.4 on the regression with (S)ARIMA errors.

The acronym ARIMA stands for the **A**uto **R**egressive **I**ntegrated **M**oving **A**verage model. According to the name, the ARIMA model is composed of 3 parts:

1. The AR part (2.4) is constructed similar to a classical regression with the difference that instead of an exogenous predictor variable x, the observations of the endogenous variable

y from the p previous periods, in this case the log fatalities from the previous periods, are used as regressors.

$$y_t = \sum_{i=0}^{p} \phi_i y_{t-i} + \eta_t, \eta_t \sim \mathcal{N}(0, \sigma_\eta^2) \tag{2.4}$$

2. The integration part (2.5) calculates the d-th difference of y_t with its preceding values and thereby aims to eliminate any trends from the time series. This enables also non-stationary conflict time series to be modeled with an ARMA process. The differenced time series is assumed to be stationary and is denoted by y_t^* Hyndman & Athanasopoulos (2014), Durbin & Koopman (2001). For forecasting of conflict fatalities, I restrict myself to using only the simple differencing $d \leq 1$, which is sufficient for most cases, in order to avoid unintended side effects, which, for example, led to a steeply increasing linear trend for the forecast of Ukraine, which could not be justified by the data.

$$y_t^* = y_t - \sum_{k=0}^{d} y_{t-k} \tag{2.5}$$

3. The MA part (2.6) models the dependence of the endogenous variable on its previous observations, unlike the AR part, indirectly by regressing on the error terms of the q previous periods η_{t-j}. The residual values $\hat{\eta_t} = y_t - \sum_{i=1}^{p} \phi_i y_{t-i}$ are realized upon the estimation of the AR parameters ϕ_i and are therefore initially unobservable.

$$y_t = \sum_{j=0}^{q} \theta_j \eta_{t-j} + \eta_t, \eta_t \sim \mathcal{N}(0, \sigma_\eta^2) \tag{2.6}$$

Summarized in Equation (2.7), these three parts yield the ARIMA(p,d,q) model in its classical form:

$$y_t^* = y_t - \sum_{k=0}^{d} y_{t-k} = \sum_{i=0}^{p} \phi_i y_{t-i} + \eta_t + \sum_{j=0}^{q} \theta_j \eta_{t-j}, \eta_t \sim \mathcal{N}(0, \sigma_\eta^2) \tag{2.7}$$

In the following, I introduce the lag operator L, which can be used to represent the ARIMA model in a shortened notation. The lag operator L, when applied to a time dependent variable such as y_t, shifts the variable back by one period in time(2.8). For the k times application of the lag operator (2.9), the variable is shifted back by k periods respectively.

$$Ly_t = y_{t-1} \tag{2.8}$$

$$\underbrace{L...L}_{k \text{ times}} y_t = L^k y_t = y_{t-k} \tag{2.9}$$

By that, the AR and MA part can be expressed by the lag polynomials

$$\Phi_p(L) = 1 - \phi_1 L^1 - \phi_2 L^2 - \cdots - \phi_p L^p \tag{2.10}$$

$$\Theta(L) = 1 - \theta_1 L^1 - \theta_2 L^2 - \cdots - \theta_q L^q \tag{2.11}$$

and the d-th order differencing part can be written with the differencing operator $\Delta = (1 - L)$

$$\Delta^d = (1 - L)^d \tag{2.12}$$

yielding an notation of the ARIMA(p,d,q) model following Fulton (2015) that is equivalent to (2.7):

$$\Phi_p(L)\Delta^d y_t = \Theta_q(L)\eta_t, \eta_t \sim \mathcal{N}(0, \sigma_\eta^2) \tag{2.13}$$

To convert the ARIMA(p, d, q) model into state space form, the model constants m and n_{states} are defined first. The order of the ARIMA(p, d, q) state space model is given by m and depends on the number of integrated AR(p) and MA(q) lags:

$$m \coloneqq max\{p, q+1\} \tag{2.14}$$

Adding the differencing order d gives the constant n_{states}, which specifies length of the state vector α_t:

$$dim(\alpha_t) = n_{states} \coloneqq m + d \tag{2.15}$$

Together, the constants m and n_{states} also determine the size of the state space model matrices Z_t, T_t, R_t. In the linear Gaussian state space model (2.1),(2.2) all matrices are time-varying in their general form. However, for all ARIMA variants in this work, the state space matrices will be constant over all time points t.

$$Z \equiv Z_t, \qquad H \equiv H_t, \qquad T \equiv T_t, \qquad R \equiv R_t, \qquad Q \equiv Q_t$$

The **design matrix Z** in the univariate case of ARIMA(p, d, q) state space model takes the form of a $1 \times n_{states}$ row vector of zeros whose first $d + 1$ elements are filled with ones.

$$Z \coloneqq \big[\underbrace{1 \quad \cdots \quad 1}_{d \text{ times}} \quad 1 \quad \underbrace{0 \quad \cdots \quad 0}_{m-1 \text{ times}}\big] \in \mathbb{R}^{1 \times n_{states}} \tag{2.16}$$

For all (S)ARIMA modeling, the error term ε_t is not required. Consequently, its **covariance matrix H** is set to zero $H \coloneqq 0$, which, together with the expected value of 0, causes the error term ε_t to disappear from the model.

The **transiton matrix T** is constructed from an autoregressive component M_{ar} and a differencing component M_d. The autoregressive component (2.17) handles the AR part of the model and hence also contains the AR coefficients $\phi_i, i = 1, ..., p$. However, for the matrix dimensions to fit together later, there must always be m AR coefficients in the matrix. The $p - m$ artificially added coefficients are set to zero $\phi_i = 0, \forall i > p$, so that they do not interfere with the model. The remaining empty space in T is filled with the $m \times n$ zero matrices $0_{m,n}$. Since T transitions the α_t state to the α_{t+1} state with the aid of $R\eta_t$, T must be of dimension $n_{states} \times n_{states}$.

$$M_{AR} \coloneqq \begin{bmatrix} \phi_1 & 1 & 0 & \cdots & 0 \\ \phi_2 & 0 & 1 & & 0 \\ \vdots & & & \ddots & 0 \\ \phi_{m-1} & & & & 1 \\ \phi_m & 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{m \times m} \tag{2.17}$$

$$M_d \coloneqq \begin{bmatrix} 1 & \cdots & 1 & 1 & 0 & \cdots & 0 \\ & \ddots & \vdots & \vdots & \vdots & & \vdots \\ 0 & & 1 & 1 & \underbrace{0 \quad \cdots \quad 0}_{m-1 \text{ times}} & & \end{bmatrix} \in \mathbb{R}^{d \times n_{states}} \tag{2.18}$$

$$T \coloneqq \begin{bmatrix} M_d & & & & \\ 0_{k,d} & 0_{k,s} & \cdots & 0_{k,s} & M_{ar} \end{bmatrix} \in \mathbb{R}^{n_{states} \times n_{states}} \tag{2.19}$$

The **selection matrix R** is multiplied by the r dimensional error term η_t and thus controls the MA part of the ARIMA state space models. R here shrinks with $r = 1$ to a $d + m$ column vector of zeros whose $(d+1)$-th element is set to one. The last m elements are the moving average coefficients $\theta_1, ..., \theta_m$. As with the AR parameters, in the case of $m > q + 1$ the excess MA parameters are set to zero $\theta_j = 0, \forall j > q + 1$.

$$R \coloneqq \big[\underbrace{0 \quad \cdots \quad 0}_{d \text{ times}} \quad 1 \quad \theta_1 \quad \cdots \quad \theta_{m-1}\big]^\intercal \in \mathbb{R}^{d+m} \tag{2.20}$$

Due to the only one-dimensional normal distribution of η_t, the $r \times r$ **covariance matrix Q** with $r = 1$ shrinks to a scalar $Q \coloneqq \big[\sigma_\eta^2\big]$.

For the ARIMA(p, d, q) state space model, the **state vector** $\alpha_\mathbf{t}$ takes the form given in (2.21), which implicitly follows from the observation and state equation using the matrices defined above. The first $d-1$ elements contain the increasingly differenced values of the prior period y_{t-1}, which corresponds to the correspondingly differenced current value y_t of the state vector prior period α_{t-1}. The $d-th$ entry is the d times differenced value y_t^*. The following m entries represent the ARIMA relationship.

$$\alpha_t \coloneqq \begin{bmatrix} y_{t-1} \\ \Delta y_{t-1} \\ \vdots \\ \Delta^{d-1} y_{t-1} \\ y_t^* \\ \phi_2 y_{t-1}^* + ... + \phi_p y_{t-m+1}^* + \theta_1 \eta_t + ... + \theta_{m-1}\eta_{t-m+2} \\ \phi_3 y_{t-1}^* + ... + \phi_p y_{t-m+2}^* + \theta_2 \eta_t + ... + \theta_{m-1}\eta_{t-m+3} \\ \vdots \\ \phi_m y_{t-1}^* + \theta_{m-1}\eta_t \end{bmatrix} \tag{2.21}$$

In the case of an ARIMA$(2, 2, 2)$ model the model order would be $m = max\{2, 2+1\} = 3$ and α_t would contain $n_{states} = 3 + 2 = 5$ elements. The observation and state equation with all of the above defined components would consequently take the following explicit form:

$$y_t = Z\alpha_t = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \alpha_t$$

$$\begin{aligned} \alpha_{t+1} &= T\alpha_t + R\eta_t \\ &= \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \phi_1 & 1 & 0 \\ 0 & 0 & \phi_2 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \alpha_t + \begin{bmatrix} 0 \\ 0 \\ 1 \\ \theta_1 \\ \theta_2 \end{bmatrix} \eta_t, \eta_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_\eta^2) \end{aligned}$$

2.1.3 SARIMA State Space Model

The SARIMA$(p, d, q) \times (P, D, Q, s)$ state space model complements the ARIMA(p,d,q) by multiplicative seasonal effects. It is difficult to make a general statement about whether and how seasonality really occurs for each country without an elaborate analysis of all 227 conflict time series. In this work, I therefore use a straight forward approach and assume an annual seasonal periodicity. For the monthly models, the seasonal periodicity is hence set to $s = 12$, and for the quarterly models to $s = 4$. As a result, the log number of conflict fatalities in the autoregression is modeled not only by the values of the p previous periods $y_{t-i}, i = 1, ..., p$, but also by the values of the P previous year periods $y_{t-is}, i = 1, ..., P$. The same principle applies to the seasonal moving average term with the Q prior seasonal values of the error $\eta_{t-js}, j = 1, .., Q$. In seasonal differencing, the previous years' values $y_{t-ks}, k = 1, ..., D$ are used accordingly.

To set up the SARIMA equation, I first define the seasonal lag operators $\tilde{P}hi(L)$ and $\tilde{\Theta}(L)$, as well as the seasonal differencing operator Δ_s, analogously to the non-seasonal operators from the previous section.

$$\tilde{\Phi}(L^s) \coloneqq 1 - \tilde{\phi}_1 L^s - \tilde{\phi}_2 L^{2s} - \cdots - \tilde{\phi}_p L^{Ps} \tag{2.22}$$

$$\tilde{\Theta}(L^s) \coloneqq 1 - \tilde{\theta}L^s - \tilde{\theta}L^{2s} - \cdots - \tilde{\theta}L^{Qs} \tag{2.23}$$

$$\Delta_s^D \coloneqq (1 - L^s)^D \tag{2.24}$$

This now allows the SARIMA$(p, d, q) \times (P, D, Q, s)$ model equation to be written in a shorthand notation:

$$\Phi_p(L)\tilde{\Phi}_P(L^s)\Delta^d\Delta_s^D y_t = \Theta_q(L)\tilde{\Theta}_Q(L^s)\eta_t, \eta_t \sim \mathcal{N}(0, \sigma_\eta^2) \tag{2.25}$$

The general form of the seasonal ARIMA in state space form is obtained by integrating seasonal components into the ARIMA state space matrices defined in Section 2.1.2. For this purpose, the ARIMA model order (2.14) is first enhanced with the order of the seasonal parameters m_s which leads to the SARIMA model order m'.

$$m_s \coloneqq max\{P \times s, Q \times s\} \tag{2.26}$$

$$m' \coloneqq m + m_s = max\{p, q + 1\} + max\{P \times s, Q \times s\} \tag{2.27}$$

Besides the model order, the dimension of the state vector α_t, which is the number of states, changes accordingly:

$$dim(\alpha_t) = n_{states} \coloneqq m' + d + D \times s \tag{2.28}$$

Within the **design matrix Z** in the seasonal case the differencing part from the ARIMA state space model (2.16) is extended by Ds entries with ones. Accordingly, all but the first $d + Ds + 1$ ones are now zero entries and the matrix has grown to a dimension of $1 \times (d + Ds + m')$, which again corresponds to $1 \times n_{states}$.

$$Z \coloneqq \big[\underbrace{1 \quad \cdots \quad 1}_{d+Ds \text{ times}} \quad 1 \quad \underbrace{0 \quad \cdots \quad 0}_{m'-1 \text{ times}}\big] \in \mathbb{R}^{1\times(d+Ds+m')} \tag{2.29}$$

The transition matrix **transition matrix T**, which contains an autoregressive and a differencing component in the ARIMA case, now additionally has to account for the seasonal autoregression and the seasonal differencing in the SARIMA case. For this purpose, the seasonal differencing component (2.30) is introduced, whose diagonal structure, is somewhat similar to that of the transposed matrix M_{AR} (see (2.17)).

$$M_D \coloneqq \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathbb{R}^{s\times s} \tag{2.30}$$

The autoregressive component M_{AR} from the ARIMA state space definition (2.17) is extended by the seasonal AR parameters $\tilde{\phi}_1, ..., \tilde{\phi}_{m_s}$ to its seasonal counterpart M_{SAR}. Again, all excess parameters $\tilde{\phi}_i$ with $i > P$ are set to zero.

$$M_{SAR} \coloneqq \begin{bmatrix} \phi_1 & 1 & 0 & \cdots & & & & \cdots & 0 \\ \phi_2 & 0 & 1 & \ddots & & & & & \vdots \\ \vdots & \vdots & \ddots & \ddots & & & & & \\ \phi_m & & & & & & & & \\ \tilde{\phi}_1 & & & & & & & & \\ 0 & & & & & & & & \\ \vdots & & & & & & & & \\ 0 & & & & & & & & \\ \tilde{\phi}_2 & & & & & & & & \\ 0 & & & & & & & & \\ \vdots & & & & & & & & \\ \vdots & & & & & & & & \\ 0 & & & & & & & & \\ \tilde{\phi}_{m_s} & & & & & & & & \\ 0 & & & & & & & \ddots & \ddots & \vdots \\ \vdots & \vdots & & & & & & \ddots & 1 & 0 \\ 0 & 0 & \cdots & & & & & \cdots & 0 & 1 \\ 0 & 0 & \cdots & & & & & & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{m' \times m'} \tag{2.31}$$

Moreover, the matrices $M_{10} \coloneqq \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{s \times m}$, $M_{01} \coloneqq \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathbb{R}^{s \times s}$ and the $(m \times n)$ zero matrix $0_{m,n}$ are required.

Together with the non-seasonal differencing component M_d (2.18) the seasonal autoregressive component (2.31), D diagonal repetitions of the seasonal differencing component M_D (2.30) and the matrices M_{10}, M_{01} and $0_{n,m}$ form the complete SARIMA transition matrix T. The dimensionality of T in the seasonal case is again $n_{states} \times n_{states}$, where n_{states} this time represents the number of states in the seasonal model (2.28).

$$T \coloneqq \begin{bmatrix} & & & M_d & & & & \\ 0_{s,d} & M_D & M_{01} & 0_{s,s} & \cdots & \cdots & 0_{s,s} & M_{10} \\ 0_{s,d} & 0_{s,s} & M_D & M_{01} & 0_{s,s} & \cdots & 0_{s,s} & M_{10} \\ \vdots & \vdots & \ddots & \ddots & \ddots & & \vdots & \vdots \\ 0_{s,d} & 0_{s,s} & \cdots & 0_{s,s} & M_D & M_{01} & 0_{s,s} & M_{10} \\ 0_{s,d} & 0_{s,s} & \cdots & \cdots & 0_{s,s} & M_D & M_{01} & M_{10} \\ 0_{s,d} & 0_{s,s} & \cdots & & \cdots & 0_{s,s} & M_D & M_{10} \\ 0_{k,d} & 0_{k,s} & \cdots & & & \cdots & 0_{k,s} & M_{SAR} \end{bmatrix} \in \mathbb{R}^{n_{states} \times n_{states}} \tag{2.32}$$

The **selection matrix R** is, because of the univariate error term η_t $(r = 1)$, again a column vector with n_{states} entries. The differencing part of the selection matrix increases with seasonal differencing by Ds zero entries. And in addition to the MA parameters of the ARIMA model $\theta_j, j = 1, ..., m$, R now contains additional m_s entries in which the $max\{P, Q\}$ seasonal MA parameters are each followed by $s - 1$ zeros. As with the artificial seasonal AR parameters in T, excess seasonal MA parameters $\tilde{\theta}_j$ with $j > Q$ can occur and will also be set to zero in this case.

$$R \coloneqq \big[\underbrace{0 \quad \cdots \quad 0}_{d+Ds \text{ times}} \quad 1 \quad \theta_1 \quad \cdots \quad \theta_{m-1} \quad \tilde{\theta}_1 \quad \underbrace{0 \quad \cdots \quad 0}_{s-1 \text{ times}} \quad \cdots \quad \tilde{\theta}_{m_s} \quad \underbrace{0 \quad \cdots \quad 0}_{s-1 \text{ times}}\big]^\intercal \in \mathbb{R}^{n_{states} \times 1} \tag{2.33}$$

The SARIMA model also has only a one-dimensional mean-zero error term η_t, whose **covariance matrix Q** therefore again collapses to the scalar $Q \coloneqq [\sigma_\eta^2]$.

In the case of a SARIMA$(1,1,1) \times (1,1,1,4)$ model the model order is $m' = max\{1, 1+1\} + max\{1 \times 4, 1 \times 4\} = 2 + 4 = 6$, the number of states is $n_{states} = 6 + 1 + 1 \times 4 = 10$ and the above defined matrices take the following explicit form:

$$
\begin{aligned}
y_t &= Z\alpha_t \\
&= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \alpha_t
\end{aligned}
$$

$$
\begin{aligned}
\alpha_{t+1} &= T\alpha_t + R\eta_t \\
&= \begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \phi_1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \tilde{\phi}_1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \alpha_t + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ \theta_1 \\ \tilde{\theta}_1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \eta_t, \eta_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_\eta^2)
\end{aligned}
$$

2.1.4 (S)ARIMAX - Regression with (S)ARIMA errors

The (S)ARIMAX model class covers the ARIMA extensions with exogenous variables. The three models `(S)ARIMAX_SE`, `(S)ARIMAX_N`, `(S)ARIMAX_SE+N` differ in the type of exogenous variables used, which however makes no difference for the statistical modeling. These three ARIMA variants are implemented in the form of a regression whose error term ε_t follows an (S)ARIMA process. The state space model for the regression with (S)ARIMA errors can be formulated very elegantly by combining a state space regression model and a (S)ARIMA state space model.

First, the not yet defined state space regression model has to be introduced. The model equation of a multiple regression with intercept β_0 and K exogenous variables $x_{t,k}$ has the general form given in Equation (2.34), which is valid for all observations $t = 1, ..., n$. In contrast to a standard regression, the error term in the (S)ARIMAX model does not follow a Gaussian distribution, but a (S)ARIMA process.

$$
y_t = \beta_0 + \sum_{k=1}^{K} x_{t,k}\beta_k + \varepsilon_t, \varepsilon_t \sim \{\text{(S)ARIMA}(p,d,q) \times (P,D,Q,s)\} \tag{2.34}
$$

The regression model can be transformed to state space form with the below formulation of the state space matrices following (Hyndman n.d., Durbin & Koopman 2001), where $\mathbb{1}$ represents the unit matrix and $0_{K+1,1}$ the corresponding zero matrix.

$$
Z_{reg} \coloneqq \begin{bmatrix} 1 & x_{t,1} & \cdots & x_{t,K} \end{bmatrix}, \qquad \alpha_{t,reg} \coloneqq \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_K \end{bmatrix}, \qquad T_{reg} \coloneqq \mathbb{1}_{K+1}, \qquad R_{reg} \coloneqq 0_{K+1,1}
$$

Altogether, this gives the observation and the state equation of the state space regression model

as indicated below.

$$
\begin{aligned}
y_t &= Z_{reg}\alpha_t + \varepsilon_t \\
&= \begin{bmatrix} 1 & x_{t,1} & \cdots & x_{t,K} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_K \end{bmatrix} + \varepsilon_t, \varepsilon_t \sim \{(\text{S})\text{ARIMA}(p,d,q) \times (P,D,Q,s)\}
\end{aligned}
$$

$$
\begin{aligned}
\alpha_t &= T_{reg}\alpha_{t-1} + R_{reg}\eta_t, \eta_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_\eta^2) \\
&= \alpha_{t-1} = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_K \end{bmatrix}
\end{aligned}
$$

For the modeling of the (S)ARIMA error term, the (S)ARIMA state space matrices formulated in the previous sections 2.1.2 and 2.1.3 can be used, which I will label in the following with the indices *(s)arima* for better distinction. However, it is important to note that now ε_t instead of y_t follows the (S)ARIMA process. The state space matrices for the combined model, the regression with (S)ARIMA errors, are now obtained by simply stacking the regression state space matrices onto the (S)ARIMA state space matrices:

$$
Z \coloneqq \begin{bmatrix} Z_{reg} & Z_{(s)arima} \end{bmatrix}, \quad \alpha_t \coloneqq \begin{bmatrix} \alpha_{t,reg} \\ \alpha_{t,(s)arima} \end{bmatrix}, \quad T \coloneqq \begin{bmatrix} \mathbb{1}_{K+1} & 0 \\ 0 & T_{(s)arima} \end{bmatrix}, \quad R \coloneqq \begin{bmatrix} R_{reg} & R_{(s)arima} \end{bmatrix}
$$

By plugging these matrices into the observation and state equation of the linear Gaussian state space model (see Equation 2.2 and 2.1), the complete regression model with (S)ARIMA errors can be written down in its general form:

$$
\begin{aligned}
y_t &= Z\alpha_t \\
&= \begin{bmatrix} Z_{reg} & Z_{(s)arima} \end{bmatrix} \begin{bmatrix} \alpha_{t,reg} \\ \alpha_{t,(s)arima} \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\alpha_t &= T\alpha_{t-1} + R\eta_t \\
&= \begin{bmatrix} \mathbb{1}_{K+1} & 0 \\ 0 & T_{(s)arima} \end{bmatrix} \begin{bmatrix} \alpha_{t,reg} \\ \alpha_{t,(s)arima} \end{bmatrix} + \begin{bmatrix} R_{reg} & R_{(s)arima} \end{bmatrix} \eta_t, \eta_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_\eta^2)
\end{aligned}
$$

2.1.5 Kalman Filter

The Kalman filter is a recursive algorithm that allows the efficient computation of optimal forecasts for the state space models presented in the previous sections. Once formulated for the state space matrices of the LGSSM, the Kalman filter equations can be easily applied to all state space models derived from the LGSSM.

The goal of the Kalman filter is to obtain the best possible estimate for the distribution of a future state of a state space system α_{t+1} based on a known set of t observations $Y_t = \{y_1, \cdots, y_t\}$ at the current time t and the distribution of the current state α_t.

In the LGSSM the current state α_t is assumed to follow a conditional normal distribution whose expected value $\mathbb{E}[\alpha_t|Y_{t-1}] = a_t$ and covariance matrix $\text{Cov}(\alpha_t|Y_{t-1}) = P_t$ are considered to be given for the recursion step from t to $t+1$. Given a_t and P_t, the parameters for the distribution of the next state can be obtained recursively by inserting the state equation (2.2) of the LGSSM

yielding

$$\begin{aligned} a_{t+1} &= \mathbb{E}[\alpha_{t+1}|Y_t] \\ &= \mathbb{E}[T_t\alpha_t + R_t\eta_t|Y_t] \\ &= T_t\mathbb{E}[\alpha_t|Y_t] \end{aligned} \tag{2.35}$$

$$\begin{aligned} \text{and } P_{t+1} &= \text{Cov}(\alpha_{t+1}|Y_t) \\ &= \text{Cov}(T_t\alpha_t + R_t\eta_t|Y_t) \\ &= T_t\text{Cov}(\alpha_t|Y_t)T_t^\intercal + R_tQ_tR_t^\intercal. \end{aligned} \tag{2.36}$$

This result can now be further simplified and reformulated, as described in Durbin & Koopman (2001), which results in the equations of the Kalman filter (2.37):

$$\begin{aligned} v_t &= y_t - Z_ta_t & F_t &= Z_tP_tZ_t^\intercal + H_t \\ K_t &= T_tP_tZ_t^\intercal F_t^{-1} & L_t &= T_t - K_tZ_t & t &= 1, ..., n \\ a_{t+1} &= T_ta_t + K_tv_t & P_{t+1} &= T_tP_tL_t^\intercal + R_tQ_tR_t^\intercal \end{aligned} \tag{2.37}$$

For models with time-invariant state space matrices, like the ones I use in this work, it can be shown that the covariance estimates P_t eventually converge to a steady state covariance $\overline{P}$ when the Kalman recursion steps are repeatedly applied. From the estimate of the future state vector α_{n+1} resulting from Kalman filtering, the forecast of future observations $\overline{y}_{n+1}$ can be derived using the observation equation (2.1) of the LGSSM.

$$\overline{y}_{n+1} = \mathbb{E}[y_{n+1}|Y_n] = \mathbb{E}[Z_{n+1}\alpha_{n+1} + \varepsilon_{n+1}|Y_n] = Z_na_{n+1} \tag{2.38}$$

With the normal distribution assumptions made for the LGSSM in Section 2.1.1, the Kalman filter with $\overline{y}_{n+1}$ provides the best estimate in terms of the mean square error, introduced in the following Section 2.3. In other words, this means that the mean square prediction error matrix

$$\overline{F}_{n+1} = \mathbb{E}\left[\left(\overline{y}_{n+1} - y_{n+1}\right)\left(\overline{y}_{n+1} - y_{n+1}\right)^\intercal\right] = Z_{n+1}P_{n+1}Z_{n+1}^\intercal + H_{n+1} \tag{2.39}$$

is minimized by the Kalman estimator $\overline{y}_{n+1}$. Even for non-normally distributed observations y_t, it can be shown for the ARIMA state space models of this paper with time invariant matrices that under certain conditions the Kalman algorithm still provides the best linear estimation. This, together with its other performance benefits, makes the Kalman filter a powerful tool for the automated prediction of conflict time series. (Kleeman 1996, Durbin & Koopman 2001)

2.2 No-Change Baseline Model

In order to not only compare the performances of the previously described ARIMA state space methods among each other, but also to allow for an absolute judgement of the predictive power, a baseline forecasting model is needed for reference. In this work, I rely on the naïve no-change model as a baseline, which is also used in the ViEWS prediction competition to evaluate different model approaches, see Hegre et al. (2022).

As its name implies, the no-change model uniformly predicts zero change from the status quo, the last known observation of a time series. In the context of conflict prediction, the no-change model for the prediction of the absolute number of fatalities can be described by carrying forward the last observed value to future periods. This is denoted in Equation (2.2) referring to definition of Alquist et al. (2013).

$$f_{t+1|t} = y_t \tag{2.40}$$

In the case of forecasting the log change in fatalities discussed in this paper, the no-change forecast can be described by Equation (2.2) which corresponds to setting all future log-change values straight to 0.

$$f_{\Delta,t} = 0, \forall t = 1, \cdots, fh \tag{2.41}$$

Thus, the naïve approach of the No-Change model does not use any information for the prediction. In order for the higher complexity of the State Space ARIMA models, which can incorporate the entire conflict history along with exogenous variables, to be considered useful, they must outperform the No-Change model. However, in the ViEWS Prediction Competition, it has been demonstrated by Vesco et al. (2022) already that even with very sophisticated models the No-Change forecasts are, at least in a classification setting, very difficult to beat. At country-month level, none of the 11 contributed models was able to outperform the no-change baseline for the out-of-sample forecasts in the TADDA metric, which is defined in the following section.

2.3 Evaluation Metrics

For the evaluation of the goodness of the out-of-sample forecasts, I use the TADDA metric, the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). All three of the metrics measure the prediction error in the unit of the logarithmic change in conflict fatalities relative to the last known in-sample observation of logarithmic fatalities. A lower value of these metrics indicates a better model performance. For the following definitions of the metrics, let fh denote the forecast horizon, f_Δ denote the $fh \times 1$ vector of the forecasted log change in fatalities and y_Δ denote the $fh \times 1$ vector containing the true log change in fatalities from the test set.

2.3.1 TADDA

The TADDA score was designed by the ViEWS team from Uppsala University especially for the purpose of measuring the error of conflict prediction systems. The acronym TADDA stands for **T**argeted **A**bsolute **D**istance with **D**irection **A**ugmentation. Like the MAE and RMSE, the TADDA score measures the magnitude of the prediction error, but additionally accounts for the direction of the forecasted change in fatalities, which indicates either conflict escalation or de-escalation. (Vesco et al. 2022)

$$TADDA(y_\Delta, f_\Delta, \epsilon) = \frac{\overbrace{\sum_{i=1}^{N} |y_{\Delta,i} - f_{\Delta,i}|}^{\|y_\Delta - f_\Delta\|_1} + t_\epsilon(y_{\Delta,i}, f_{\Delta,i})}{N} \tag{2.42}$$

Hence, the first part of the TADDA metric is an L1 term that measures the absolute deviation of the forecast f_Δ from the true value y_Δ. The additional penalization of an incorrectly predicted directional change is realized by adding a directional penalty term $t_\epsilon(y_{\Delta,i}, f_{\Delta,i})$, with $sgn(x)$ representing the signum function. The absolute value of the change $|f_{\Delta,i}|$ predicted in the wrong direction, thereby, determines the magnitude of the penalty. For this work, I choose to set the threshold ϵ to the close-to-zero value 0.048.

$$t_\epsilon(y_{\Delta,i}, f_{\Delta,i}) = \left\{ \begin{array}{rl} |f_{\Delta,i}|, & \text{if } sgn(y_{\Delta,i}) = sgn(f_{\Delta,i}) \text{ and } |y_{\Delta,i} - f_{\Delta,i}| > \epsilon \\ 0, & \text{if } sgn(y_{\Delta,i}) \neq sgn(f_{\Delta,i}) \text{ or } |y_{\Delta,i} - f_{\Delta,i}| \leq \epsilon \end{array} \right\} \tag{2.43}$$

$$sgn(x) = \left\{ \begin{array}{rl} -1, & \text{if } x < 0 \\ 0, & \text{if } x = 0 \\ 1, & \text{if } 0 < x \end{array} \right\} \tag{2.44}$$

2.3.2 Mean Absolute Error (MAE)

The mean absolute error, contrary to the TADDA score, only measures the magnitude of the predictions error using an L1-term.

$$MAE(y_\Delta, f_\Delta) = \frac{1}{N}\|y_\Delta - f_\Delta\|_1 = \frac{1}{N}\sum_{i=1}^{N}|y_{\Delta,i} - f_{\Delta,i}| \tag{2.45}$$

2.3.3 Root Mean Square Error (RMSE)

The mean squared error is based on the mean square error $MSE = \frac{1}{N}\|y_\Delta - f_\Delta\|_2$. The latter is equivalent to MAE, except that it uses an L2 term instead of an L1. By squaring the individual errors, the RMSE, compared to the MAE, penalizes extreme prediction errors extra strongly and rewards small deviations close to zero. The square root ensures that the RMSE has the same unit as the TADDA score and the MAE, which facilitates the interpretation of the evaluation results.

$$RMSE(y_\Delta, f_\Delta) = \sqrt{\frac{1}{N}\|y_\Delta - f_\Delta\|_2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_{\Delta,i} - f_{\Delta,i})^2} \tag{2.46}$$

3 Data

Three data sets with a global coverage were used as input to the state space ARIMA models formulated in the previous chapter. The most important one is the ACLED data set which serves as the ground truth of conflict history and is described in Section 3.1. Hence, I generate the target variable, the sum of conflict fatalities, and the conflict indicators of neighboring countries used in the `ARIMAX_N` models from the ACLED dataset. Additionally, I also analyze the average conflict fatalities per capita at country level, which will eventually lead to a categorization of the countries according to their historical conflict prevalence used for the later model performance analysis in Chapter 5.

The other two data sets used are a subset of the World Development Indicators by the United Nations World Bank, which was selected for conflict prediction by experts in the German Federal Foreign Office and the World Economic Outlook by the International Monetary Fund. These two data sets contain a set of socio-economic indicators and are incorporated together into the `ARIMAX_SE` models as exogenous predictor variables. Depending on the ARIMA model variant, often more than 100 exogenous variables originating from these three data sets are available to one country model in the later model fitting. Mueller & Rauh (2022) point out that variables, for which a causal conflict relationship is proven by conflict theory, do not always have to be good predictors. Therefore, I do not perform any further causality based selection of exogenous variables, but instead include the entire set of varied exogenous indicators from the selected sources. In the later model building process (see Chapter 4.4), the exogenous indicators will then be automatically incorporated into the corresponding (S)ARIMAX models in a data-driven approach as PCA principal components.

3.1 Armed Conflict Location and Event Data Project (ACLED)

The Armed Conflict Location and Event Dataset (ACLED)[1] provides event-based, real-time information on political or violent conflict events worldwide and thus constitutes the foundation for conflict modeling in this thesis. It is maintained by the US-registered non-profit organization ACLED, that updates the data set with new conflict events on a weekly basis. The ACLED data is published in form of fully disaggregated events and is enriched with a detailed description of the event, the involved actors, an event category and the number of fatalities. In addition, each ACLED event is geolocated with latitude and longitude and is assigned to one of 227 countries or first level administrative units. Disputed border regions or territories that are not internationally recognized are listed individually. (Raleigh et al. 2010)

An important concern using ACLED as ground truth is that the countries were included in the dataset at different start dates. While the entire African continent is included in the dataset since January 1997, the coverage of events in Australia and Oceania, for example, does not start until January 2021. As a result, the predictive models for African countries have much more training data available than other countries. This can introduce bias to the model results, making the country-level models not entirely comparable among each other. Table 7 in the Appendix shows a detailed list of the start dates with the respective number of countries and their corresponding ISO-3 country codes.

As described in Chapter 2, I build each model for each country individually, once on a monthly level and once on a quarterly level. In a first preprocessing step, I generate a set of conflict indicators from the disaggregated ACLED event data. By counting the conflict events and summing the

[1] Retrieved from *https://acleddata.com/data-export-tool/* (2022, March 03)

fatalities per country and per month/quarter and including the event categories *protests (protests and riots)*, *security related incidents (violence against civilians, explosions/remote violence and battles)* and *strategic developments*, the 8 ACLED conflict indicators listed in table 8 are generated. Assuming that ACLED lists all relevant conflict events, for time periods in which no events exist, the sum of the conflict fatalities and the number of events are set to zero. The aggregated ACLED indicators thus do not contain any missing values and are therefore excluded from the imputation in the implementation of the `SARIMAX_LinInt_PCA()` class (see Chapter 4.2).

The sum of conflict fatalities per time period `SUM(FATALITIES)` serves as the endogenous target variable for the state space ARIMA models described in Chapter 2. In addition, in the `(S)ARIMAX_N` models, all 8 ACLED conflict indicators of all bordering neighbor countries are included with the values from the previous period $t-1$ as exogenous explanatory variables. Descriptive statistics for the at this point still untransformed absolute values of the ACLED indicators is given in Table 8 in the Appendix.

A second and last preprocessing step consists of a log transformation of all 8 ACLED indicators, as described by Equation (3.1). It follows that the state space ARIMA models also predict the absolute log fatality value y'_t. The advantage of the log transformation is that these absolute log forecasts can be converted to a relative log change value $y_{\Delta,t} = y'_t - y'_{t0} = log(\frac{y_t+1}{y_{t0}+1})$ by a simple subtraction with the logarithmic value of a base period $t0$. The error of the log change forecasts can now also be evaluated using the TADDA metric defined in Chapter 2.3.1, allowing for the assessment of the predicted direction of change and a better comparability of the now relative forecast errors among countries. An overview of the final ACLED indicators with their role and exact description is provided in Table 9 in the Appendix.

$$y'_t = log(y_t + 1), \qquad t = 1, ..., fh \qquad \text{with } fh \in \{4, 12\} \tag{3.1}$$

3.1.1 Analysis of Conflict Incidence and Country Categorization

Empirical findings from conflict theory claim that conflict data such as ACLED exhibit some particular characteristics: According to (Mueller & Rauh 2022, Collier & Sambanis 2002) the risk of conflict is not equally distributed across all countries. In fact, there is a strong dependence of future conflicts on conflict history. In general, there is very little risk for the outbreak of a violent conflict and thus for an increased number of conflict-related deaths. However, once a conflict erupts in a country, this country often remains stuck in a repeated cycle of violence, and an increased probability for the onset of new conflicts persists for several years after the start of the conflict. The conflict research community refers to this phenomenon as the conflict trap.

Figure 2 demonstrates that the ACLED dataset reflects these phenomena of the very low risk on one hand and the conflict trap on the other hand are very well by the average sum of monthly conflict fatalities. To eliminate the influence of different population sizes on the level of conflict fatalities, I normalize this figure to the average of monthly conflict fatalities per 1 million inhabitants. The map on the left side of Figure 2 indeed shows that in the majority of countries the conflict fatality incidence is very low. In 67 of 227 of the ACLED countries (29.52%), not a single conflict fatality has been recorded. And as the histogram on the right side of Figure 2 shows, the 90th-percentile already lies at an average monthly value of 3.28 fatalities per 1 million inhabitants. The distribution, thus, is extremely right-skewed and the number of conflict fatalities seems to accumulate in only a few countries, marked in red on the map.

In this work, I therefore not only want to examine the different state space ARIMA model configurations, but also whether a relationship can be established between the predictability of different countries and their conflict history. For this purpose, I divide the 227 ACLED countries into the three disjoint categories *no-conflict countries, low-conflict countries* and *high-conflict countries* shown in Table 2. I set the boundary between the *low-conflict* and *high-conflict countries* using the 90th-percentile, so that the 10% most conflict-ridden countries fall into the *high-conflict countries* category. The three maps in Figure 3 visualize the resulting classification of countries and their fatalities incidences. In Chapter 5.2, this categorization eventually allows the comparison

of the forecast errors of the state space ARIMA models across the different country categories and the derivation of possible patterns in the predictability of the forecasted countries.

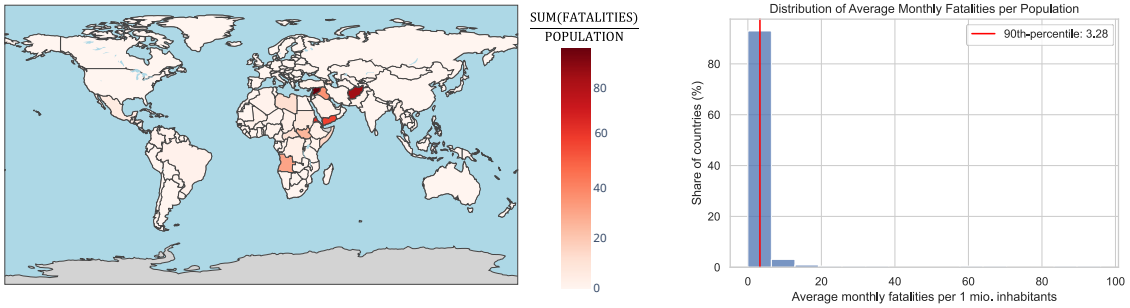

Figure 2: Map and histogram of average monthly conflict fatalities per 1 million inhabitants of the 227 ACLED countries. The map (left side) shows that conflict incidence is very low in the majority of countries, whereas very high numbers of conflict deaths accumulate in only a few countries such as Syria, Afghanistan, Yemen, Eritrea, Angola. The histogram (right side) further visualizes the heavily right-skewed distribution of average monthly conflict deaths normalized to population size.

Country category	**Avg. monthly fatalities per 1 mio. inhabitants**	**Total number of countries (Share in %)**
No-conflict countries	0	67 (29.52%)
Low-conflict countries	$(0, 3.28)$	137 (60.35%)
High-conflict countries	$[3.28, 96.01]$	23 (10.13%)
All countries	$[0, 96.01]$	227 (100%)

Table 2: Disjoint classification of the 227 ACLED countries according to their average number of monthly conflict fatalities into the categories *no-conflict, low-conflict* and *high-conflict countries*.

3.2 International Monetary Fund - World Economic Outlook Database (IMF-WEO)

The World Economic Outlook[2] is a report published by the International Monetary Fund (IMF) twice a year. It analyzes global economic developments and projects them five years into the future. The time series of this data set start in 1980 and end with the 5-year ahead projections in 2027, which I do not include in my models. Hence, the economic indicators are aggregated on country level with a annual periodicity. One advantage of the data set is that it also includes territories such as the West Bank & Gaza and Taiwan, which are not internationally recognized as sovereign states, but are certainly important to include for a good crisis early warning system.

I incorporate these macroeconomic time series, contrary to the ACLED indicators, as exogenous variables in the `(S)ARIMAX_SE` forecast models, in an attempt to include possible economic causes for the presence of deadly conflicts.

Since I develop monthly and quarterly models in this work, the IMF indicators must first be converted from an annual temporal aggregation to a monthly and a quarterly time series. This is done by simply filling the months and quarters with the indicator value of the corresponding year, so that the 12 months and 4 quarters of a year always have the same value.

[2]Retrieved from *https://www.imf.org/en/Publications/WEO/weo-database/2022/April* (2022, May 10)

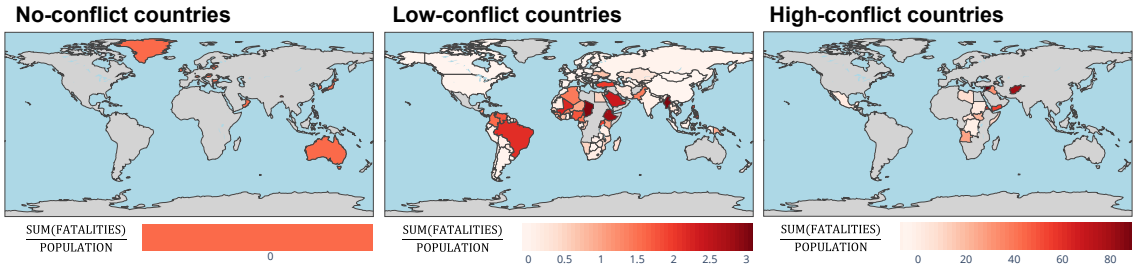

Figure 3: The geographical distribution of countries and their conflict incidence by the three country categories.

Some of the economic indicators, such as the GDP or the population, show a continuous (upwards) trend over the years, which however is to a large extent exclusively attributable to the progression of the years, the temporal dimension. To avoid these non-stationary indicators causing the models to incorrectly capture such spurious relationships and disrupt the predictions, I systematically sort out these indicators. Table 3 shows the resulting subset of the selected IMF indicators used in the models. In the following of this work, I assume that this subset of indicators satisfies a sufficient degree of stationarity.

IMF-WEO Indicator Code	Indicator Description
NGDP_RPCH	Gross domestic product, constant prices
NGAP_NPGDP	Output gap in percent of potential GDP
PPPSH	GDP based on purchasing-power-parity (PPP) share of world total
PPPEX	Implied PPP conversion rate
NID_NGDP	Total investment
NGSD_NGDP	Gross national savings
PCPIPCH	Inflation, average consumer prices
PCPIEPCH	Inflation, end of period consumer prices
TM_RPCH	Volume of imports of goods and services
TMG_RPCH	Volume of Imports of goods
TX_RPCH	Volume of exports of goods and services
TXG_RPCH	Volume of exports of goods
LUR	Unemployment rate
GGX_NGDP	General government total expenditure
GGXCNL_NGDP	General government net lending/borrowing
GGSB_NPGDP	General government structural balance
GGXONLB_NGDP	General government primary net lending/borrowing
GGXWDN_NGDP	General government net debt
GGXWDG_NGDP	General government gross debt
BCA_NGDPD	Current account balance

Table 3: The selected 20 of the originally 44 macroeconomic indicators of the IMF World Economic Outlook published in April 2022. These indicators serve as exogenous predictor variables in the `(S)ARIMAX_SE` and `(S)ARIMAX_SE+N` models.

3.3 World Bank - World Development Indicators (WB-WDI)

The World Development Indicators (WDI)[3] are a set of 1400 cross-country comparable indicators documenting socio-economic developments since 1960 with worldwide coverage. They are updated quarterly by the World Bank Group, which compiles data from officially recognized sources. Geographically, the WDI data set covers 207 countries, including the Palestinian territories, while Taiwan, however, is not considered separately. Like the IMF indicators, the WDI time series are provided aggregated on country level with annual periodicity.

[3]Retrieved from *https://datatopics.worldbank.org/world-development-indicators/* (2022, June 1)

Given the very high number of WDI indicators, I rely on a pre-selection of 56 WDI indicators identified as conflict-related by experts at the German Federal Foreign Office. For the use in the monthly and quarterly models, I conduct the same upsampling and indicator selection process used for the IMF indicators. Table 10 in the Appendix shows the resulting subset of the WDI time series that now have a monthly and quarterly periodicity and are again assumed to be stationary.

3.4 Variable Overview and Missing Data

Putting it all together, the endogenous variable y which is required for all ARIMA model variants consists only of the ACLED indicator `SUM(FATALITIES)` of the respective country and does not contain any months or quarters with missing values.
The matrix X of exogenous variables appears only in the `(S)ARIMAX` models, the regressions with (S)ARIMA error. X is composed of different combinations of the ACLED indicators of all neighboring countries and the IMF and World Bank indicators, depending on the model variant. The merged exogenous data sets always start simultaneously with the y time series in the year of the country-specific ACLED start date. See Table 7 in the Appendix for the country-specific start dates. While the ACLED indicators are assumed to be complete, missing data points do occur in the IMF and World Bank time series and thus in the versions of X in which these indicators appear. Figure 4 shows the different shares of missing data points that appear in the IMF indicators, the World Bank indicators, and the entire resulting monthly training datasets $X_{SE,train}$ of the `SARIMAX_SE` models. The histograms only contain the 125 of 227 ACLED countries for which a valid model could be calculated in the automated model building process in Chapter 4.4.

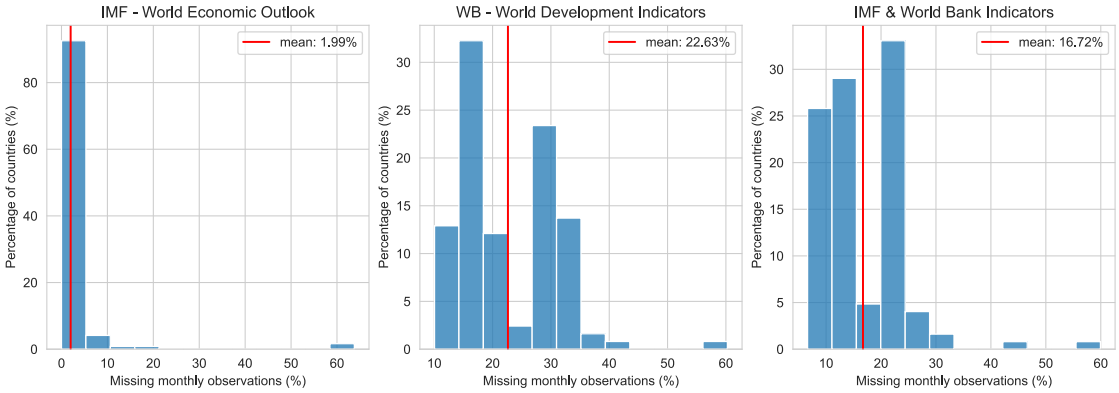

Figure 4: The distributions of missing monthly observations for the socio-economic indicators from the training set over the subset of the 158 analyzed ACLED countries. With a few exceptions, very few data points are missing from the IMF indicators (left histogram), with an average of 1.99%. In the World Bank's World Development Indicators (middle histogram), significantly more data points are missing, averaging 22.63%. Combined, the average proportion of missing data in the socio-economic indicators is 16.72% (right histogram).

Table 4 reveals how this affects the exogenous training data of the state space models: Since the socio-economic indicators have a combined average share of 16.72% of missing observations, the models `(S)ARIMAX_SE` are also missing the most exogenous training data points. The absence of missing data in ACLED reduces the share of missing exogenous indicators for the `(S)ARIMAX_SE+N`, while no training data at all is missing in the `(S)ARIMA, (S)ARIMAX_N` models.

The existence of missing values requires the imputation of the affected time series. To avoid data leakage, I perform the imputation after splitting the data into the training, validation and test sets. The exact imputation strategy and the problem of data leakage are discussed in detail in Section 4.2 `SARIMAX_PCA_LinInt()`.

	Composition of X	Avg. missing obs. in X_{train}
(S)ARIMA	No exogenous variables	-
(S)ARIMAX_N	ACLED neighbor indicators	0.0%
(S)ARIMAX_SE	IMF and WB indicators	16.72%
(S)ARIMAX_SE+N	ACLED neighbor, IMF and WB indicators	13.02 %

Table 4: Average share of missing exogenous observations in the training data sets for each of the 8 models. The share of missing data does not differ between monthly and quarterly models due to the only annual resolution of the IMF and World Bank data and the completeness of the ACLED indicators.

4 Implementation in Python

The implementation of the automated calculation of the ARIMA state space methods was realized in Python, following the object-oriented programming paradigm. With the two files *forecasters.py* and *grid_search.py* I split the global problem modularly: In the *forecasters.py* file, a general (S)ARIMA(X) state space model and a no-change baseline model are defined, each with the necessary functions for the model fitting and prediction process. n contrast, the automated and parallelized parameter selection and validation is performed separately by the `GridSearch_CV` class, which is defined in the *common_functions.py* file. In addition, there are several auxiliary functions that, for instance, allow the correct retrieval of the data or the calculations of the evaluation metrics.

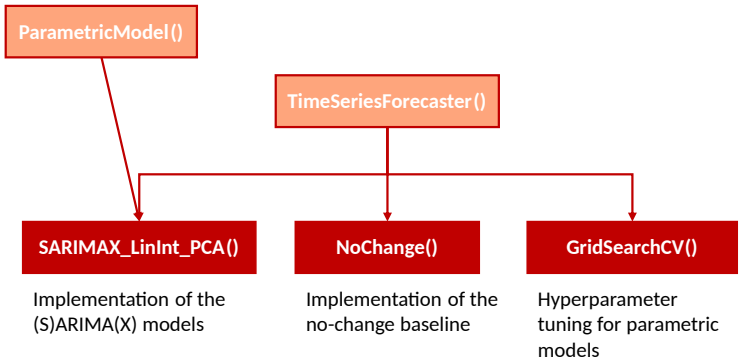

Figure 5: Hierarchy of the forecaster classes and interfaces. The light orange boxes contain the abstract classes and the dark red boxes correspond to the derived forecaster classes. The red arrows indicate the implementation of an interface.

4.1 No-Change Forecaster Class

The `NoChange()` class is kept very simplistic. In the method `fit()` only the last observation from `y_train` is stored as `y_t0`. Within the method `predict()` this last observation `y_t0` is then used to fill the $fh \times 1$ vector containing the no-change forecasts, where fh specifies the forecast horizon.

```
def predict(self, X_test, fh):
    y_hat0 = np.empty(fh)
    y_hat0[:] = self.y_t0 # last observed value carried forward
    return y_hat0
```

The `evaluate_model()` method retrieves the forecast from the `predict()` method and calculates the log-change by subtracting `y_t0` from the future y-values. Finally, the prediction results are visualized consistent with `GridSearch_CV.evaluate_model()` and returned as a table with all the relevant information for the evaluation.

```
def evaluate_model(self, X_test, y_test, fh):
    ...
    forecast = self.predict(X_test, fh)
    forecast = pd.DataFrame(forecast, index = X_test.index)
```

```
    ### LOG-CHANGE CALCULATION
    y_test_logchange = y_test - self.y_t0
    forecast_logchange = forecast - self.y_t0
    ...
    ### BUILD PREDICITON RESULT DF FOR MODEL EVALUATION
    table = pd.DataFrame()
    table["MONTH"] = y_test.index
    table["FAT_PRED"] = forecast.to_numpy()
    table["FAT_ACTUAL"] = y_test.to_numpy()
    # log change
    table["LC(FAT_PRED)"] = forecast_logchange.to_numpy()
    table["LC(FAT_ACTUAL)"] = y_test_logchange.to_numpy()
    ...
    return table
```

4.2 State Space ARIMA Forecaster Class

To enable the `SARIMAX_LinInt_PCA()` model and possible other future models to be used with the grid search from the class `GridSearchCV()`, a general structure of a *grid-searchable* parametric time series models must be specified. This is done by defining the abstract super class `ParametricTimeSeriesForecaster()`, which dictates implementations of the abstract methods `fit(self, X_train, y_train, params)`, `predict(self, X_test, fh, ci = False)` and `predict_insample(self)` for its subclasses.

The class `SARIMAX_LinInt_PCA()` shall cover all state space ARIMA variants mentioned in Chapter 2. In the case of the (S)ARIMAX models this includes the integration of exogenous regressors. Therefore, the method `fit()` first of all checks and preprocesses the input data. The endogenous variable originates from the ACLED data set and thus, as described in section 3.1, requires no further preprocessing. The exogenous variables, however, can cause even two problems for the (S)ARIMAX models: First, missing values may occur in the IMF and World Bank data sets, which are the source of the socio-economic regressors. Second, the high dimensionality of the X matrix with in many cases over 85 indicator columns may cause problems. As with standard regression models, a too high number of regressors may not only lead to higher computational effort, but also to overfitting, which can strongly affect the model performance.

To solve these problems, the matrix X of exogenous variables must be passed through a preprocessing pipeline. The three sequential steps of the pipeline, namely linear interpolation, standardization and dimensionality reduction in the form of a principal component analysis (PCA), are described in the following.

Linear Interpolation

In the first step, all columns that do not contain any values at all for any month or quarter are deleted from the matrix `X_train`. In the remaining columns, the gaps between two existing data points of a column are filled by linear interpolation. However, there are still missing values before the first known month/quarter and after the last month/quarter. These missing values at the margins are filled according to the first observation carried backward (FOCB) principle at the beginning of the time series or last observation carried forward (LOCF) at the end of the time series. (Adejumo et al. 2021)

```
### LINEAR INTERPOLATION
# delete all full NaN columns
if X_train_preprocessed is not None:
    X_train_preprocessed = X_train.dropna(axis=1, how='all')
    self.X_train_selected_cols = X_train_preprocessed.columns
```

```
6
7   # linear imputation
8   if X_train_preprocessed is not None:
9       X_train_preprocessed = X_train_preprocessed.interpolate(method="linear",
        ↪  axis=0, limit_direction="both")
10
11  if X_train_preprocessed.shape[1] == 0:
12      X_train_preprocessed = None
13
14  self.X_train_imputed = X_train_preprocessed
```

Standardization

In the second pipeline step, the exogenous variables are standardized using the sample mean and standard deviation from the imputed training data set. This step is essential prior to the principal component analysis, since in the PCA the variances of the individual variables are crucial for determining the principal components. Without standardization, the calculation of the principal components would be biased in favor of exogenous variables with high variance. After standardization, the transformed X'_{train} is column-wise mean zero and has a unit variance.

$$x'_{i,j} = \frac{x_{i,j} - \overline{x}_j}{\sigma_{x_j}} \tag{4.1}$$

In the code, this is done by fitting a `StandardScaler()` object from the scikit-learn package (Pedregosa et al. 2011) on the training data `X_train`, which is later reused in the prediction to transform the test data `X_test` the same way.

```
1   ### TRANSFORMATION OF X: StandardScaler
2   if X_train_preprocessed is not None:
3       self.scaler = StandardScaler()
4       X_train_preprocessed = self.scaler.fit_transform(X_train_preprocessed)
```

Principal Component Analysis

The principal component analysis (PCA) is a dimensionality reduction method that calculates a lower dimensional representation of the K exogenous variables in the form of p principal components. The first of these principal components is calculated as the linear combination of all exogenous variables with the maximum variance. The second principal component is the linear combination that is orthogonal to the first principal component and again maximizes the variance under this condition. This principle is continued for the subsequent principal components: The i-th principal component is the linear combination that is orthogonal to all preceding principal components and thereby maximizes the variance.

To now reduce the dimensionality of the K-dimensional vector space, only the first $p < K$ principal components are kept. The matrix X''_{train} resulting from the reduction is now only p-dimensional and explains the following proportion of the variance of the initial matrix X'_{train} depending on the principal components involved. (James et al. 2021)

$$\text{Proportion of Variance Explained (PVE)} = \frac{\sum_{i=1}^{p} Var(PC_i)}{\sum_{j=1}^{K} Var(x^{(j)})} \in (0, 1) \tag{4.2}$$

```
1   ### DIMENSIONALITY REDUCTION OF X: PCA
2   if n_components != 0 and X_train_preprocessed is not None
3       and X_train_preprocessed.shape[1] >= n_components:
4       self.pca = PCA(n_components=n_components)
5       X_train_preprocessed = self.pca.fit_transform(X_train_preprocessed)
```

```
else:
    X_train_preprocessed = None
```

Model Fitting

The final step in the `fit()` method is the actual fitting of the (S)ARIMA(X) model. This is implemented by passing the case specific (S)ARIMA(X) parameters to the `SARIMAX()` class of the statsmodels package (Seabold & Perktold 2010, Fulton 2015).

```
### (S)ARIMA(X)
if y_train is None:
    self.fitted_model = None
else:
    try:
        self.fitted_model = SARIMAX(endog = y_train,
                                    exog =   X_train_preprocessed,
                                    order = order,
                                    seasonal_order = seasonal_order,
                                    trend = trend
    ).fit(disp=0)
except:
    self.fitted_model = None
```

Data Leakage Problem

This pipeline approach ensures separate preprocessing of `X_train` and `X_test` and thus prevents information derived from the test data set from being used for model training. If the data set of the exogenous variables were already interpolated, scaled and dimension-reduced before separation, a data leakage problem would occur: Incorporating data points from test observations in the preprocessing of `X_train` can potentially provide a better interpolation, more accurate estimations of expected values and variances for the standardization, and a better computation of the principal components. However, this information from the test data is in fact not available to the model in productive operation, as it lies in the unknown future. The incorrect inclusion of this additional future information, thus, in the worst case, leads to a negative bias in the estimation of the prediction errors in the form of TADDA, MAE or RMSE. Overall, the actual predictive power of the model in productive operation use would be overestimated. Regardless, in the case of this kind of data leakage, it will also be difficult to compute the predictions in productive operation, since the test data will simply not be available at that point of time. (Kapoor & Narayanan 2022)

Prediction

After the model has been fitted, the `predict()` method can be called to forecast future values of y. Here, the test data `X_test` must now pass through a similar preprocessing pipeline with some differences to the fitting pipeline. Before imputation, it must be ensured that the full NaN columns removed from `X_train` are also removed from `X_test`, regardless of whether values exist in one of these columns in `X_test`. By this, exactly the same exogenous variables are used for prediction.

```
### IMPUTATION & FEATURE SYNCHRONIZATION ###
# feature synchronization
if X_test_preprocessed is not None:
    X_test_preprocessed = X_test[self.X_train_selected_cols]
```

Since the values of the exogenous variables in `X_test` are as little known in operational use as the endogenous variable y to forecast, `X_test` is here completely filled with NaN values. This

procedure again prevents the model from making predictions during testing which themselves are already based on future information.

```
### DATA LEAKAGE PREVENTION ###
X_test_preprocessed.iloc[:, :] = np.nan
```

Thus, all values from `X_test` must be imputed without exception. As a linear imputation is not possible, the columns are filled according to the LOCF principle with the last known observation from the imputed training data set.

```
nan_cols = X_test_preprocessed.columns[X_test_preprocessed.isnull().all(axis=0)].tolist()
X_test_preprocessed[nan_cols] = self.X_train_imputed[nan_cols].iloc[-1, :]
```

In the following, the same standardization is performed with the sample mean and sample standard deviation from the training dataset and then the same PCA transformation based on the fitted object `pca` from the `fit()` method is applied.

```
### TRANSFORMATION: StandardScaler
if X_test_preprocessed is not None and self.scaler is not None:
    X_test_preprocessed = self.scaler.transform(X_test_preprocessed)

### PCA
if self.pca is not None and X_test is not None:
    X_test_preprocessed = self.pca.transform(X_test_preprocessed)
else:
    X_test_preprocessed = None # if n_components==0
```

The prediction is again performed using the statsmodels `SARIMAX()` class (Seabold & Perktold 2010, Fulton 2015). Negative predictions are set to zero manually, as negative log-transformed fatalities make no sense.

```
### ARIMA
forecast = self.fitted_model.get_forecast(steps=fh, exog=X_test_preprocessed)
y_pred = forecast.predicted_mean
# set negative predictions to zero
y_pred = y_pred.clip(lower=0)
if ci == True:
    conf_int = forecast.summary_frame(alpha=0.1)
    return y_pred, conf_int
else:
    return y_pred
```

4.3 Grid Search Class

The class `SARIMAX_PCA_LinInt()` used for the implementation is able to train an arbitrarily specified (S)ARIMA(X) model. However, it expects the specification of the SARIMA parameters $(p, d, q)\times(P, D, Q, s)$ as well as the number of principal components $n_{components}$ for the specification of the dimensionality reduction in the PCA. The problem of finding the best values of these hyperparameters is solved by the class `GridSearchCV()`. For this purpose, a forecaster of the class `ParametricTimeSeriesForecaster()` must be passed whose `fit()` and `predict()` methods can be accessed by the `GridSearchCV()` instance for the optimization. The `GridSearchCV()` instance thereby becomes a forecaster itself by finding the optimal hyperparameters and then training the parametric forecaster in its own `train()` method. The selection of the hyperparameters is carried out by a grid search with an extending window cross validation strategy.

Training

After the initialization of a `GridSearchCV()` object, the training method can be called to perform an exhaustive, cross validated search over the hyperparameter grid. The cross validation is only feasible if the training data `y_train` contains at least two years of data $MIN\_OBS\_CV = 2 \times s$, which corresponds to 24 monthly or 8 quarterly observations. This is because at least one year for model training and one year for validation are required in cross validation, as shown in Figure 6. Together with the withheld 1 year test dataset `y_{test}`, it follows that for countries with less than 3 years (36 months or 12 quarters) of available ACLED data, a grid search is not feasible.

With sufficient data being available, the preparation of the cross validation follows. The number of folds on which the hyperparameter combinations are validated depends on the data availability and the length of the forecast horizon, which in my case corresponds to the seasonal periodicity *sp*. To ensure proper feasibility, the number of folds is bounded to a minimum of 1 fold in the case of very little training observations and a maximum of 5 folds in the case of many training observations. Formula 4.3 describes this relationship.

$$\texttt{folds} = min\left\{\frac{\texttt{len(y\_train)}}{sp} - 1, \texttt{MAX\_FOLDS}\right\} \tag{4.3}$$

The number of folds is used in the next step to generate cut-off indices (`cv_cutoffs`) that mark the boundary between training and validation set in the individual folds.

$$\begin{aligned}\texttt{cv\_cutoffs} = \{&\texttt{len(y\_train)} - 5sp, len(\texttt{y\_train}) - 4sp, \texttt{len(y\_train)} - 3sp,\\ &\texttt{len(y\_train)} - 2sp, \texttt{len(y\_train)} - 1sp\}\end{aligned} \tag{4.4}$$

```
MAX_FOLDS = self.cv_folds
folds = min(int(len(y_train) / self.seasonal_periodicity) - 1, MAX_FOLDS)
cv_cutoffs = []
for f in range(1, folds+1):
    new_cutoff = len(y_train)-1 - f*self.seasonal_periodicity
    cv_cutoffs.append(new_cutoff)
cv_cutoffs.reverse()
```

With this list of cut-off indices given by Equation 4.4, the cross validation can now be started for all hyperparameter combinations by calling the parallelized method `cross_validate()`. This method returns a cross validation score for each hyperparameter combination. Based on the cross validation score, the optimal hyperparameter combination is identified as the one with the lowest cross validation score.

```
# list of cross validation results
tasks = (delayed(self.cross_validate)(X_train, y_train, params, cv_cutoffs)
        for params in tqdm_notebook(self.param_grid))

# save cv results
self.cv_results = executor(tasks)
```

Finally, the actual model training can begin, in which the passed parametric forecaster with the identified optimal hyperparameters is retrained on the full training data set. The `GridSearchCV()` instance is now in control of a trained forecaster and can therefore be treated as trained itself.

Extending Window Cross Validation

The method `cross_validate()` loops through the folds generated by the `train()` method: For each fold it splits the training data `y_train` into a smaller training set `y_train_cv` and a holdout data set for validation `y_val_cv` according to the calculated cut-offs from `train()`.

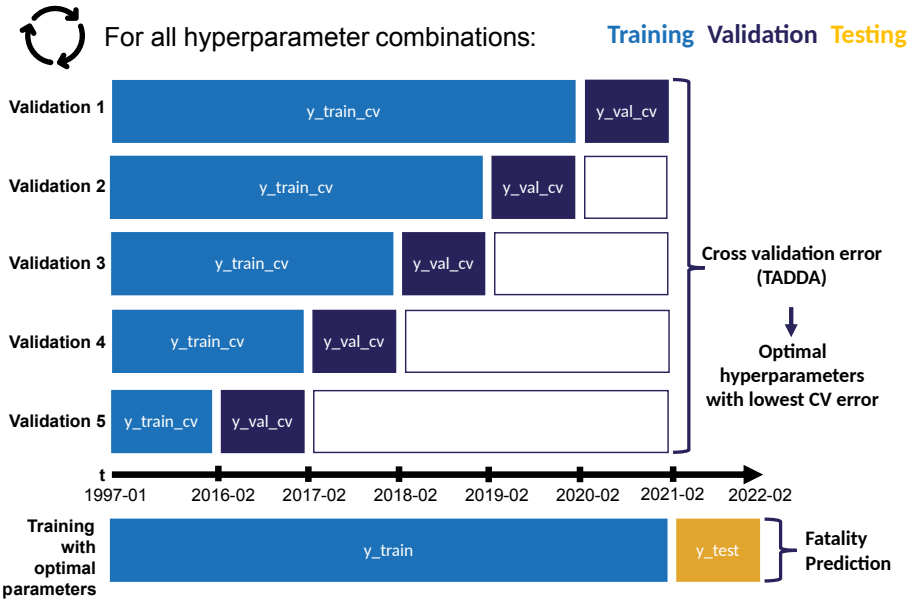

Figure 6: Visualization of the 5-fold expanding-window cross-validation. For countries with enough data available, a 5-fold expanding-window cross validation is performed. Countries with too little data for 5 folds are validated only with the last 1-4 folds.

```
FH = self.seasonal_periodicity
for fold, cutoff in enumerate(cv_cutoffs):
    X_train_cv = X_train.iloc[:cutoff]
    X_val_cv = X_train.iloc[cutoff:cutoff+FH]
```

Figure 6 shows that the size of the training data set `y_train_cv` grows with each fold by the forecast horizon, here 1 year, for which reason this validation approach is called extending window cross validation. The validation is accomplished by iteratively training the model with the same given hyperparameter combination on each different training dataset. Subsequently, each of the models provides an out-of-sample forecast for the respective validation period, here the last 1-5 years before the test data set.

```
model = self.forecaster()
model_fit = model.fit(X_train_cv, y_train_cv, params)
y_pred = model.predict(X_test = X_val_cv, fh = len(y_val_cv))
```

Using the TADDA metric from Chapter 2.3.1, the forecasts can be matched against the corresponding validation data set to calculate an average cross validation score across the 1-5 folds. The cross validation score is finally reported back to the caller method `train()`, where it is compared to the scores of the other hyperparameter candidates.

$$cv\_score = \frac{1}{folds} \sum_{i=1}^{folds} TADDA(y\_val\_cv_{\Delta,i}, f_{\Delta,i}, \epsilon = 0.048) \tag{4.5}$$

The purpose of this more complicated approach is to obtain the best possible estimate for the true out-of-sample prediction error, and thus the performance of the hyperparameter combination. In contrast to a simple 1-fold validation approach, this 1-5 fold extending window cross validation reduces the risk that a hyperparameter combination might perform well in the validation by pure chance, although it actually leads to much worse model predictions.

Model Evaluation

The method `evaluate_model()` is similar to its counterpart in the class `NoChange()` with the only difference that the `GridSearchCV()` instance does not have its own `predict()` method. The `GridSearchCV()` instance therefore relies on the predict method of the parametric forecaster `self.best_model()`, whose hyperparameters have been optimized in the `train()` method.

```
forecast, conf_int = self.best_model.predict(X_test = X_test,
                                              fh = len(y_test),
                                              ci = True)
```

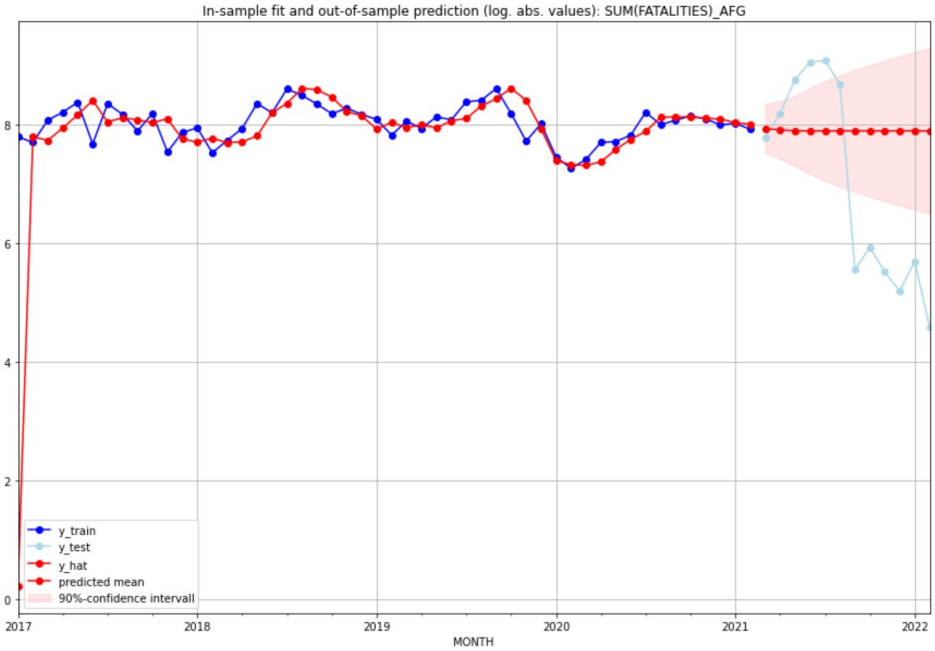

Figure 7: In-sample fit and out-of-sample forecast of the log absolute number of conflict fatalities with the `ARIMAX_SE_MONTHLY` model for Afghanistan.

4.4 Automated Model Building Process

Finally, the classes described in the previous sections are used to fit each of the models from Chapter 2 for all 227 countries of interest and evaluate its predictive power. Figure 8 visualizes how the interaction of the programmed classes and auxiliary functions makes this process completely automated and exclusively data-driven.

At the beginning, the desired model is specified. For this purpose, seasonality `S`, inclusion of socio-economic indicators `X_SE`, inclusion of ACLED neighbor indicators `(X_N` can each be enabled or disabled and the desired periodicity of the forecasts (12 for monthly, 4 for quarterly forecasts) can be specified. In addition, any ACLED variable, such as the number of protests, security related incidents, or simply the number of conflict events, can be set as the target variable y. In this work, however, I limit myself to predicting the change in the number of fatalities. For the `ARIMAX_N_MONTHLY` the configuration is as follows:

```
TARGET_VARIABLE = "SUM(FATALITIES)"
SEASONAL_PERIODICITY = 12 # for data frequency & seasonality S

```

```
SEASONALITY = False # S
NEIGHBORS = True # X_N
SOCIO_ECO_VARS = False # X_SE
```

Also, the maximum number of folds for the cross-validation is consistently set to 5 for all models.

Subsequently, the hyperparameter grid is defined with all parameter combinations to be tested. The following hyperparameter space is searched with the grid search, as far as the respective hyperparameter occurs in the model. $n_{components}$ stands for the number of PCA principal components to be used in the case of exogenous predictors.

$$grid = n_{components} \times p \times d \times q \times P \times D \times Q \times s \tag{4.6}$$

- $n_{components} \in \{0, 1, 2, 3\}$
- $p \in \{1, 2, 3\}$
- $d \in \{0, 1\}$
- $q \in \{1, 2, 3\}$
- $P \in \{1, 2, 3\}$
- $D \in \{0, 1\}$
- $Q \in \{1, 2, 3\}$
- $s \in \{12\}$

That is followed by the actual fitting process, repeated in a loop for each country:

1. Querying the function `getData()` returns the exogenous and endogenous variables X, y for the desired country at the desired temporal resolution.

```
### GET DATA
y, X = getData(target_variable = target_variable,
               target_country = target_country,
               predictor_countries = predictor_countries,
               socio_eco_vars = SOCIO_ECO_VARS,
               n_lags_X = N_LAGS_X,
               seasonal_periodicity = SEASONAL_PERIODICITY)
```

2. The function `train_test_split()` splits off the last 12 months or 4 quarters of X and y as an out-of-sample test dataset, as shown in Figure 6. This allows for an unbiased estimation of the prediction error in the model evaluation.

```
### SPLIT DATA
X_train, X_test, y_train, y_test = train_test_split(y = y,
                                                    X = X,
                                                    forecast_horizon =
                                                    ↪ SEASONAL_
                                                    ↪ PERIODICITY)
```

3. This is followed the hyperparameter grid search and model training through the interaction of the `GridSearchCV()` instance `gscv` with the forecaster `TimeSeriesForecasterPCA_LinInt` above.

```
### TRAINING WITH GRIDSEARCHCV
gscv = GridSearchCV(param_grid = PARAM_GRID,
                    forecaster = TimeSeriesForecasterPCA_LinInt,
                    seasonal_periodicity = SEASONAL_PERIODICITY,
                    cv_folds = CV_FOLDS)
training_res = gscv.train(X_train, y_train)
```

4. If the grid search and model fitting are successful, the obtained model is evaluated on the test dataset and the results are saved. If the grid search fails due to insufficient data availability or other irregularities (see `GridSearchCV()`), the `NoChange()` model is used as a replacement. This ensures that there is a working forecast model for each country. However, in Chapter 5, all countries with a failed grid search for any model are filtered out to ensure comparability between different countries.

```
    if training_res is not None:
        model = gscv
        ...
    else:
        model = NoChange().fit(X_train, y_train)
        best_parameters = np.nan

    ### PREDICTION
    prediction_results = model.evaluate_model(X_test, y_test, fh =
    ↪ SEASONAL_PERIODICITY)
    ...
    return prediction_results
```

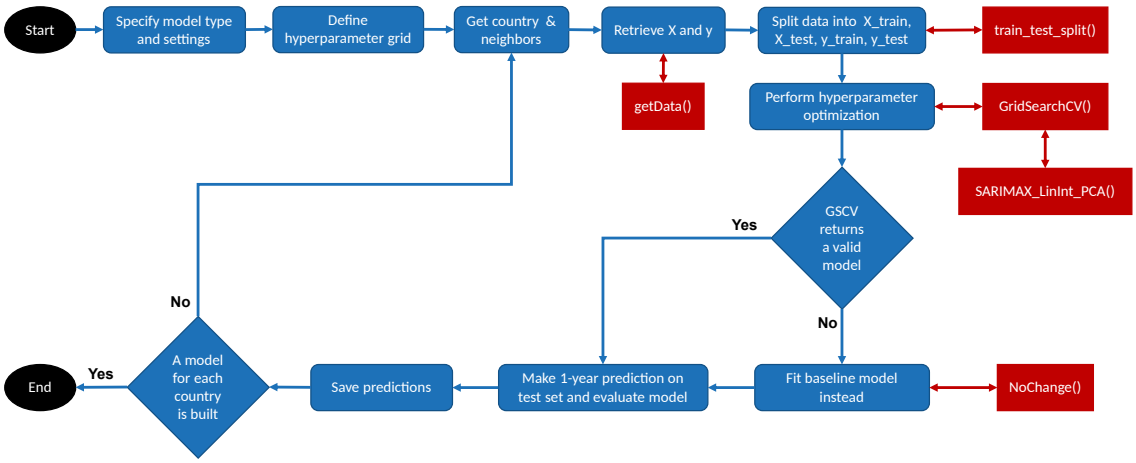

Figure 8: The interplay of all classes and functions for the automated generation of forecasts for all countries. Red boxes indicate the use of own python classes and functions.

5 Results

After running through the automated model building process, the 1-year forecasts of the 18 state space ARIMA models, including the no-change forecasts, are available for all countries. In this chapter, I conduct a detailed analysis of the different forecasts and identify patterns in the occurrence of forecast errors using the metrics TADDA, MAE and RMSE defined in Section 2.3.

For doing so, in the first section, I examine at a global level how the ARIMA models perform overall compared to the no-change baseline. Furthermore, from the 16 state space ARIMA variants, I identify the factors that contribute to a global improvement in the prediction of conflict fatalities.

In the second section, I analyze the model forecast results at country level. I identify the best and worst predictable countries using state space ARIMA methods compared to the baseline, and show patterns in the distribution of prediction errors using the geographic region and the categorization of ACLED countries from 3.1.

In the analysis of the model results in the following sections, however, it must be taken into account that the in the grid search integrated fitting and prediction process of the state space models, for the reasons described in section 4.4, is not always successful for all countries and models. If the computation of a state space model fails, the no-change model is used instead to obtain a model with global coverage in all cases. Table 5 shows that the percentage of failed model fits is relatively high, ranging from 27.31% to 29.07% depending on the model.

Model	**Share of no-change fits**	**Share of successful (S)ARIMA(X) fits**
ARIMAX_N_MONTHLY	27.31%	72.69%
ARIMAX_N_QUARTERLY	29.07%	70.93%
ARIMAX_SE+N_MONTHLY	27.31%	72.69%
ARIMAX_SE+N_QUARTERLY	27.75%	72.25%
ARIMAX_SE_MONTHLY	27.31%	72.69%
ARIMAX_SE_QUARTERLY	27.75%	72.25%
ARIMA_MONTHLY	27.31%	72.69%
ARIMA_QUARTERLY	28.63%	71.37%
NO_CHANGE_MONTHLY	100.0%	0.0%
NO_CHANGE_QUARTERLY	100.0%	0.0%
SARIMAX_N_MONTHLY	27.31%	72.69%
SARIMAX_N_QUARTERLY	28.19%	71.81%
SARIMAX_SE+N_MONTHLY	27.31%	72.69%
SARIMAX_SE+N_QUARTERLY	27.75%	72.25%
SARIMAX_SE_MONTHLY	27.31%	72.69%
SARIMAX_SE_QUARTERLY	27.75%	72.25%
SARIMA_MONTHLY	27.31%	72.69%
SARIMA_QUARTERLY	27.75%	72.25%

Table 5: The share of countries for which the grid search failed and succeeded. With a successful grid search, the desired (S)ARIMA(X) state space model was computed. In case of failure a no-change baseline model was fit instead.

To ensure that the model comparisons are not biased by a higher proportion of no-change predictions, in the following analyses only the 158 (69.60%) of 227 ACLED countries with successful fits for all state space model variants are considered.

5.1 Global Model Performances

First, the average prediction errors of the 18 models are analyzed at a globally aggregated level. Figure 9 visualizes the quantification of the model's prediction errors using the three metrics

TADDA (left), MAE (middle), and RMSE (right) described in Section 2.3. As the models calculate the future log change in conflict fatalities based on the last known observation, the errors have to be interpreted as the deviation of the predicted from the true change in conflict fatalities for the forecasted time periods.

TADDA, MAE, and RMSE are all in the same log-change unit but, as indicated by the formulas in Section 2.3, penalize errors of different magnitudes to different degrees. For example, the TADDA prediction errors, which contain an additional directional penalty term, range from 0.476 to 0.740. As expected, these errors turn out to be higher than the MAE values, which range from 0.452 to 0.642. The only exception are the no-change models, which never predict any change. The RMSE values with a range of 0.586 to 1.037 are on average significantly higher than the TADDA and the MAE values, since here the squaring penalizes large deviations in the prediction particularly strongly.

Forecasts become more inaccurate the further they reach into the future.

In Figure 10, the forecast error of the individual models is plotted over the one-year prediction periods starting in March 2021. For a better overview, only the monthly models are plotted. It can be observed that, despite some interim fluctuations, the average prediction error for all three metrics shows a clear upward trend as predictions are made further into the future. The quarterly models exhibit a similar temporal trend which is depicted in Figure 20 in the appendix. The quality of the state space ARIMA forecasts, as well as the no-change forecasts, thus, strongly depends on the forecast horizon. The better performance of the short-term ARIMA forecasts can be explained by the structure of the ARIMA equations under assumed stationarity conditions: While the ARIMA forecasts slowly converge towards the sample mean (unconditional mean), the forecast error increases with each step into the future and eventually converges towards the sample variance. The ARIMA state space models are therefore more suitable for short term forecasts.

State space ARIMA methods can be a very competitive framework for conflict prediction.

To judge the forecast quality of the 16 tested state-space ARIMA models, I start with comparing their average prediction error to the no-change baseline model. It can be noticed that naïve baseline predictions of the monthly no-change model, which turn out to be by far better than the quarterly no-change predictions, are in general very competitive with an average error of 0.390 (TADDA and MAE) and 0.539 (RMSE) calculated across all 158 analyzed countries. As Figure 9 shows, only 3 of the 8 monthly models, namely `ARIMAX_N_MONTHLY`, `ARIMAX_SE_MONTHLY`, `ARIMA_MONTHLY` produce better predictions than the monthly baseline model `NO_CHANGE_MONTHLY`. Especially in the TADDA score, where no-change predictions are the only ones that are not penalized, the lead of the best model `ARIMAX_SE_MONTHLY` over the baseline `NO_CHANGE_MONTHLY` shrinks to as little as -1.78%. This is despite the fact that in the cross-validated grid search in Section, described in 4.3, the (S)ARIMA(X) hyperparameters were optimized by minimizing the TADDA cross-validation score.

Also, the modeling results from the ViEWS predictions show that generating better predictions than the no-change model is not trivial. Vesco et al. (2022) find in the evaluation of the competition that none of the 9 country-month models submitted is able to beat the no-change baseline. It should be noted that contrary to the global forecasts of this work, in the ViEWS prediction competition only the African countries were predicted.

The relatively good performance of the no-change model seems to lie in the nature of the conflict prediction problem: Because of the low baseline risk of conflict onset described by Mueller & Rauh (2022), projecting the peaceful status quo is a prediction strategy that proves to be extremely successful in most of the rather peaceful countries. In contrast, conflict-affected countries, according to Collier & Sambanis (2002), are often stuck in the conflict trap explained in the previous Chapters 2 and 3.1.1. In these cases, carrying forward an existing number of conflict deaths is therefore also a strategy that is in line with conflict theory. The onset of new conflicts in previously peaceful

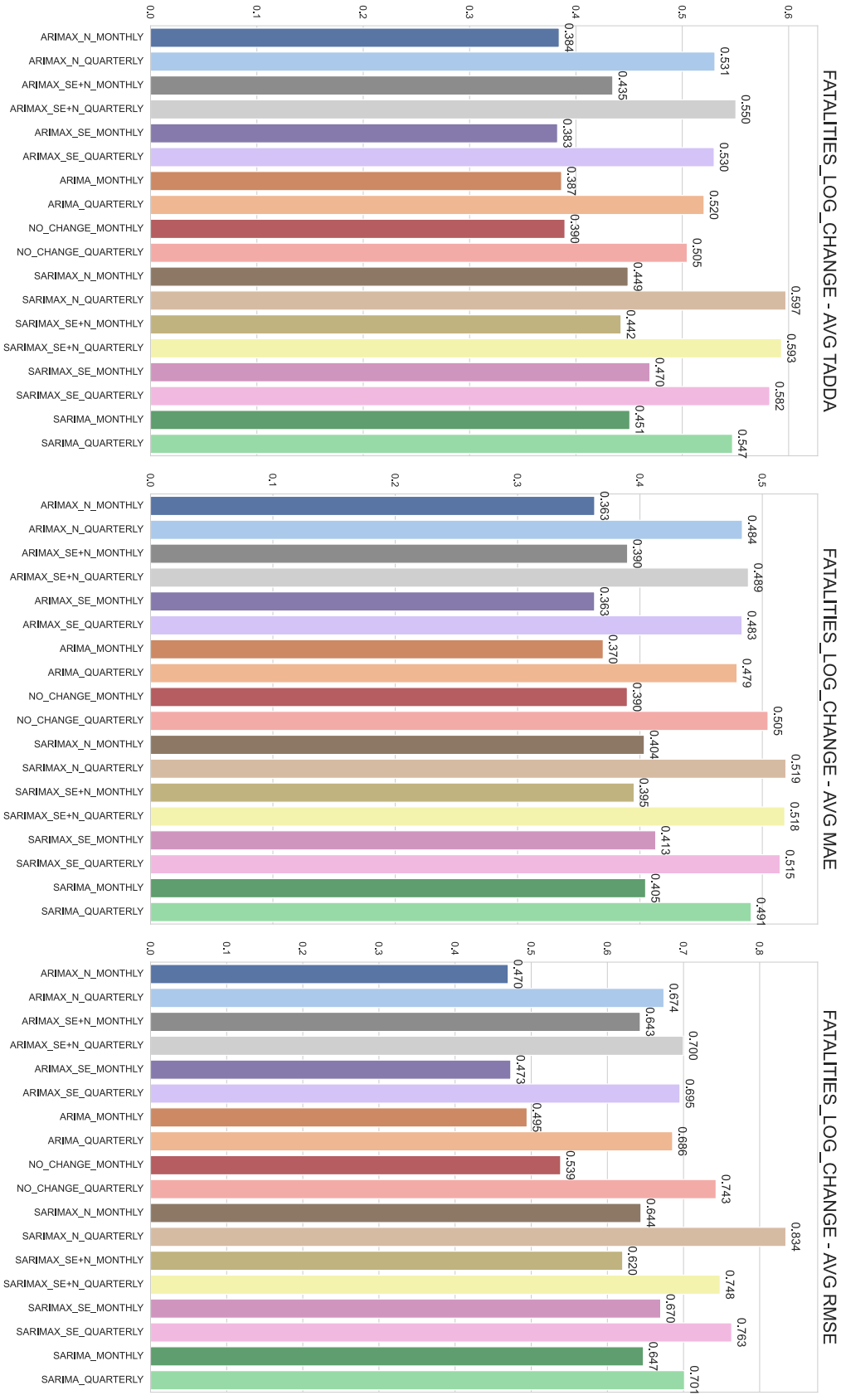

Figure 9: Average out-of-sample TADDA, MAE and RMSE prediction errors aggregated over all countries and prediction periods.

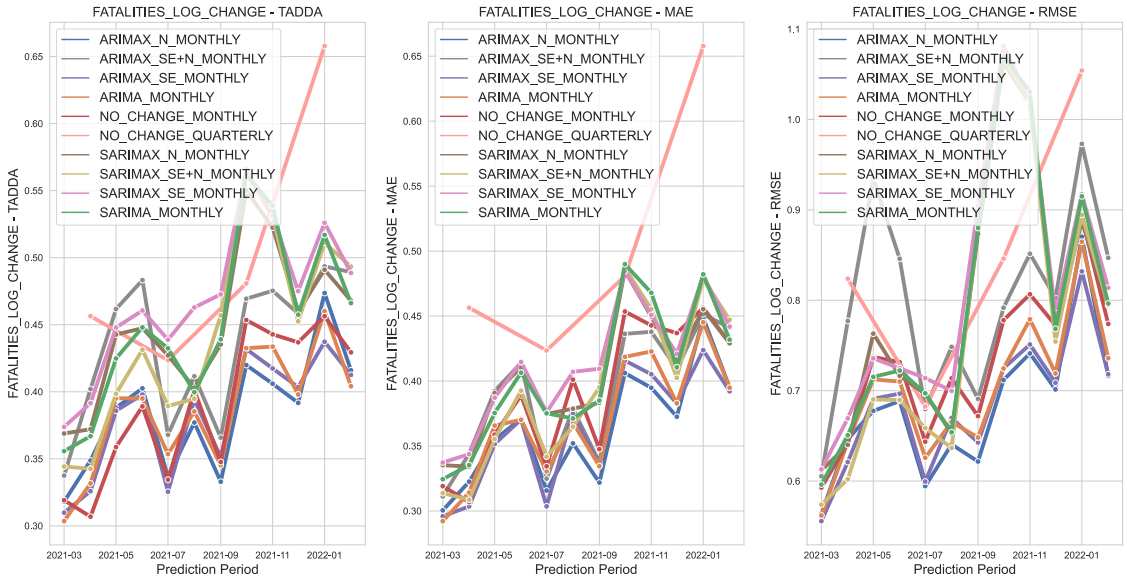

Figure 10: Out-of-sample prediction error of the monthly models over the 12 prediction periods. The red lines mark the monthly and quarterly no-change baseline for a comparison of the model performances.

countries and the escape from the conflict trap, however, are very rare events (Mueller & Rauh 2022, Collier & Sambanis 2002, Collier et al. 2003).

Despite the fact that the baseline model was only narrowly outperformed, it can be concluded from the just stated difficulty of the prediction problem and the results of the ViEWS prediction competition that the state space ARIMA approach presented in this thesis is very competitive when compared to other state-of-the-art models. A detailed analysis of differences in conflict predictability across different categories of countries is provided in Section 5.2.

The monthly models clearly outperform the quarterly models.

Looking again at the global average errors in Figure 9, it can be observed that the forecast errors of the monthly models (darker colors) behave relatively to each other like the errors of the quarterly models (lighter colors). However, it is immediately apparent all the monthly models perform much better than their quarterly counterparts in each metric. Even the best quarterly model does not manage to beat the worst monthly model. The heatmap in Figure 11 demonstrates this gap by comparing the aggregated average errors of all monthly and all quarterly models. It shows that, on average, the prediction error of the monthly models is 23.63% lower in the TADDA score, 22.02% lower in the MAE and 19.65% lower when measure in the RMSE.

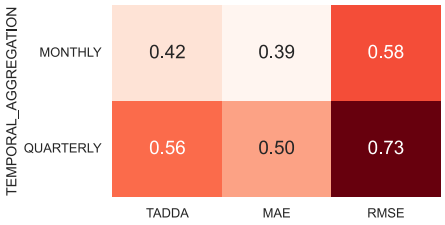

Figure 11: Comparison of the average forecast performance of the monthly models and the quarterly models. The forecasts of the monthly models are on average 23.63%, 22.02% and 19.65% better in the TADDA metric, MAE and RMSE.

The relatively poor performance of the quarterly models can be explained by the fact that the quarterly input data contains a by the factor 3 smaller number of observations, which leads to a higher uncertainty in the estimates of the model coefficients. In addition, the aggregation of three months of ACLED events into a single data point is naturally accompanied by a substantial loss of information. For the socioeconomic indicators, however, no information is lost through quarterly aggregation, as the IMF and World Bank indicators all are provided with an only annual temporal resolution.

The integration of seasonality disturbs the model's ability to make good predictions.

After a very clear statement could be made in favor of the monthly models, I investigate how seasonality affects the forecast quality. To do so, I split the monthly and quarterly models according to whether they belong to the ARIMA or SARIMA state space variants. The resulting average forecast errors are shown in the heatmap in Figure 12. As the heatmap illustrates, the integration of multiplicative seasonal components causes the prediction errors to increase significantly. More precisely, the non-seasonal monthly models achieve an average TADDA score of 0.40, MAE of 0.37 and RMSE of 0.52. After incorporating seasonality into the same models, the performance is on average 12.50%, 8.11% and 25.00% worse with a TADDA score of 0.45, MAE of 0.40 and RMSE of 0.65. Thus, my approach of modelling seasonality seems to impede the model's ability to make good predictions in the majority of countries. Although in conflict theory signs for seasonal effects, such as the harvest effect pointed out by Guardado & Pennings (2020), could be found, the seasonality in this work seems to be modeled too rigidly as a fixed dependence of a forecasting period on the same period of previous years. In reality, however, seasonality might occur with different non-annual periodicities and might change over time. In addition to that, the magnitude of seasonal effects can vary considerably across the country time series. Since the model fitting is not preceded by any seasonality check, seasonality is also imposed on possibly non-seasonal time series. This false additional complexity then acts as noise in these time series and increases the variance of the prediction errors.

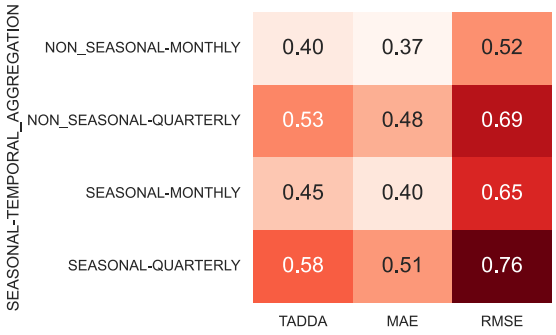

Figure 12: Comparison of out-of-sample forecast errors of the seasonal and non-seasonal models for both the monthly and quarterly models.

Included individually, the ACLED neighbor indicators and the socioeconomic indicators both improve the predictions.

Not only does integrating seasonality interfere with conflict predictions. In general, an increased model complexity also seems to be related to higher average forecasting errors: Figure 13 shows that excluding the poorly performing seasonal models, only the tier 0 model `ARIMA_MONTHLY` and the tier 1 models beat the monthly no-change model. The individual inclusion of the ACLED neighbor indicators (`X_N`) or the socioeconomic indicators (`X_SE`) leads to a slight improvement with all three metrics over the Tier 0 `ARIMA_MONTHLY` model. Thus, with seasonality excluded,

the low complexity tier 1 models provide the best fatality prediction results. However, when using the combination `X_SE+N` of the ACLED neighbor indicators and the socio-economic indicators, the predictions worsen significantly.

It can be concluded that the integration of both types of exogenous variables can be beneficial in principle. However, in the nonseasonal monthly models, the ACLED country indicators and the socioeconomic country variables seem to interfere with each other. This could be due to the encounter of two different temporal resolutions in the PCA. While the usually few ACLED neighbor indicators vary monthly, the more numerous socio-economic variables usually change only once a year. This may complicate the meaningful calculation of monthly-resolved principal components in the PCA.

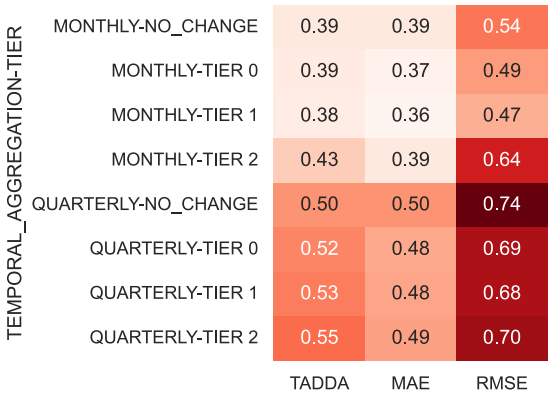

Figure 13: Average out-of-sample prediction errors of the non-seasonal models by model complexity for the monthly and quarterly models separately.

5.2 Country-Level Model Performances

After identifying the model specifications that are essential for a successful fatality forecast, in this chapter I investigate whether and which country-specific properties can be associated with good predictability by the state space ARIMA methods.

To do so, I first take a look at the prediction errors of the in the previous section identified best model variant `ARIMAX_SE_MONTHLY`. Second, I compare these prediction errors to those of the monthly no-change model `NO_CHANGE_MONTHLY` at country-level and then at a regional level of aggregation. Finally, I use the country categorization from Section 3.1 to examine whether there is a relationship between the average level of conflict and the predictability of a country by state space ARIMA methods.

The best model outperforms the no-change baseline in only 17% of the countries.

First, I take a look at the distribution of the TADDA prediction error of the `ARIMAX_SE_MONTHLY` model, that was identified as the best model, in the analyzed countries. As the map in Figure 14 shows, some dark red colored countries stand out as the worst predicted countries. These countries include, in descending order by TADDA score, Myanmar (TADDA: 2.69), Kenya (2.36), Turkey (1.89), Afghanistan (1.62), and Saudi Arabia (1.41). With regard to the overthrow of the regime in Afghanistan by the Taliban in August 2021 and the military coup in Myanmar in February 2021, it seems that, at least for these two countries, poor predictability can be attributed to the high conflict dynamics in the forecasting period.

For further analysis, I examine how well the `ARIMAX_SE_MONTHLY` model performs at the country level compared to the baseline, `NO_CHANGE_MONTHLY`.

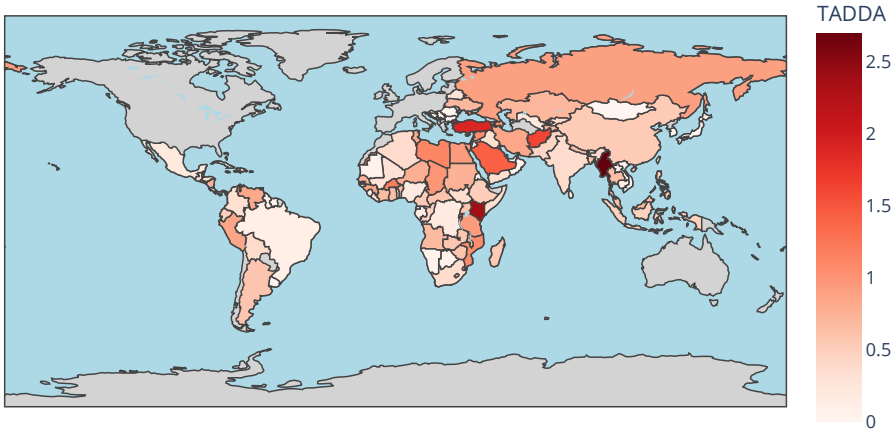

Figure 14: Country-level out-of-sample TADDA prediction errors of the best model `ARIMAX_SE_MONTHLY`. The countries marked in gray are not considered in the analysis due to insufficient data availability.

The map in Figure 15 visualizes the difference in the TADDA prediction errors for these two models. In the green colored countries, `ARIMAX_SE_MONTHLY` beats the baseline while in the red colored countries it is inferior to the no-change model.

It turns out that the `ARIMAX_SE_MONTHLY` model gives better predictions than the no-change model in 28 (17.72%) of the 158 countries studied, equally good predictions in 84 (53.16%) cases, and worse predictions in 46 cases (29.11%).

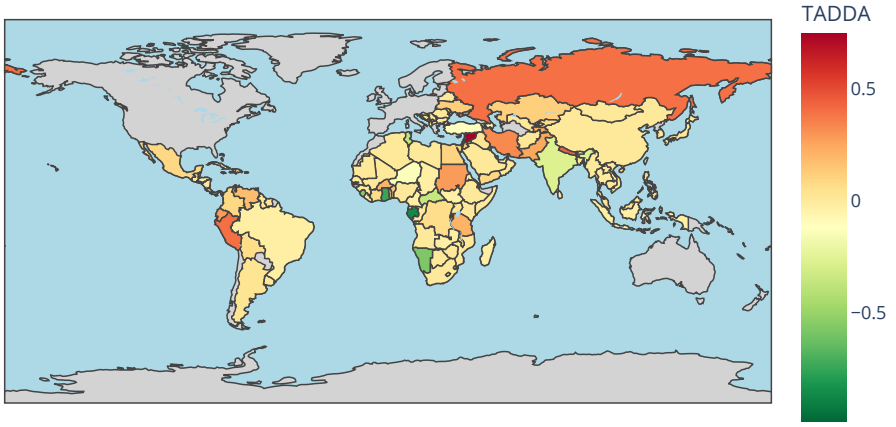

Figure 15: Country-level difference in the out-of-sample TADDA prediction errors of the `ARIMAX_SE_MONTHLY` and the `NO_CHANGE_MONTHLY` model. Positive values indicate the countries with better `ARIMAX_SE_MONTHLY` predictions, and negative values the countries with better baseline predictions.

Figure 16 lists the 10 most extreme cases for this comparison, sorted by the absolute difference in TADDA score: As the left side of the bar plot shows, the `ARIMAX_SE_MONTHLY` model is able to beat the no-change baseline by a particularly large lead of up to -0.99 TADDA units in the countries of Sierra Leone (SLE), Namibia (NAM), Ghana (GHA), Gabon (GAB), and Rwanda (RWA). In Syria (SYR), Nepal (NPL), Peru (PER), Russia (RUS), and Iran (IRN), the no-change baseline dominates, and the `ARIMAX_SE_MONTHLY` state space model performs significantly worse with an up to 0.74 points lower TADDA score.

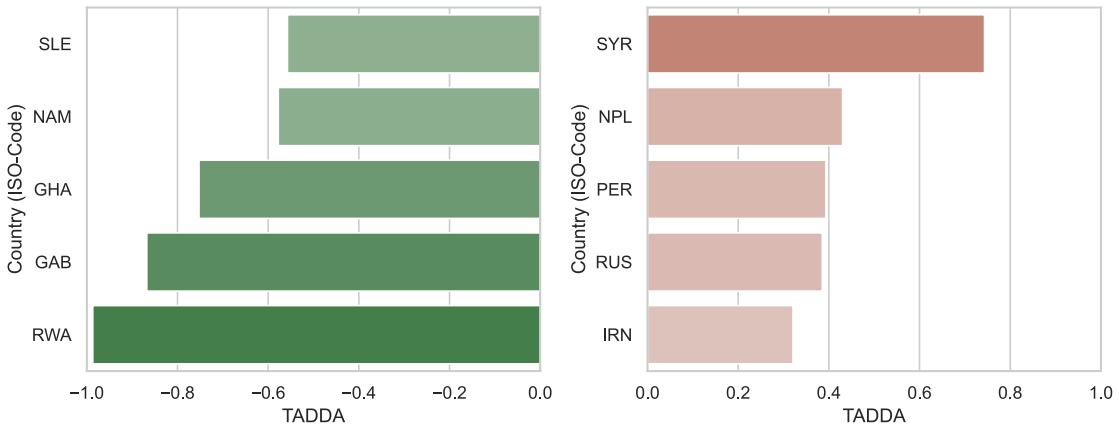

Figure 16: The 5 best and the 5 worst predicted countries of the `ARIMAX_SE_MONTHLY` compared to the baseline predictions, indicated by the difference in the out-of-sample TADDA prediction error. The green bars (left side) show the top 5 countries where the `ARIMAX_SE_MONTHLY` Model performed best in comparison to the monthly no-change model. The red bars (right side) show the top 5 countries where the no-change model `ARIMAX_SE_MONTHLY` provided the most superior forecasts.

The model performances depend on the geographical region.

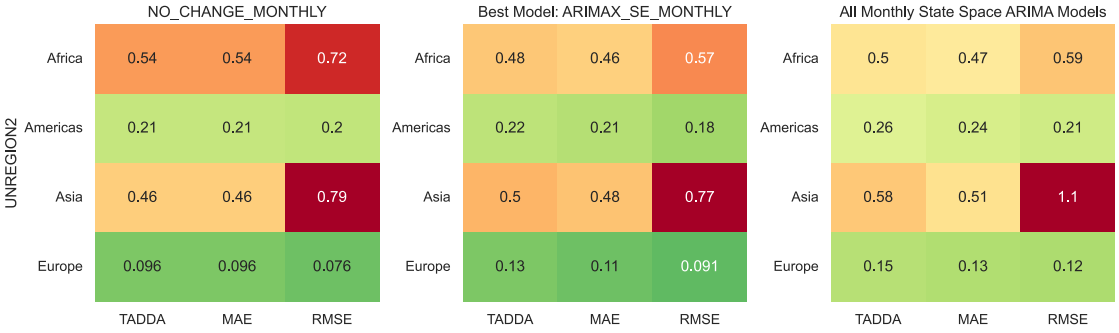

Figure 17: Average out-of-sample prediction errors by continent of the no-change baseline (left), the `ARIMAX_SE_MONTHLY` model (center) and all monthly state space ARIMA models (right).

When analyzing the model performance on a regional level, significant geographical differences in the forecast quality can be observed: Figure 17 gives a first overview of the average forecast errors in the four continents of the analyzed countries. The no-change model (left) performs better in Americas and Europe, but much worse in Africa. Similarly, the best state space ARIMA variant (middle) performs slightly worse in Europe, but much better in Africa than the baseline. The patterns in the average forecast errors of all monthly state space ARIMA models (right) resemble those of the `ARIMAX_SE_MONTHLY` model in the intercontinental comparison. African and Asian countries seem to be harder to predict in general.

Figure 18 breaks down this comparison into subcontinental regions. With the average TADDA score, it can be noticed that in Africa, especially the Northern Africa region is driving the high prediction error. The countries of the Northern Africa region are, with an extremely high TADDA score of 0.85 (baseline), 0.85 (best model) and 1.1 (all monthly state space models), as bad

predictable with the `ARIMAX_SE_MONTHLY` as with the no-change forecasts. In the remaining African regions, however, the best state space model is superior to the baseline. On the equally poorly predictable continent of Asia, Eastern Asia is an exception that is very well predicted by all models, while the other regions of the Asian continent exhibit TADDA scores far above the global average which lie at 0.44 for the baseline, 0.40 for `ARIMAX_SE_MONTHLY` and 0.52 for all monthly state space ARIMA models.

In general, the conflict fatalities in the Caribbean, South America and all regions of Europe were predicted very accurately. In Southern Europe, the average TADDA error for monthly models is well below the 0.1 mark. Conflict fatalities in Western Europe can even be perfectly predicted by all models without any errors. This is due to the fact that no conflict fatalities at all are registered for these countries in the ACLED dataset.

Due to the reduction in the number of countries for analysis from 227 to 158 presented at the beginning of the chapter, some regions are omitted entirely, as shown in the map in Figure 21 in the Appendix.

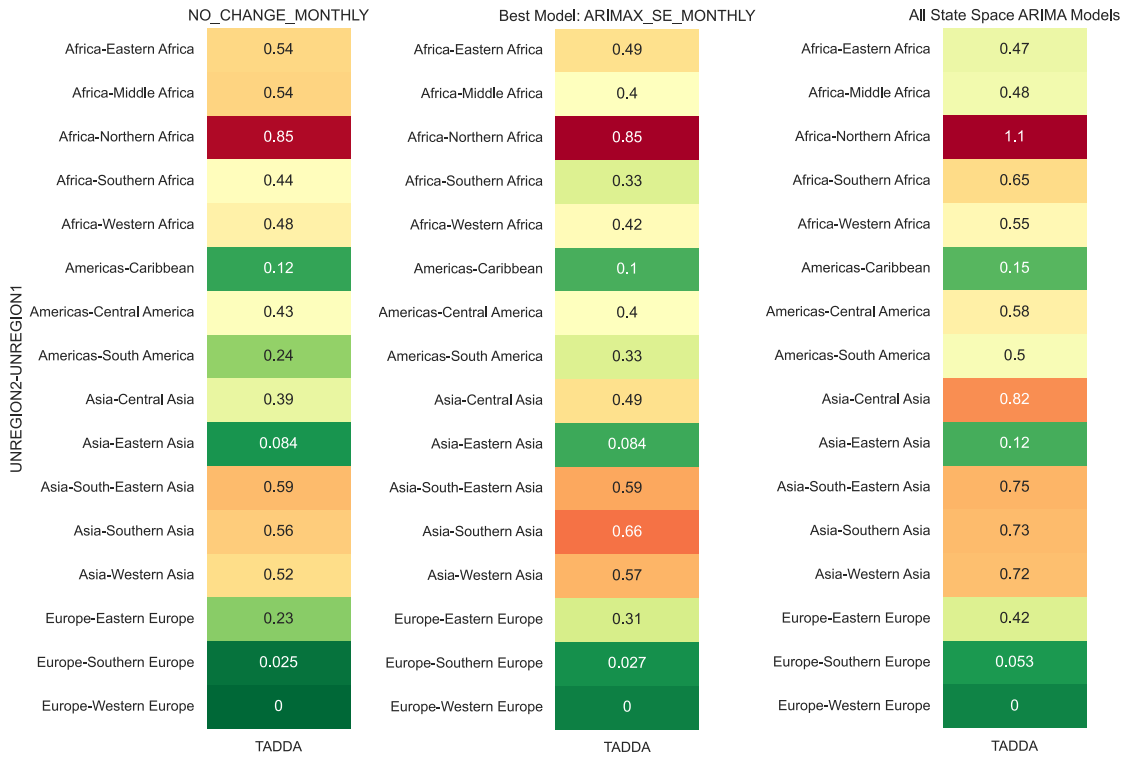

Figure 18: Average out-of-sample prediction errors of the no-change baseline (left), `ARIMAX_SE_MONTHLY` (center) and all monthly state space ARIMA models (right) aggregated by subcontinental regions.

The models struggle to make correct predictions for conflict countries.

The preceding findings suggest a relationship between conflict history and the predictability of countries, which Mueller & Rauh (2022), Collier & Sambanis (2002) have also addressed. I analyze this using the in Chapter 3.1.1 introduced categorization of countries by their conflict incidence, which I defined as the average monthly conflict fatalities per 1 million inhabitants.

As already explained at the beginning of this chapter, only for 158 of the 227 ACLED countries models all model variants could be calculated successfully. Thus, the number of countries in each

category also changes, as shown in Table 6. The number of *no-conflict* countries has more than halved, while 78.10% of the *low-conflict* countries and all the 23 *high-conflict* countries remain.

Country category	**Avg. monthly fatalities per 1 mio. inhabitants**	**Total number of countries (Share in %)**	**Countries with valid fits for all models (Remaining in %)**
No-conflict	0	67 (29.52%)	28 (41.79%)
Low-conflict	$(0, 3.28)$	137 (60.35%)	107 (78.10%)
High-conflict	$[3.28, 96.01]$	23 (10.13%)	23 (100%)
All countries	$[0, 96.01]$	227 (100%)	158 (69.60%)

Table 6: Disjoint classification of the 227 ACLED countries according to their average number of monthly conflict fatalities into the categories *no-conflict, low-conflict* and *high-conflict* countries.

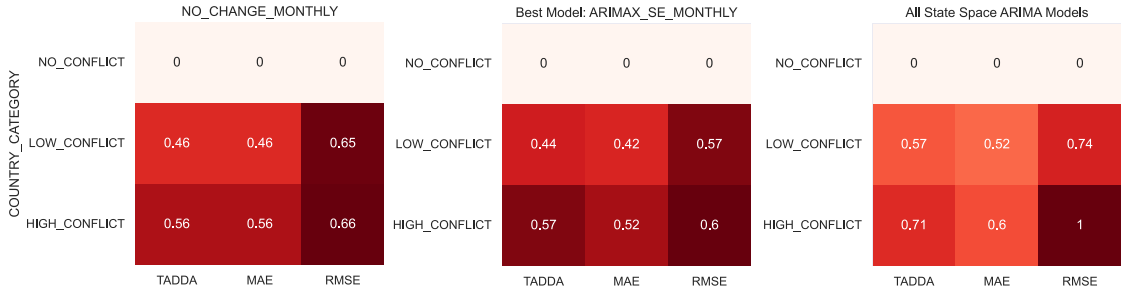

Figure 19: Average out-of-sample prediction errors of the no-change baseline (left), `ARIMAX_SE_MONTHLY` (center) and all monthly state space ARIMA models (right) aggregated by country category.

Figure 19 shows the average out-of-sample prediction errors of the no-change baseline, `ARIMAX_SE_MONTHLY` and all monthly state space ARIMA models aggregated by country category. It is immediately apparent that all models in Figure 19 provide perfect forecasts for the *no-conflict* countries with a TADDA, MAE, and RMSE score of 0. Nevertheless, it cannot be concluded from these results that the state space ARIMA methods and the no-change baseline are both able to predict conflicts in countries with peaceful conflict histories. Instead, the *no-conflict* countries represent the trivial cases where, because of the zero time series in the training dataset, the state space ARIMA models also degenerate to a 0 predicting no-change model which will never predict any conflict. The supposedly good performance for *no-conflict* countries thus only highlights the hard problem of conflict prediction formulated by Mueller & Rauh (2022) instead of solving it. When the *no-conflict* countries are dropped, the state space ARIMA models perform 21.54% worse on average. The *no-conflict* countries thus distort the prediction results to a certain extent, depending on the desired interpretation.

More interesting are the forecasts of the *low-conflict* and *high-conflict* countries. Here we see that both the no-change baseline and the state space ARIMA variants are clearly less reliable at predicting *high-conflict* countries. It is also noticeable that the `SARIMAX_SE_MONTHLY` performs slightly better in the prediction of the *low-conflict* countries in particular than the naïve no-change forecasts. From this, it can be concluded that at least in these countries, the state space ARIMA approach is able to detect certain conflict dynamics in the conflict time series or through changes in exogenous variables.

6 Conclusion

In this thesis, I have developed an automated machine learning framework capable of optimizing arbitrary econometric ARIMA state space models in a purely data-driven approach, from which country-level one-year ahead conflict forecasts can be generated.

The machine learning framework is based on a modular hierarchy of Python classes, which allows for easy extension and maintenance as well as easy integration with existing conflict prediction systems, such as the PREVIEW system at the German Federal Foreign Office. The central element is the combination of a strict model pipeline approach with an extending window cross-validation of the model hyperparameters in an exhaustive grid search that adapts to the respective data availability. The pipeline approach in the generalized `SARIMAX_LinInt_PCA()` class provides a strict separation of training and testing data, allowing imputation, standardization and dimensionality reduction of exogenous variables whilst avoiding data leakage. This machine learning framework was applied to 16 different ARIMA state space variants that model conflict based on conflict history, seasonality, spill-over effects, and socio-economic factors, as well as all combinations of these components. Moreover, each model was tested once with monthly and once with quarterly periodicity. In evaluating the performances of these model variants, I limited myself to forecasting the number of conflict fatalities, although this framework can easily be applied to any other conflict indicators, such as the number of protests. Through an out-of-sample evaluation of the model forecasts compared to a naive no-change baseline, I found that the state space ARIMA methods in this framework are highly competitive in their ability to predict conflict. In contrast to all 9 country-month models submitted to the ViEWS prediction competition, 3 of the 8 state space ARIMA monthly models outperformed the no-change baseline in Africa and even globally in all three metrics including the TADDA score (Vesco et al. 2022).

Also, in comparison with the quarterly models used at PREVIEW in the Federal Foreign Office as of July 20, 2022, the state space ARIMA models of this thesis perform extremely well. Even the best ensemble models used at PREVIEW come only close to the no-change baseline's TADDA score, which the most successful state space ARIMA models of this thesis can slightly beat. The best state space ARIMA model is superior to all PREVIEW models in the TADDA and MAE prediction errors. The performance of the Auto ARIMA model used by PREVIEW could even be nearly halved by the state space ARIMA approach from 0.818 to 0.383 in the TADDA score and 0.649 to 0.363 in the MAE. The conclusions of these comparisons are also valid if, instead of the 158 countries I used in the analysis, again all 227 ACLED countries are considered. Since mainly the perfectly predictable *no-conflict* countries were selected, the scores of all models of this thesis even improve significantly in this case. Thus, the results of this thesis in particular show that extensively tuned statistical time series models can at least keep up with other state-of-the-art conflict prediction models, if not even set a new standard.

In the evaluation of the predictions of the 16 examined state space ARIMA variants, the crucial factors responsible for the success of the state space ARIMA approach were identified. First, it could be shown that the monthly time series models provide significantly better predictions than quarterly models. Whether an even higher resolution of the input data leads to even better predictions remains an open question, and could be investigated in future work by training models on a biweekly or even weekly basis.

It has also been shown that integration of rigid annual seasonality interfered more with the conflict predictions than improved them. However, this does not mean that modeling seasonality can have no added value at all. Rather, as already suggested in Chapter 5.1, the straight forward approach to modeling seasonality used here is simply too restrictive. Before training a country model, I do not check whether there actually is any form of seasonality in the time series, nor which periodicity it has. By means of for example a preceding Fourier analysis or statistical seasonality

test, the present seasonality could be determined for each country specifically in future models. It would still remain to be shown whether a more sophisticated modeled seasonality could achieve an added value in at least the forecasts of some countries.

Furthermore, the results of the analysis have revealed that the inclusion of socio-economic country indicators and the modeling of spill-over effects through the integration of conflict indicators from neighboring countries can, in principle, contribute to an improvement of conflict predictions. One problem with ARIMA models is that, unlike machine learning methods such as Random Forest, XGBoost, or neural networks, ARIMA models become very susceptible to overfitting with just a handful of exogenous variables. I address this problem by reducing the dimensionality of the exogenous variables to a maximum of 3 principal components using PCA. An intelligent procedure for selecting the most important variables could, as an alternative or in combination with PCA, further expand the positive effect of the exogenous variables on the predictions.

Another general difficulty in integrating exogenous variables is that they are also unknown for the prediction period and therefore have to be extrapolated. I extrapolate the exogenous variables for the forecast year very simply using the last known observation from the training dataset. A better method, which for example carries trends in the exogenous variables into the forecast period, would certainly be able to integrate exogenous crisis signals into the forecasts more effectively. However, this would require not only more sophisticated extrapolation methods but also a higher resolution of the data, which, as in the case of the socio-economic country indicators of the IMF and the World Bank, are only available aggregated annually.

In Chapter 5.2 it was further demonstrated that the predictability of the countries by the state space ARIMA models is strongly linked to the geographic location of the respective country. In the country comparisons, it is important to note that the start dates of the conflict event recording in the ACLED dataset vary considerably. This introduces a certain bias into the model performance comparisons at country level, which is attributable to the different length of the training data on the one hand and the complete absence of conflict fatalities in very short time series on the other hand. Alternatively, other conflict datasets with uniform start dates and longer conflict histories such as the UCDP Georeferenced Event Dataset (GED) might be used, which in turn introduce other drawbacks in return (Raleigh & Kishi 2019).

Furthermore, by analyzing the prediction errors via categorizing the countries according to their conflict incidence and identifying Myanmar and Afghanistan as the most poorly predicted countries by the winning model, the underlying problem of conflict onset prediction has been highlighted once again. While the state space ARIMA models provided perfect but useless zero forecasts in the *no-conflict* countries due to the complete absence of conflicts, they failed in predicting countries with violent conflict histories. The rarity of the crucial conflict escalation and de-escalation events in the available datasets still poses challenges for the models to generate operationally useful forecasts. The poor prediction results of the winner model during the takeover of power by the Taliban in Afghanistan and the military coup in Myanmar are symbolic of the difficulty of the early detection of precisely those highly dynamic events that are important for effective crisis prevention.

The solution to the problem of prediction of conflict onset can thus only be partially addressed by the state space ARIMA approach of this thesis. It will consequently be up to future research to find innovative ways of defining conflict onset and to develop new modeling approaches for its prediction. This will lead to the further advancements in the field of early crisis warning.

Bibliography

Adejumo, O., Onifade, O. & Albert, S. (2021), 'Kalman filter algorithm versus other methods of estimating missing values: Time series evidence', *African Journal of Mathematics and Statistics Studies* **4**(2), 1–9.
URL: *https://doi.org/10.52589/ajmss-vfvnmqlx*

Alquist, R., Kilian, L. & Vigfusson, R. J. (2013), Chapter 8 - forecasting the price of oil, *in* G. Elliott & A. Timmermann, eds, 'Handbook of Economic Forecasting', Vol. 2 of *Handbook of Economic Forecasting*, Elsevier, pp. 427–507.
URL: *https://www.sciencedirect.com/science/article/pii/B9780444536839000086*

Collier, P., Elliott, V. L., Hegre, H., Hoeffler, A., Reynal-Querol, M. & Sambanis, N. (2003), *Breaking the Conflict Trap*, The World Bank.
URL: *https://elibrary.worldbank.org/doi/abs/10.1596/978-0-8213-5481-0*

Collier, P. & Sambanis, N. (2002), 'Understanding civil war: A new agenda', *The Journal of Conflict Resolution* **46**(1), 3–12.
URL: *http://www.jstor.org/stable/3176236*

Commandeur, J. J. F. & Koopman, S. J. (2007), *An introduction to state space time series analysis*, Practical econometrics, Oxford Univ. Press, Oxford [u.a.]. Literaturverz. S. 171 - 172.
URL: *http://www.gbv.de/dms/mpib-toc/527857335.pdf*

Durbin, J. & Koopman, S. J. (2001), *Time Series Analysis by State Space Methods*, number 9780198523543 *in* 'OUP Catalogue', Oxford University Press.
URL: *https://ideas.repec.org/b/oxp/obooks/9780198523543.html*

Fulton, C. (2015), 'Estimating time series models by state space methods in python: Statsmodels'.

Guardado, J. & Pennings, S. (2020), *The Seasonality of Conflict. Policy Research Working Paper No. 9373*, The World Bank.
URL: *https://openknowledge.worldbank.org/handle/10986/34421*

Guterres, A. (2017), 'Secretary-general's remarks to the security council open debate on "maintenance of international peace and security: Conflict prevention and sustaining peace"'. last accessed 11/10/2022.
URL: *https://www.un.org/sg/en/content/sg/statement/2017-01-10/secretary-generals-remarks-security-council-open-debate-maintenance*

Hegre, H., Karlsen, J., Nygard, H. M., Strand, H. & Urdal, H. (2013), 'Predicting Armed Conflict, 2010â20501', *International Studies Quarterly* **57**(2), 250–270.
URL: *https://doi.org/10.1111/isqu.12007*

Hegre, H., Vesco, P. & Colaresi, M. (2022), 'Lessons from an escalation prediction competition', *International Interactions* **0**(0), 1–34.
URL: *https://doi.org/10.1080/03050629.2022.2070745*

Hyndman, R. J. (n.d.), 'State space models'. last accessed 11.07.2022.
URL: *https://robjhyndman.com/talks/ABS3.pdf*

Hyndman, R. J. & Athanasopoulos, G. (2014), *Forecasting : principles and practice*, Otexts, Lexington, Ky.

James, G., Witten, D., Hastie, T. & Tibshirani, R. (2021), *Unsupervised Learning*, Springer US, New York, NY, pp. 497–552.
URL: *https://doi.org/10.1007/978-1-0716-1418-1*$_1$2

Kapoor, S. & Narayanan, A. (2022), 'Leakage and the reproducibility crisis in ml-based science'.
URL: *https://arxiv.org/abs/2207.07048*

Kleeman, L. (1996), Understanding and applying kalman filtering, *in* 'Proceedings of the Second Workshop on Perceptive Systems, Curtin University of Technology, Perth Western Australia (25-26 January 1996)'.

Mueller, H. & Rauh, C. (2022), 'The Hard Problem of Prediction for Conflict Prevention', *Journal of the European Economic Association* . jvac025.
URL: *https://doi.org/10.1093/jeea/jvac025*

Murdoch, J. C. & Sandler, T. (2002), 'Economic growth, civil wars, and spatial spillovers', *The Journal of Conflict Resolution* **46**(1), 91–110.
URL: *http://www.jstor.org/stable/3176241*

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. (2011), 'Scikit-learn: Machine learning in Python', *Journal of Machine Learning Research* **12**, 2825–2830.

Raleigh, C. & Kishi, R. (2019), 'Comparing conflict data, similarities and differences across conflict datasets'. last accessed 11/07/2022.
URL: *https://www.acleddata.com/wp-content/uploads/2019/09/ACLED-Comparison_8.2019.pdf*

Raleigh, C., Linke, A., Hegre, H. & Karlsen, J. (2010), 'Introducing acled: An armed conflict location and event dataset: Special data feature', *Journal of Peace Research* **47**(5), 651–660.
URL: *https://doi.org/10.1177/0022343310378914*

Seabold, S. & Perktold, J. (2010), statsmodels: Econometric and statistical modeling with python, *in* '9th Python in Science Conference'.

UNHCR (2022), 'Global trends report 2021'. last accessed 11/10/2022.
URL: *https://www.unhcr.org/publications/brochures/62a9d1494/global-trends-report-2021.html*

Vesco, P., Hegre, H., Colaresi, M., Jansen, R. B., Lo, A., Reisch, G. & Weidmann, N. B. (2022), 'United they stand: Findings from an escalation prediction competition', *International Interactions* **0**(0), 1–37.
URL: *https://doi.org/10.1080/03050629.2022.2029856*

Ward, M. D., Greenhill, B. D. & Bakke, K. M. (2010), 'The perils of policy by p-value: Predicting civil conflicts', *Journal of Peace Research* **47**(4), 363–375.
URL: *https://doi.org/10.1177/0022343309356491*

A Appendix

A.1 Figures

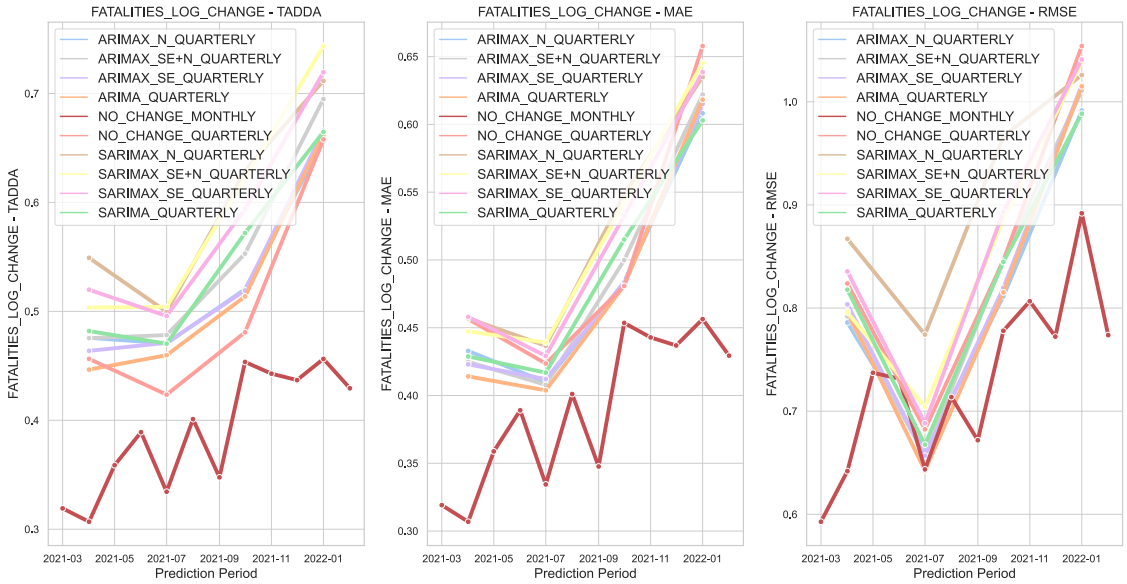

Figure 20: Prediction error of the quarterly models over the 12 prediction periods starting in March 2021. The red lines mark the monthly and quarterly no-change baseline for an comparison of the model performances.

A.2 Tables

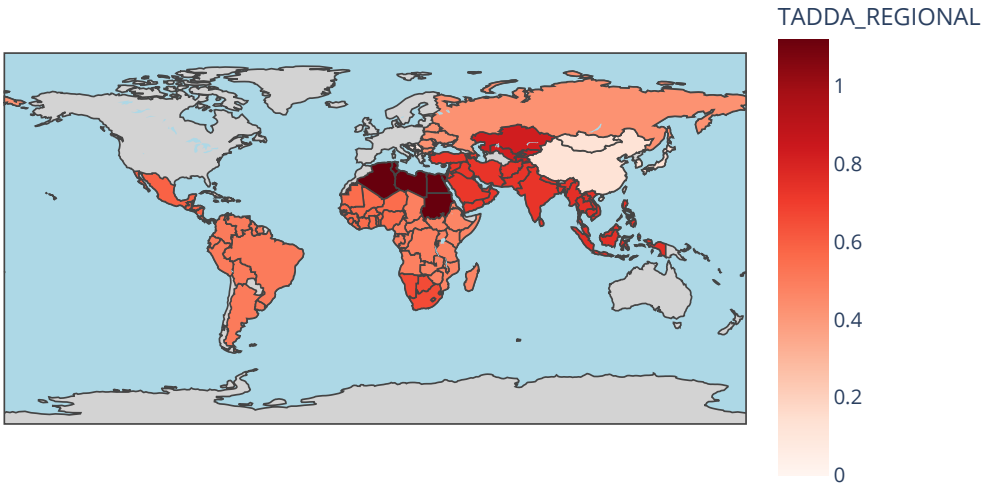

Figure 21: Average regional out-of-sample prediction errors of the best model `ARIMAX_SE_MONTHLY`. The countries marked in gray are not considered in the analysis due to insufficient data availability.

Start date	Countries (ISO-3 country code)	Count
1997-01-01	AGO, BDI, BEN, BES, BFA, BLM, BWA, CAF, CIV, CMR, COD, COG, COM, CPV, DJI, DZA, EGY, ERI, ESH, ETH, GAB, GGY, GHA, GIN, GMB, GNB, GNQ, IMN, JEY, KEN, LBR, LBY, LSO, MAR, MDG, MLI, MOZ, MRT, MUS, MWI, MYT, NAM, NER, NGA, REU, RWA, SDN, SEN, SHN, SLE, SOM, SSD, STP, SWZ, SYC, TCD, TGO, TLS, TUN, TZA, UGA, XAR, XIT, XKK, ZAF, ZMB, ZWE	67
2010-01-01	BGD, KHM, LAO, LKA, MMR, NPL, PAK, THA, VNM	9
2015-01-01	IDN, SAU, YEM	3
2016-01-01	BHR, IND, IRN, IRQ, ISR, JOR, KWT, LBN, OMN, PHL, PSE, TUR	12
2017-01-01	AFG, ARE, QAT, SYR	4
2018-01-01	ALB, ARG, ARM, ATG, AZE, BGR, BHS, BIH, BLR, BLZ, BOL, BRA, BRB, CHL, CHN, COL, CRI, CUB, CUW, CYP, DMA, DOM, ECU, GEO, GLP, GRC, GRD, GTM, GUF, GUY, HKG, HND, HRV, HTI, JAM, JPN, KAZ, KGZ, KNA, KOR, LCA, MAC, MDA, MEX, MKD, MNE, MNG, MTQ, MYS, NIC, PAN, PER, PRI, PRK, PRY, ROU, RUS, SLV, SRB, SUR, SXM, TCA, TJK, TKM, TTO, TWN, UKR, URY, UZB, VEN, XKO, XNC	72
2019-01-01	ABW, CYM	2
2020-01-01	AIA, ALA, AND, AUT, BEL, BRN, BTN, CHE, CZE, DEU, DNK, ESP, EST, FIN, FLK, FRA, FRO, GBR, GIB, GRL, HUN, IRL, ISL, ITA, LIE, LTU, LUX, LVA, MCO, MDV, MLT, MSR, NLD, NOR, POL, PRT, SGP, SJM, SMR, SVK, SVN, SWE, USA, VAT, VCT, VIR	46
2021-01-01	ASM, ATA, AUS, BMU, CAN, COK, FJI, FSM, GUM, MNP, NCL, NZL, PNG, PYF, SLB, TON, VUT, WLF, WSM	19

Table 7: Count and ISO-3 codes of countries grouped by their ACLED start date.

	Count	Mean	Std	Min	25%	50%	75%	Max
SUM(FATALITIES)/POP	227.0	2.6209	10.8942	0.0000	0.0000	0.0381	0.5011	96.0119
NUMBER_EVENTS/POP	227.0	6.7481	17.8421	0.0398	0.6797	2.5132	6.8078	219.7802
SUM(COUNT_PRO)/POP	227.0	4.3736	12.3327	0.0000	0.3137	1.7336	4.6628	146.5201
SUM(COUNT_SRZ)/POP	227.0	1.5650	6.4799	0.0000	0.0118	0.0779	0.3682	72.7027
SUM(COUNT_STRA)/POP	227.0	0.8095	5.1576	0.0000	0.0192	0.0704	0.2817	73.2601
SUM(FATALITIES_PRO)/POP	227.0	0.0410	0.1152	0.0000	0.0000	0.0028	0.0304	1.0430
SUM(FATALITIES_SRZ)/POP	227.0	2.5721	10.8523	0.0000	0.0000	0.0245	0.4468	95.7263
SUM(FATALITIES_STRA)/POP	227.0	0.0078	0.0473	0.0000	0.0000	0.0000	0.0000	0.5980

Table 8: Descriptive statistics for the 8 ACLED conflict indicators. All indicators are given as monthly averages of the 227 ACLED countries.

	Role in model	ACLED indicator	Explanation
1	Target and Predictor	SUM(FATALITIES)	Log. sum of monthly/quarterly conflict fatalities
2	Predictor	COUNT(EVENTS)	Log. number of monthly/quarterly conflicts
3	Predictor	SUM(FATALITIES_PRO)	Log. sum of monthly/quarterly conflict fatalities in protest or riot events
4	Predictor	COUNT(EVENTS_PRO)	Log. number of monthly/quarterly protest or riot events
5	Predictor	SUM(FATALITIES_SRI)	Log. sum of monthly/quarterly conflict fatalities in events with violence against civilians, explosions/remote violence or battle events
6	Predictor	COUNT(EVENTS_SRI)	Log. number of monthly/quarterly events with violence against civilians, explosions/remote violence or battle events
7	Predictor	SUM(FATALITIES_STRA)	Log. sum of monthly/quarterly conflict fatalities in events with strategic developments
8	Predictor	COUNT(EVENTS_STRA)	Log. number of monthly/quarterly events with strategic developments

Table 9: Overview of the conflict indicators generated from the ACLED data set. The ACLED indicators 1&2 were generated by summing all fatalities and counting all conflict events per country and time period (monthly/quarterly). The 6 more specific ACLED indicators 3-8 were created by additionally grouping the events by three groups of the official ACLED event categories *protests, security related incidents, strategic developments.*

WB-WDI Indicator Code	Indicator Description
CC.EST	Control of Corruption: Estimate
GE.EST	Government Effectiveness: Estimate
PV.EST	Political Stability and Absence of Violence/Terrorism: Estimate
RL.EST	Rule of Law: Estimate
RQ.EST	Regulatory Quality: Estimate
VA.EST	Voice and Accountability: Estimate
SI.POV.NAHC	Poverty headcount ratio at national poverty lines (% of population)
SI.POV.DDAY	Poverty headcount ratio at $2.15 a day (2017 PPP) (% of population)
SI.SPR.PC40.ZG	Annualized average growth rate in per capita real survey mean consumption or income, bottom 40% of population (%)
SI.POV.GINI	Gini index
SP.POP.GROW	Population growth (annual %)
SE.PRM.ENRR	School enrollment, primary (% gross)
SL.TLF.CACT.ZS	Labor force participation rate, total (% of total population ages 15+) (modeled ILO estimate)
SL.AGR.EMPL.ZS	Employment in agriculture (% of total employment) (modeled ILO estimate)
SL.UEM.TOTL.ZS	Unemployment, total (% of total labor force) (modeled ILO estimate)
SL.UEM.1524.MA.ZS	Unemployment, youth male (% of male labor force ages 15-24) (modeled ILO estimate)
SH.STA.MMRT	Maternal mortality ratio (modeled estimate, per 100,000 live births)
SH.DYN.MORT	Mortality rate, under-5 (per 1,000 live births)
SL.TLF.TOTL.FE.ZS	Labor force, female (% of total labor force)
SE.ENR.PRSC.FM.ZS	School enrollment, primary and secondary (gross), gender parity index (GPI)
AG.LND.AGRI.ZS	Agricultural land (% of land area)
AG.LND.PRCP.MM	Average precipitation in depth (mm per year)
EG.ELC.ACCS.ZS	Access to electricity (% of population)
NY.GDP.TOTL.RT.ZS	Total natural resources rents (% of GDP)
SH.STA.BASS.UR.ZS	People using at least basic sanitation services, urban (% of urban population)
SH.STA.BASS.RU.ZS	People using at least basic sanitation services, rural (% of rural population)
SH.H2O.SMDW.ZS	People using safely managed drinking water services (% of population)
NY.GDP.MKTP.KD.ZG	GDP growth (annual %)
NE.EXP.GNFS.ZS	Exports of goods and services (% of GDP)
NE.IMP.GNFS.ZS	Imports of goods and services (% of GDP)
NY.GNP.PCAP.CD	GNI per capita, Atlas method (current US$)
BX.TRF.PWKR.DT.GD.ZS	Personal remittances, received (% of GDP)
IC.TAX.GIFT.ZS	Firms expected to give gifts in meetings with tax officials (% of firms)
IC.TAX.TOTL.CP.ZS	Total tax and contribution rate (% of profit)
MS.MIL.XPND.GD.ZS	Military expenditure (% of GDP)
MS.MIL.TOTL.TF.ZS	Armed forces personnel (% of total labor force)
VC.IDP.NWDS	Internally displaced persons, new displacement associated with disasters (number of cases)
VC.IDP.NWCV	Internally displaced persons, new displacement associated with conflict and violence (number of cases)
VC.IHR.PSRC.P5	Intentional homicides (per 100,000 people)
IT.NET.USER.ZS	Individuals using the Internet (% of population)
GB.XPD.RSDV.GD.ZS	Research and development expenditure (% of GDP)
TG.VAL.TOTL.GD.ZS	Merchandise trade (% of GDP)
SM.POP.REFG	Refugee population by country or territory of asylum
SM.POP.REFG.OR	Refugee population by country or territory of origin
ST.INT.RCPT.XP.ZS	International tourism, receipts (% of total exports)
NE.RSB.GNFS.ZS	External balance on goods and services (% of GDP)
FP.CPI.TOTL.ZG	Inflation, consumer prices (annual %)

Table 10: The selected 47 of the originally 56 socio-economic WB indicators that are used together with the IMF indicators as exogenous predictor variables in the `(S)ARIMAX_SE` and `(S)ARIMAX_SE+N` models.

Data and Code Provision

https://github.com/adrian-io/sarimax-conflict-forecast/