Appendix for *What Drives State-Sponsored Violence?: Evidence from Extreme Bounds Analysis and Ensemble Learning Models*[*]

Danilo Freire Gary Uzonyi

11 June 2018

Contents

[*]Freire: PhD candidate, Department of Political Economy, King's College London. Email address: danilo-freire@gmail.com. Uzonyi: Assistant Professor, Department of Political Science; Research Fellow, Howard H. Baker Jr. Center for Public Policy, University of Tennessee. Email address: guzonyi@utk.edu. All replication materials are available at https://github.com/danilofreire/mass-killings.

1 Introduction

This appendix contains all required information to replicate the numerical analyses presented in "What Drives State-Sponsored Violence?: Evidence from Extreme Bounds Analysis and Ensemble Learning Models." R code can be found in subsection 1.7 and the data are available on the following GitHub repository: https://github.com/danilofreire/mass-killings. We used R version 3.4.4 (15-03-2018) and Ubuntu 16.04.4 LTS to perform all statistical calculations.

1.1 Variable Selection

We employ some criteria to select our explanatory variables. First, we included only published articles in our sample. Although working papers and policy may also provide important insights about the onset of mass killings, we believe that peer-reviewed research is probably better suited for our purposes. Also, we included only papers that use regression methods on a global sample and were published from 1995 to 2015. Our final sample comprises 45 articles: Anderton and Carter (2015), Balcells (2010, 2011), Besançon (2005), Bulutgil (2015), Bundervoet (2009), Clayton and Thomson (2016), Colaresi and Carey (2008), Downes (2006, 2007), Easterly et al. (2006), Eck and Hultman (2007), Esteban et al. (2015), Fazal and Greene (2015), Fjelde and Hultman (2014), Goldsmith et al. (2013), Harff (2003), Joshi and Quinn (2017), Kim (2010), Kim (2016), Kisangani and Wayne Nafziger (2007), Koren (2017), Krain (1997), Manekin (2013), McDoom (2013, 2014), Melander et al. (2009), Montalvo and Reynal-Querol (2008), Pilster et al. (2016), Querido (2009), Raleigh (2012), Rost (2013), Rummel (1995), Schneider and Bussmann (2013), Siroky and Dzutsati (2015), Stanton (2015), Sullivan (2012), Tir and Jasinski (2008), Ulfelder and Valentino (2008), Ulfelder (2012), Uzonyi (2015, 2016) Valentino et al. (2004), Valentino et al. (2006), Verpoorten (2012), Wayman and Tago (2010), Wig and Tollefsen (2016), and Yanagizawa-Drott (2014).

We find that in those 45 studies scholars made use of nearly 180 measurements to capture roughly 30 key concepts related to threat and costs of mass killings. To be added to our models, a variable should appear in at least two articles. The covariates are summarised in table 1. A complete list of variables is available at https://github.com/danilofreire/mass-killings/data.

Table 1: Independent Variables

Variable	Coded	Source
Assassination	Dichotomous	Banks (1999)
CINC	Continuous	Singer et al. (1972)
Coup d'état	Dichotomous	Marshall et al. (2017)
COW civil war onset	Dichotomous	Singer et al. (1972); Singer (1988)
COW civil war ongoing	Dichotomous	Singer et al. (1972); Singer (1988)
Democracy (Polity IV ≥ 6)	Dichotomous	Authors' own calculations
Discriminated dummy	Dichotomous	Cederman et al. (2010)
Discriminated population	Continuous	Cederman et al. (2010)
Ethnic diversity (ELF)	Continuous	Fearon and Laitin (2003)
Ethnic war start	Dichotomous	Cederman et al. (2010)
Ethnic war ongoing	Dichotomous	Cederman et al. (2010)
Excluded population	Continuous	Cederman et al. (2010)
Interstate war	Dichotomous	Singer (1988); Singer et al. (1972)
Guerrilla	Dichotomous	Balcells and Kalyvas (2014)
Military expenditure	Continuous	Singer et al. (1972)
Military personnel	Continuous	Singer et al. (1972)
Militias	Dichotomous	Carey et al. (2013)
Mountainous Terrain	Continuous	Fearon and Laitin (2003)
Physical integrity	Continuous	Cingranelli and Richards (2010)
Polarisation (all groups/main group)	Continuous	Authors' own calculations
Polarisation (all groups/population)	Continuous	Authors' own calculations
Polarisation (included groups/population)	Continuous	Authors' own calculations
Polarisation (included groups/main group)	Continuous	Authors' own calculations
Polity IV	Continuous	Marshall et al. (2017)
Polity IV squared	Continuous	Authors' own calculations
Population	Continuous	Gleditsch (2002)
Post-Cold War	Dichotomous	Authors' own calculations
Real GDP	Continuous	Gleditsch (2002)
Real GDP per capita	Continuous	Gleditsch (2002)
Real GDP per capita (log)	Continuous	Authors' own calculations
Regime transition	Continuous	Authors' own calculations
Riot	Dichotomous	Banks (1999)
Total battle deaths	Continuous	Lacina and Gleditsch (2005)
Total trade	Continuous	Singer et al. (1972)
Trade dependence (total trade/real GDP)	Continuous	Authors' own calculations
UCDP civil war onset	Dichotomous	Allansson et al. (2017); Gleditsch et al. (2002)
UCDP civil war ongoing	Dichotomous	Allansson et al. (2017); Gleditsch et al. (2002)
Urban population (percentage)	Continuous	Singer et al. (1972)
Years since last mass killing	Continuous	Authors' own calculations
War with territory aims	Dichotomous	Allansson et al. (2017); Gleditsch et al. (2002)

1.2 Descriptive Statistics

Table 2: Descriptive Statistics

Statistic	N	Mean	St. Dev.	Min	Max
Country code	9,162	452.84	247.74	2	950
Year	9,162	1,983.56	18.77	1,945	2,013
Genocide/politicide onset	8,933	0.005	0.07	0	1
Mass killing onset	9,162	0.01	0.11	0	1
Independent Variables					
Assassination dummy	8,991	0.08	0.27	0	1
CINC	8,767	0.01	0.02	0.00	0.38
Coup dummy	8,587	0.05	0.21	0	1
COW civil war onset	8,160	0.01	0.12	0	1
COW civil war ongoing	8,160	0.07	0.25	0	1
Democracy dummy	8,991	0.37	0.48	0	1
Discriminated dummy	6,981	0.35	0.48	0	1
Discriminated population	6,981	0.06	0.15	0.00	0.98
Ethnic diversity (ELF)	6,981	0.41	0.31	0	1
Ethnic war start	7,760	0.01	0.12	0	1
Ethnic war ongoing	7,760	0.11	0.31	0	1
Excluded population	6,981	0.16	0.22	0.00	0.98
Interstate war	8,159	0.04	0.19	0	1
Guerrilla dummy	714	0.81	0.40	0	1
Military expenditure	8,290	4,607,120	27,785,906	0	693,600,000
Military personnel	8,620	176.70	520.90	0	12,500
Militias	4,097	0.22	0.42	0	1
Mountainous Terrain	7,358	2.14	1.43	0.00	4.56
Physical integrity	4,499	4.73	2.31	0	8
Polarisation (all groups/main group)	6,981	0.70	0.26	0.05	1
Polarisation (all groups/population)	6,981	0.63	0.32	0	1
Polarisation (included groups/population)	5,610	0.64	0.32	0	1
Polarisation (included groups/main group)	6,981	0.23	0.35	0	1
Polity IV	8,558	0.42	7.50	−10	10
Polity IV squared	8,558	56.35	32.59	0	100
Population	8,293	32,993.61	112,886.40	118.21	1,324,353.00
Post-Cold War	8,991	0.40	0.49	0	1
Real GDP	8,293	215,317.70	804,827.20	129.68	13,193,478.00
Real GDP per capita	8,293	8,104.20	18,376.73	132.82	632,239.50
Real GDP per capita (log)	8,293	8.25	1.20	4.89	13.36
Regime transition	1,221	−4.24	41.50	−77	99
Riot dummy	8,991	0.16	0.36	0	1
Total battle deaths	714	6,050.86	24,404.78	100	350,000
Total trade	8,174	53,804.01	222,209.90	0.80	4,825,363.00
Trade dependence	7,670	0.26	0.69	0.0001	22.11
UCDP civil war onset	8,733	0.02	0.14	0	1
UCDP civil war ongoing	8,733	0.15	0.36	0	1
Urban population (percentage)	8,767	0.22	0.17	0.00	1.51
Years since last mass killing	9,162	23.81	17.71	0	68
War with territory aims	8,924	0.07	0.26	0	1

Note: All independent variables were lagged one year.

1.3 Extreme Bounds Analysis Extensions

1.3.1 Main Model

We present a series of histograms with the coefficients' distribution of all variables in the main EBA model. There are 36 variables in total, seven of which are robust: Log GDP per capita, post-Cold War period, onset and ongoing civil wars (measured by the UCDP), previous riots, ethnic diversity and the squared term of the Polity IV index.

Variable	**Avg. β**	**Avg. SE**	**% Sig.**	**CDF(0)**	**Models**
Base variables					
Log GDP per capita	-0.0091	0.0052	76.055	0.9335	226707
Additional variables					
Post-Cold War years	-0.0133	0.0085	72.845	0.9472	35614
UCDP civil war onset	0.0529	0.0321	52.378	0.9441	20854
Previous riots	0.0140	0.0100	56.242	0.9216	35614
UCDP ongoing civil war	0.0172	0.0115	65.652	0.9092	20854
Ethnic diversity (ELF)	0.0184	0.0137	56.674	0.9050	35614
Polity IV squared	-0.0002	0.0001	61.206	0.9031	35614

Table 3: Extreme Bounds Analysis – Mass killings

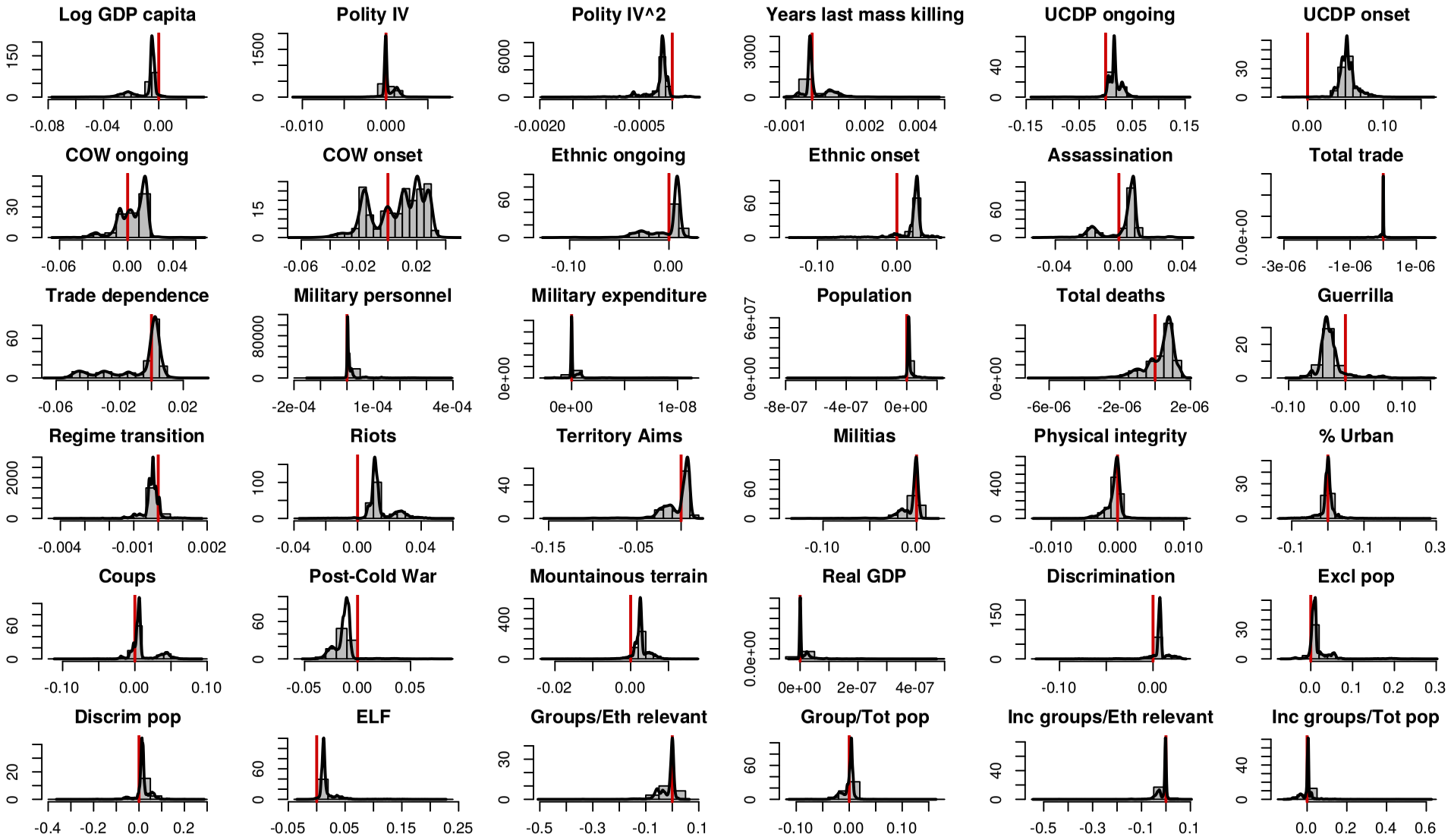

Figure 1: Extreme Bounds Analysis – Mass Killings

1.3.2 Genocides during Civil Wars

Next, we discuss genocides that occur during wartime. We use three covariates that denote ongoing civil conflicts: one by the Uppsala Conflict Data Program (Allansson et al. 2017; Gleditsch et al. 2002), another by the Correlates of War (Sarkees and Wayman 2010), and a third indicating the onset of ethnic conflict as coded by Cederman et al. (2010). The variables that reach significance in this set of models below are notably different from those obtained in the main estimation. This result provides evidence that mass violence during wartime time follows a separate logic from state killings in peacetime.

Variable	**Avg. β**	**Avg. SE**	**% Sig.**	**CDF(0)**	**Models**
UCDP data					
Territory aims	-0.044	0.019	74.997	0.9804	17902
Post-Cold War years	-0.038	0.019	66.574	0.9222	17902
COW data					
Physical integrity	0.024	0.013	66.674	0.9564	17902
Militias	-0.099	0.048	73.104	0.9490	17902
Years since last mass killing	0.006	0.002	88.208	0.9472	101583
Previous riots	0.078	0.041	65.412	0.9348	17902
Ethnic diversity (ELF)	0.095	0.062	48.615	0.9000	17902
Cederman et al. data					
Territory aims	-0.051	0.026	74.288	0.9167	17902
Militias	-0.050	0.035	52.240	0.9101	17902

Table 4: EBA – Mass Killings during Civil Wars

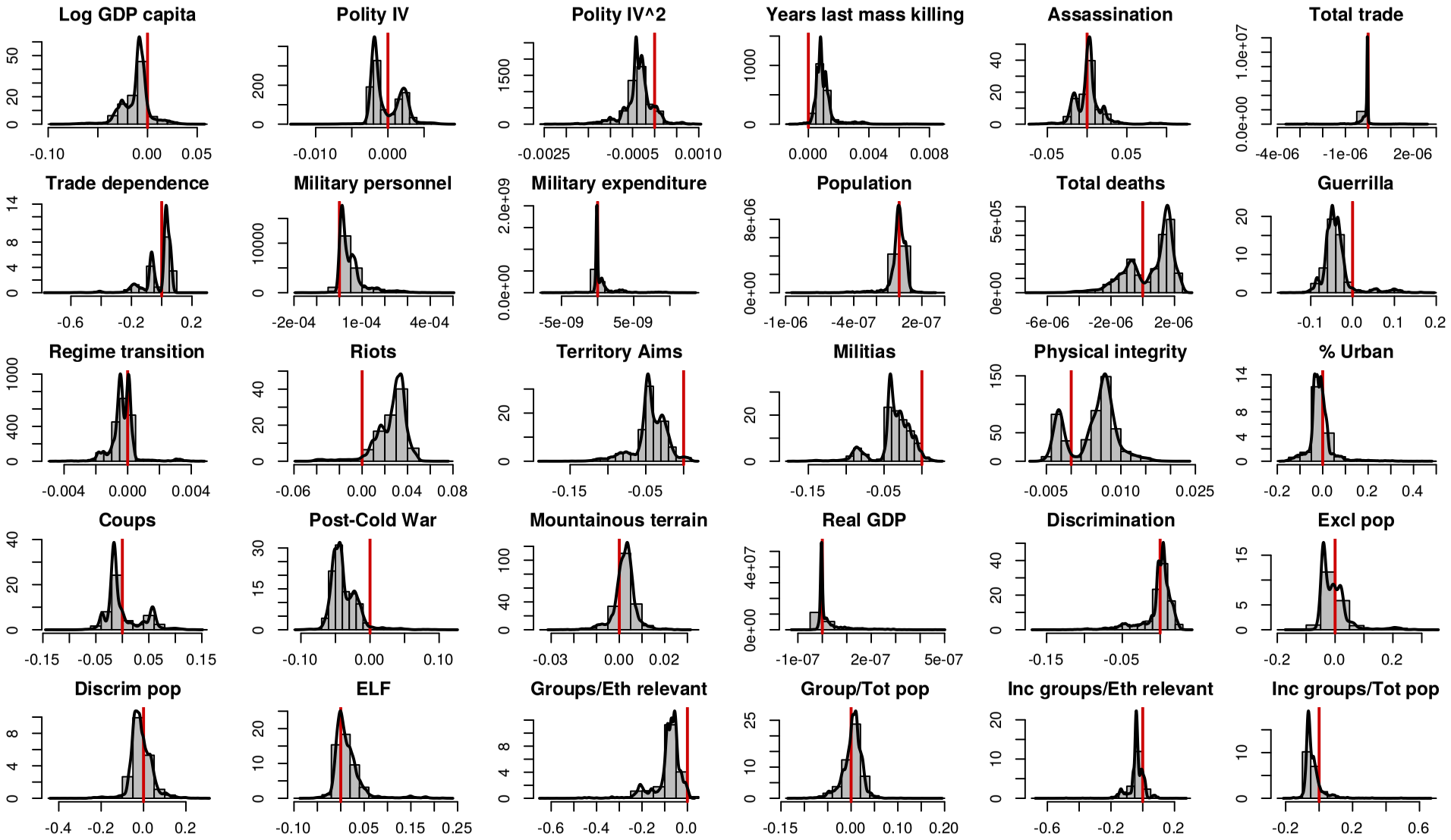

Figure 2: EBA – Mass Killings during Civil Wars (UCDP Data)

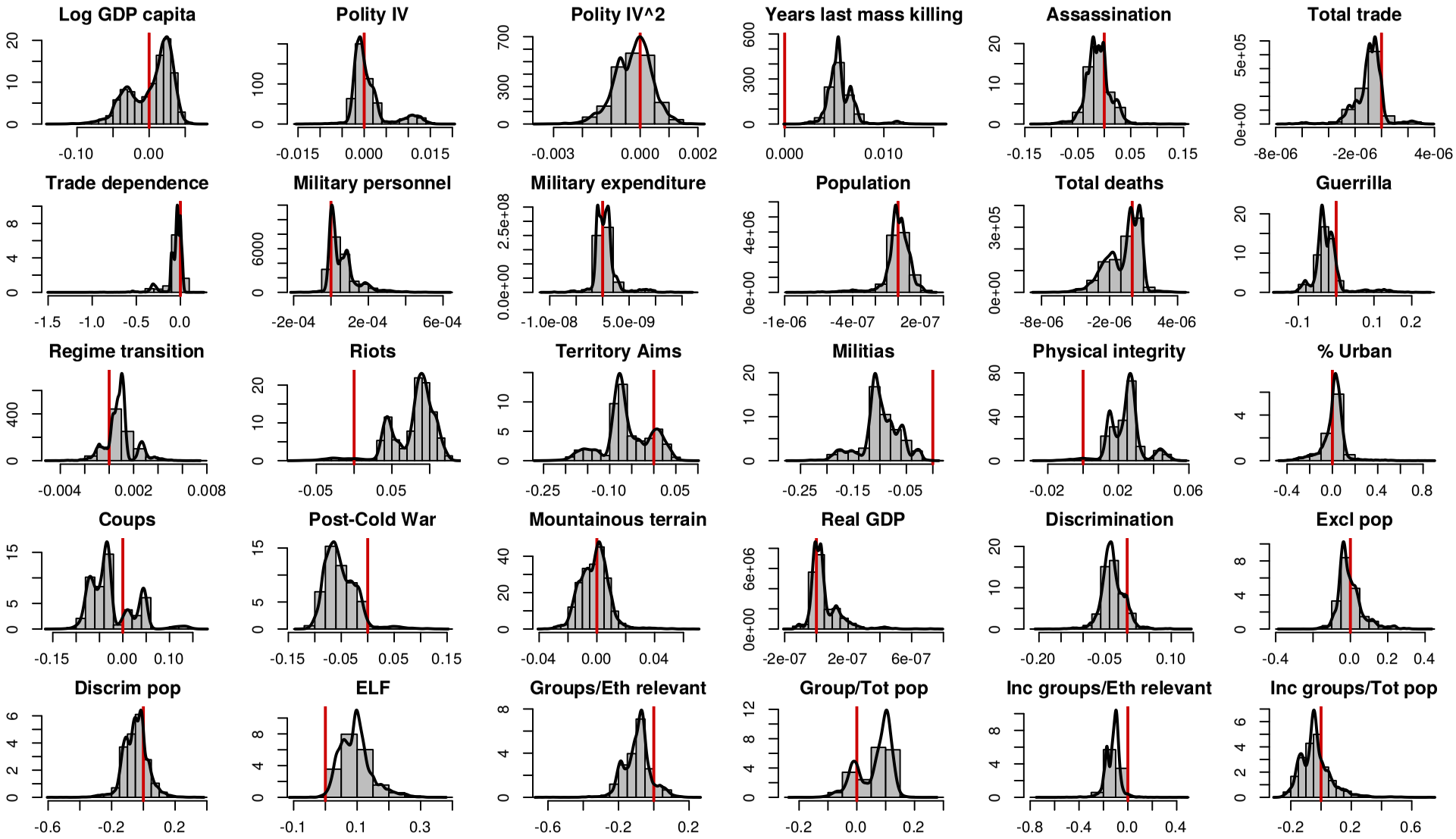

Figure 3: EBA – Mass Killings during Civil Wars (COW Data)

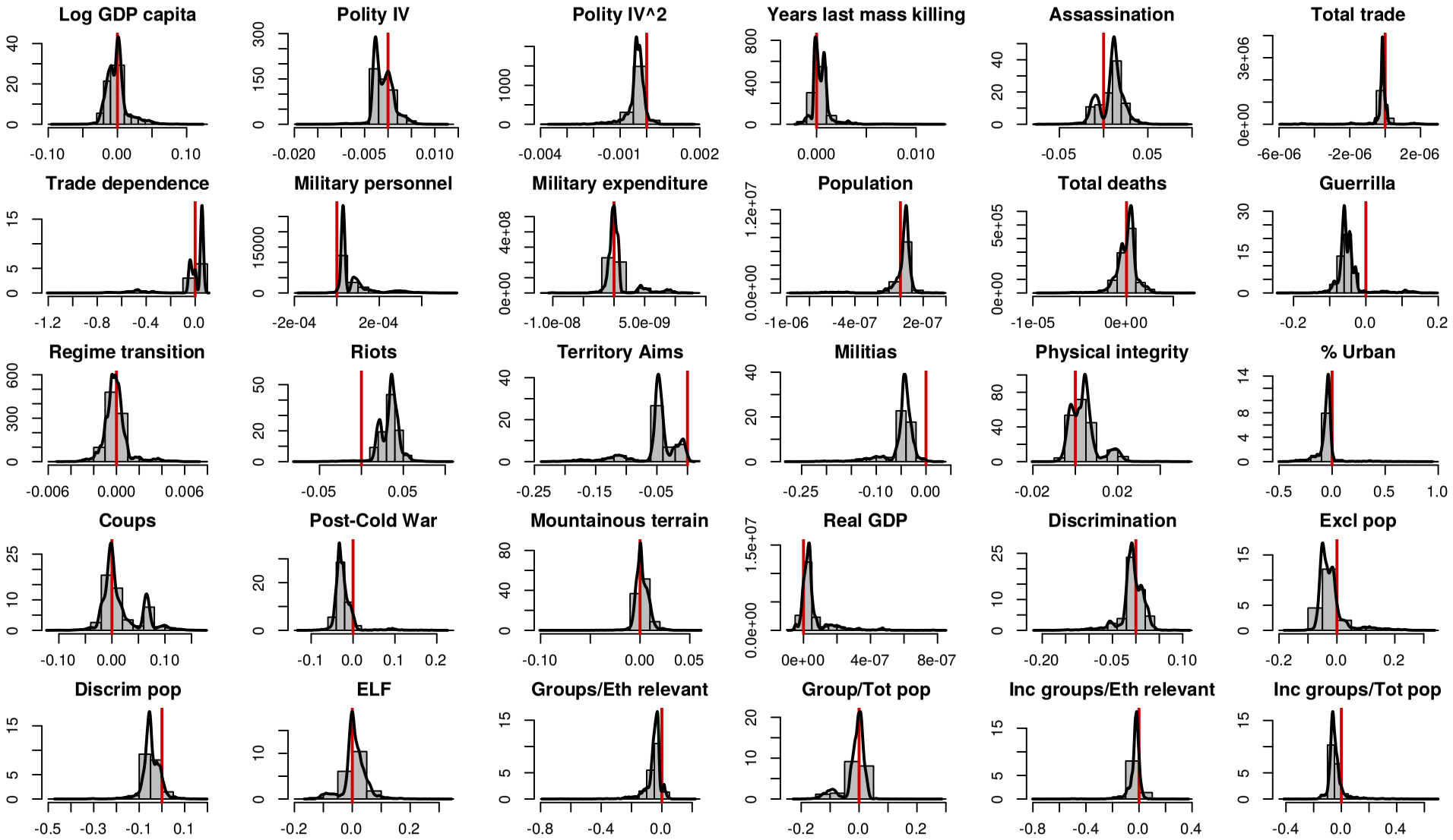

Figure 4: EBA – Mass Killings during ethnic civil wars (Cederman et al. Data)

1.3.3 Alternative Number of Variables

The models below are based on 50,000 random draws from the full set of all possible regression models. Sala-i-Martin et al. (2004, 819) argue that random sampling produces unbiased estimates of the regression coefficients with low computational time. The models presented in the article, however, include the full set of possible regressions.

The following table shows the results of an EBA with 3 variable combinations per model. The results are very similar to those reported above.

Variable	**Avg. β**	**Avg. SE**	**% Sig.**	**CDF(0)**	**Models**
Base variables					
Log GDP per capita	0.0082	0.0043	81.439	0.9504	40677
Additional variables					
Post-Cold War years	-0.0121	0.0069	77.804	0.9609	5064
UCDP civil war onset	0.0523	0.0292	62.561	0.9574	3304
Previous riots	0.0134	0.0084	65.936	0.9401	5064
UCDP ongoing civil war	0.0177	0.0094	72.367	0.9372	3304
Polity IV squared	-0.0002	0.0001	66.035	0.9268	5064
Ethnic diversity (ELF)	0.0162	0.0110	70.794	0.9266	5064

Table 5: EBA – 3 Variables

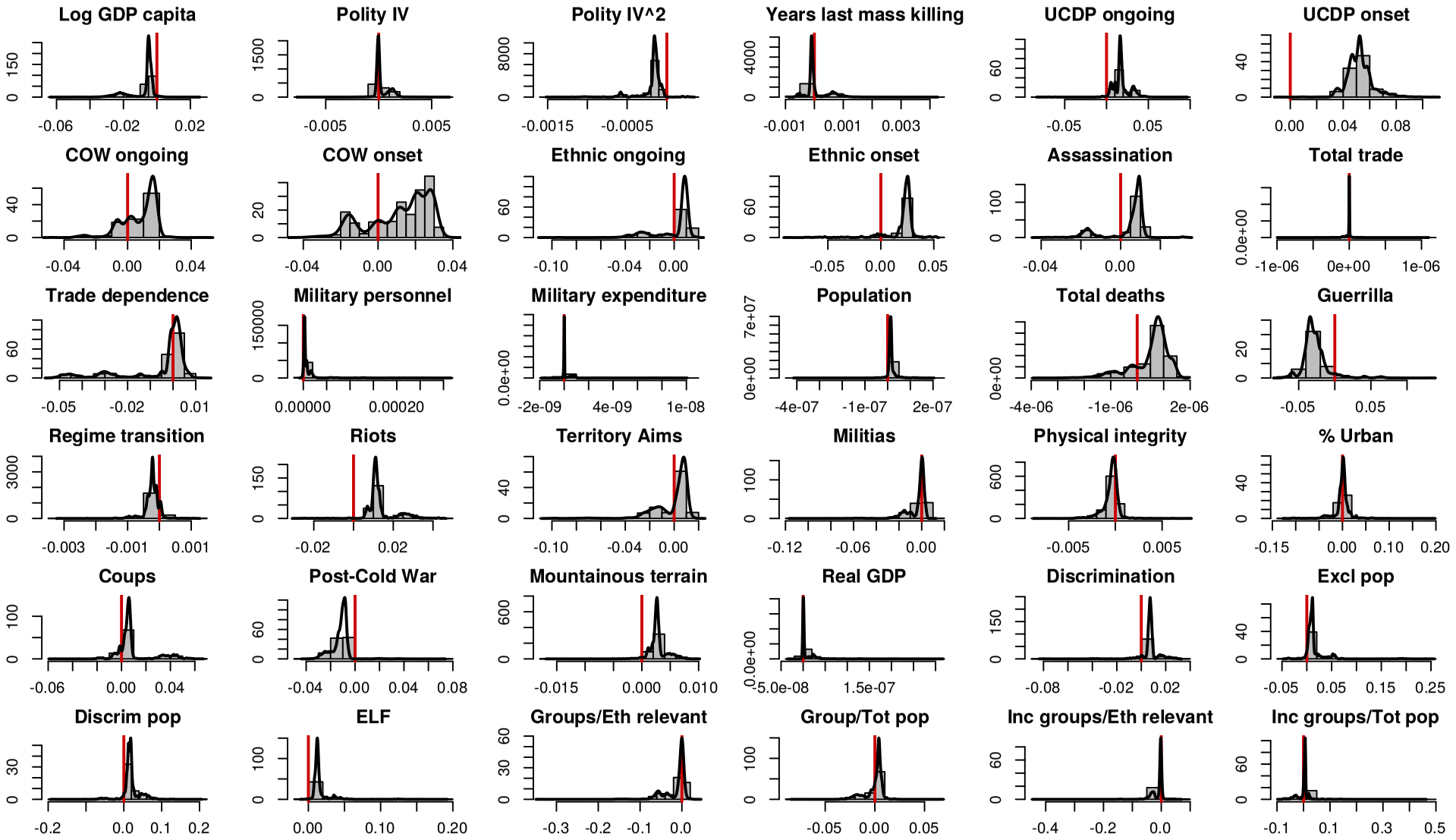

Figure 5: EBA – 3 Variables

Table 6 presents the results for models with up 5 variables in each regressions. In contrast with our main EBA model, the indicators of UCDP ongoing civil wars, ethnic diversity, and Polity IV square drop out of significance. Their individual CDFs(0) are about 0.88, just marginally below our specified threshold of 0.9.

Variable	**Avg. β**	**Avg. SE**	**% Sig.**	**CDF(0)**	**Models**
Base variables					
Log GDP per capita	-0.010	0.006	70.806	0.9161	50000
Additional variables					
Post-Cold War years	-0.014	0.010	68.496	0.9336	9532
UCDP civil war onset	0.053	0.035	44.784	0.9308	5100
Previous riots	0.015	0.012	47.988	0.9047	9569

Table 6: EBA – 5 Variables

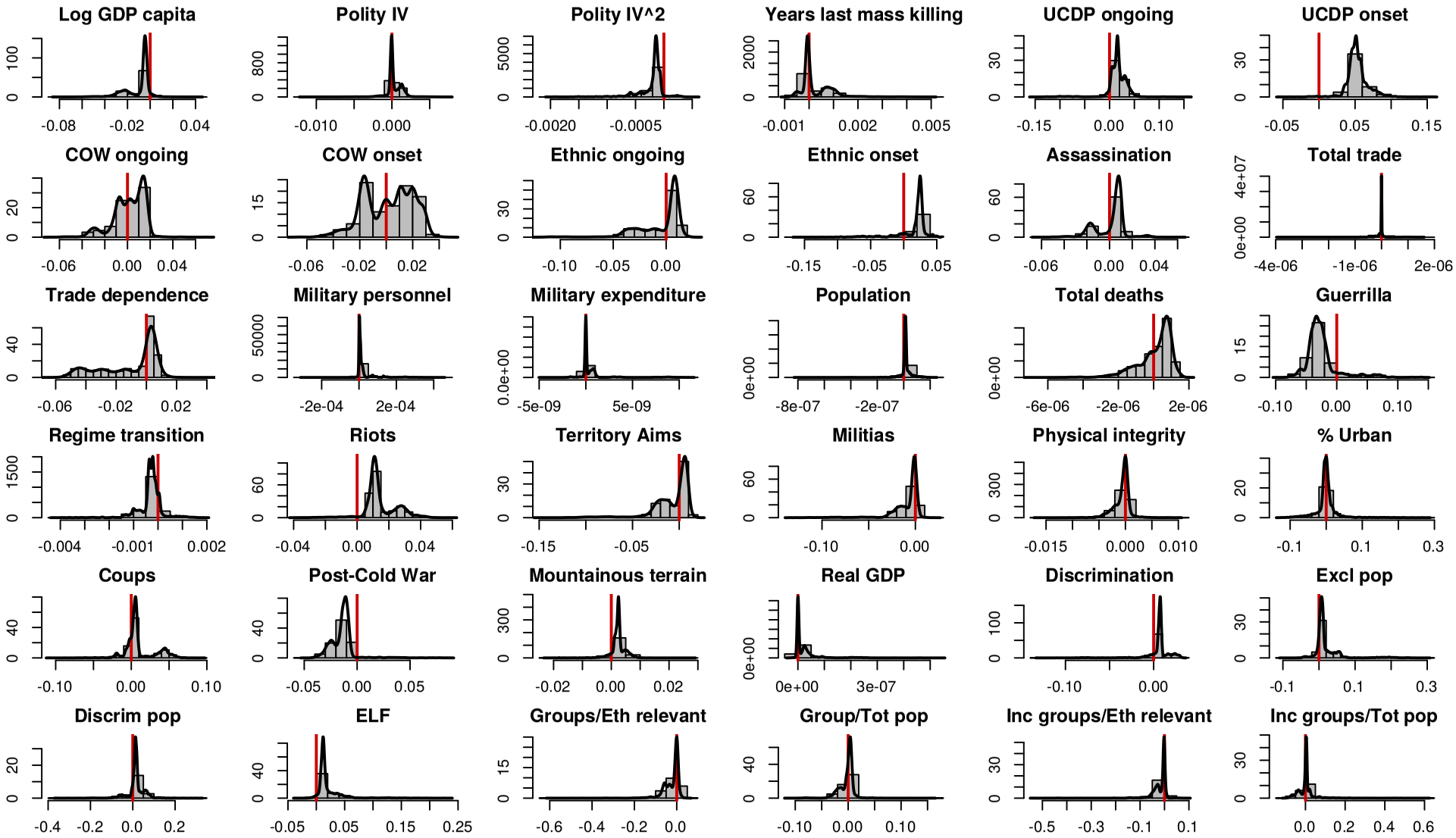

Figure 6: EBA – 5 Variables

1.3.4 Alternative Variance Inflation Factors

In this subsection, we estimate EBA models with different values of Variance Inflation Factor (VIF), which is a measure of multicollinearity. There is no standard definition about what constitutes an acceptable VIF value, although researchers often use 10 as rule of thumb to indicate strong multicollinearity (O'Brien 2007, 674). Our original model used a slightly more conservative value of 7 as a cutoff. Here, we test the same model with VIF = 10 (less strict), 2.5 (more conservative), and a model without VIF restrictions. The results are essentially identical to those of the main model. In the model with no VIF restriction, however, ethnic fractionalisation fails to meet the threshold by a very small margin. The CDF(0) of that covariate is 0.897, very close to the required value of 0.9.

Variable	**Avg. β**	**Avg. SE**	**% Sig.**	**CDF(0)**	**Models**
Base variables					
Log GDP per capita	-0.0091	0.0052	76.354	0.9343	50000
Additional variables					
Post-Cold War years	-0.0134	0.0084	73.540	0.9495	7929
UCDP civil war onset	0.0529	0.0322	52.141	0.9438	4553
Previous riots	0.0140	0.0100	56.433	0.9216	7772
UCDP ongoing civil war	0.0172	0.0113	66.013	0.9113	4587
Ethnic diversity (ELF)	0.0182	0.0136	56.872	0.9056	8076
Polity IV squared	-0.0002	0.0001	60.791	0.9021	7835

Table 7: EBA – VIF 10

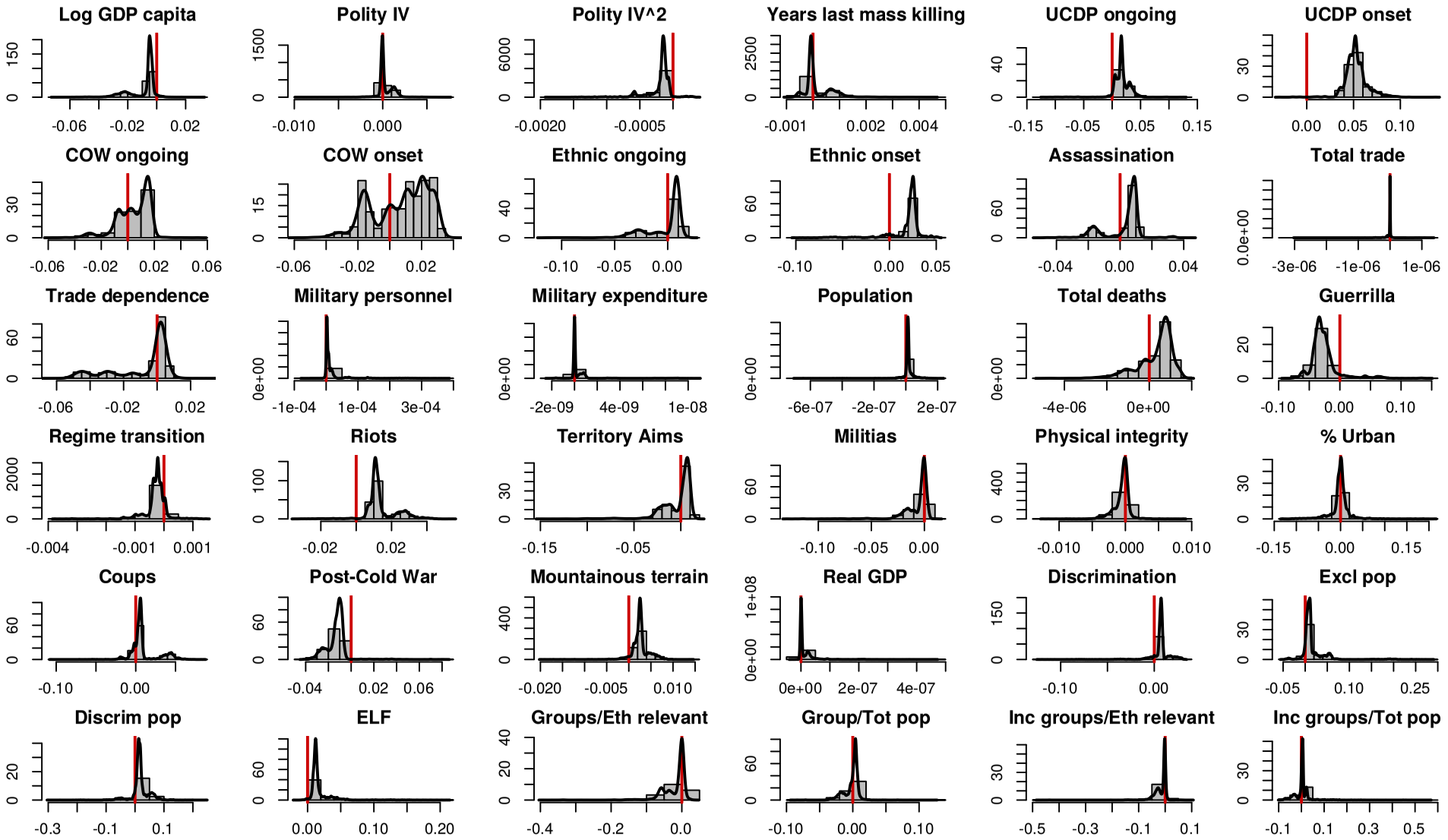

Figure 7: EBA – VIF 10

Variable	Avg. β	Avg. SE	% Sig.	CDF(0)	Models
Base variables					
Log GDP per capita	-0.0090	0.0051	76.055	0.9343	49620
Additional variables					
Post-Cold War years	-0.0132	0.0084	72.845	0.9490	7929
UCDP civil war onset	0.0529	0.0322	52.378	0.9438	4553
Previous riots	0.0141	0.0101	56.242	0.9199	7772
UCDP ongoing civil war	0.0174	0.0114	65.652	0.9103	4587
Ethnic diversity (ELF)	0.0184	0.0137	56.674	0.9054	8076
Polity IV squared	-0.0002	0.0001	61.206	0.90267	7835

Table 8: EBA – VIF 2.5

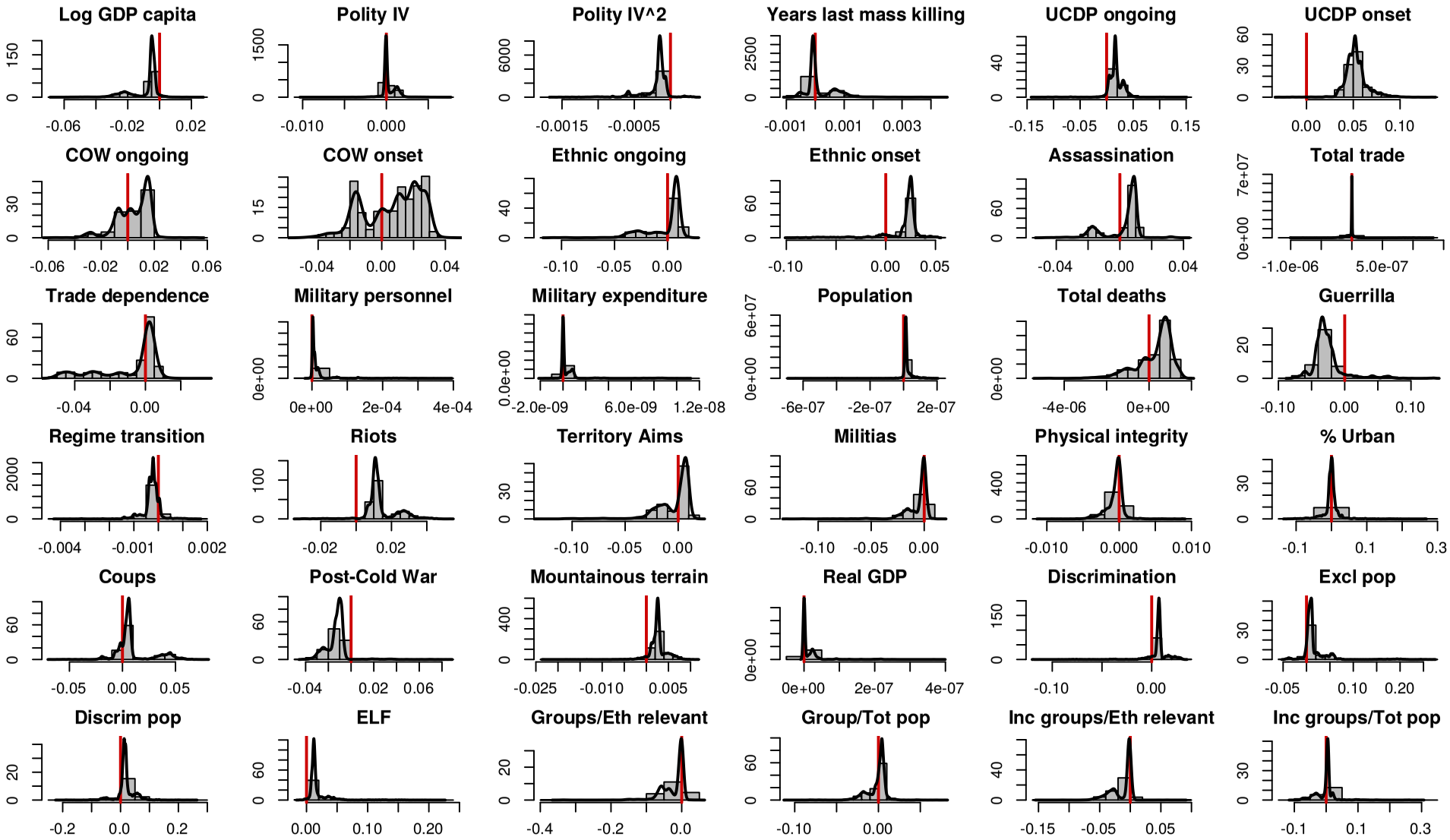

Figure 8: EBA – VIF 2.5

Variable	Avg. β	Avg. SE	% Sig.	CDF(0)	Models
Base variables					
Log GDP per capita	-0.0091	0.0052	75.940	0.9343	50000
Additional variables					
Post-Cold War years	-0.0133	0.0085	72.756	0.9469	7800
UCDP civil war onset	0.0531	0.0321	53.068	0.9452	4596
Previous riots	0.0140	0.0101	56.139	0.9200	7811
UCDP ongoing civil war	0.0170	0.0116	64.487	0.9057	4497
Ethnic diversity (ELF)	0.0184	0.0137	56.814	0.9056	7808
Polity IV squared	-0.0002	0.0001	60.825	0.9009	7903

Table 9: EBA – No VIF Restriction

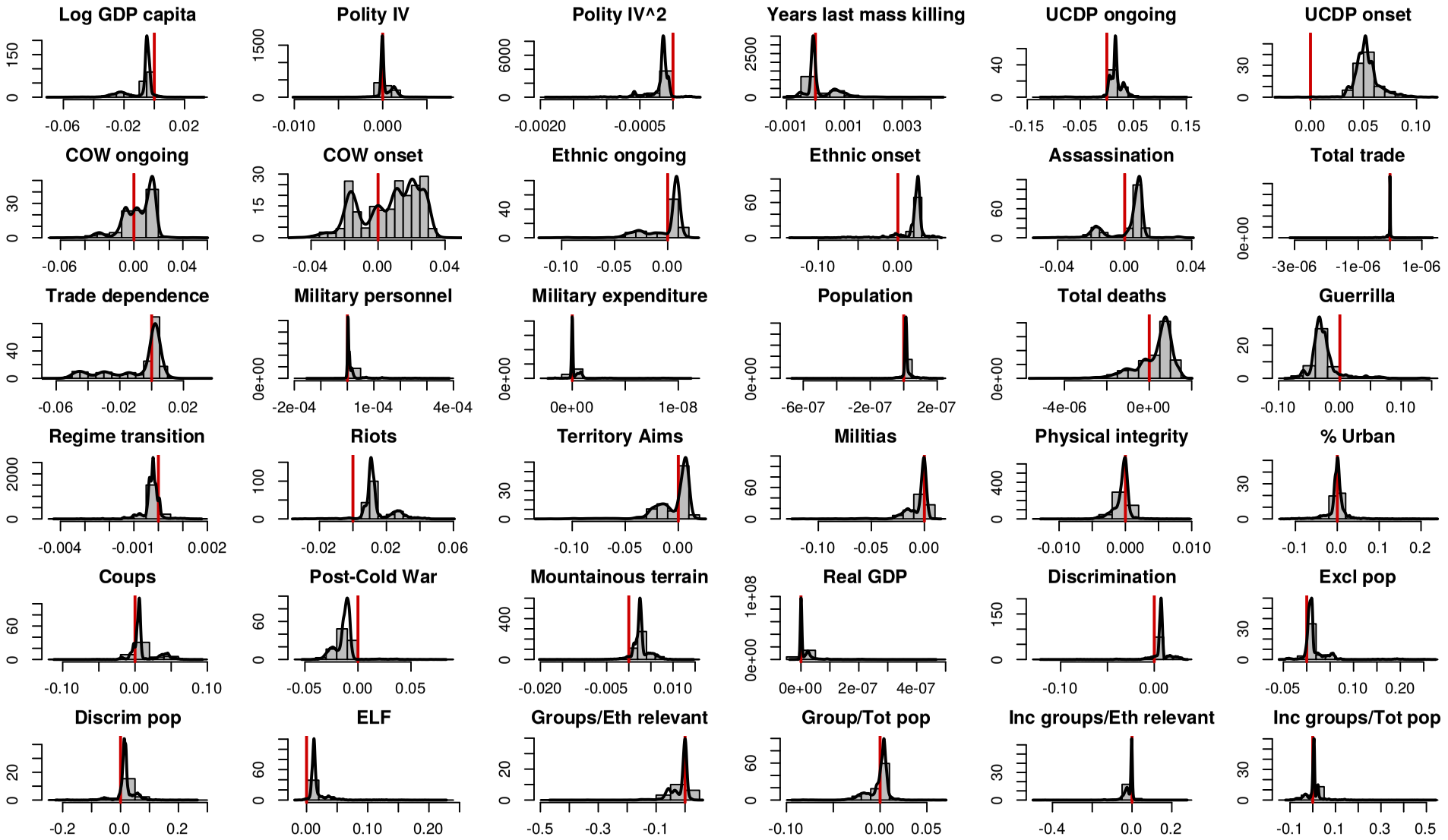

Figure 9: EBA – No VIF restriction

1.3.5 Generalised Linear Models

We reestimate the main EBA model with logit and probit models. Nevertheless, logistic and probit regressions may have issues of complete separation, that is, some covariates may perfectly separate zeros and ones in the outcome variable. In that case, the estimations fail to converge. We address this problem by adding a weak prior to the regression coefficients as suggested by Gelman et al. (2008).[1] First, we scaled the non-binary variables to have a mean of 0 and a standard deviation of 0.5, then added a Cauchy distribution with centre 0 and scale 2.5. The probit regressions use a scale of 2.5×1.6, which is also recommended by the authors (Gelman and Su 2016). Ethnic diversity and ongoing civil wars come close to meeting our threshold values (0.88 and 0.84, respectively), and civil war onset (UCDP) has a higher percentage of significant coefficients and a high CDF(0) area than in the linear probability models.

Variable	**Avg. β**	**Avg. SE**	**% Sig.**	**CDF(0)**	**Models**
Base variables					
Log GDP per capita	0.434	0.223	75.570	0.9267	50000
Additional variables					
UCDP civil war onset	1.308	0.530	87.261	0.9742	4506
Post-Cold War years	-0.911	0.428	70.456	0.9448	7890
Previous riots	0.744	0.38	66.778	0.9383	7805
Polity IV squared	-0.015	0.008	68.038	0.9285	7975

Table 10: EBA – Logistic Regression

[1] We thank Mark Bell for sharing R code to estimate penalised-likelihood models.

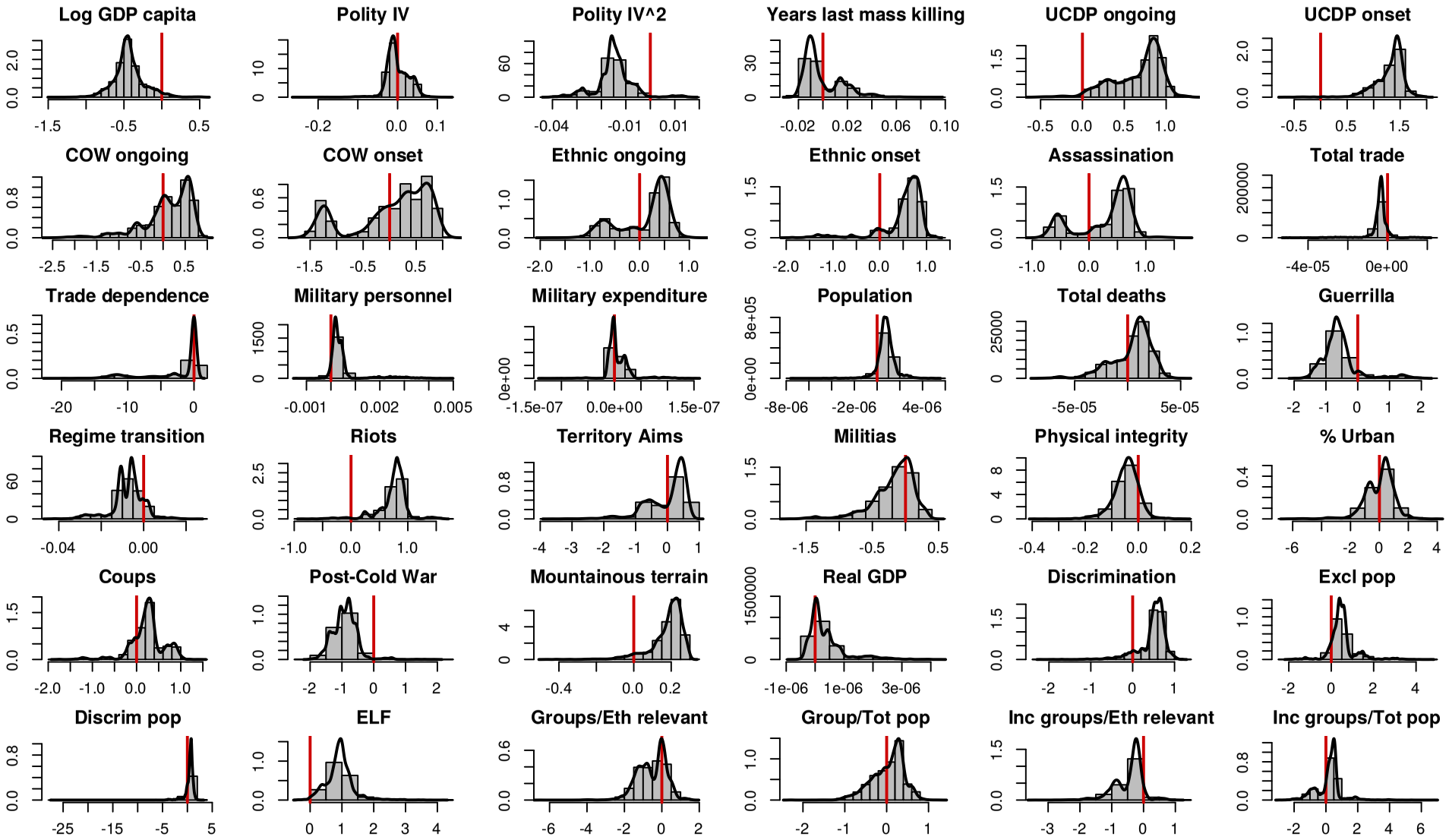

Figure 10: EBA – Logistic Regression

Variable	Avg. β	Avg. SE	% Sig.	CDF(0)	Models
Base variables					
Log GDP per capita	-0.1924	0.1031	76.118	0.9258	50000
Additional variables					
UCDP civil war onset	0.6422	0.2582	89.225	0.9772	4501
Previous riots	0.3367	0.1743	71.813	0.9436	7851
Post-Cold War years	-0.3709	0.1830	71.465	0.9404	7836
Polity IV squared	-0.0061	0.0032	70.155	0.9315	7931

Table 11: EBA – Probit Regression

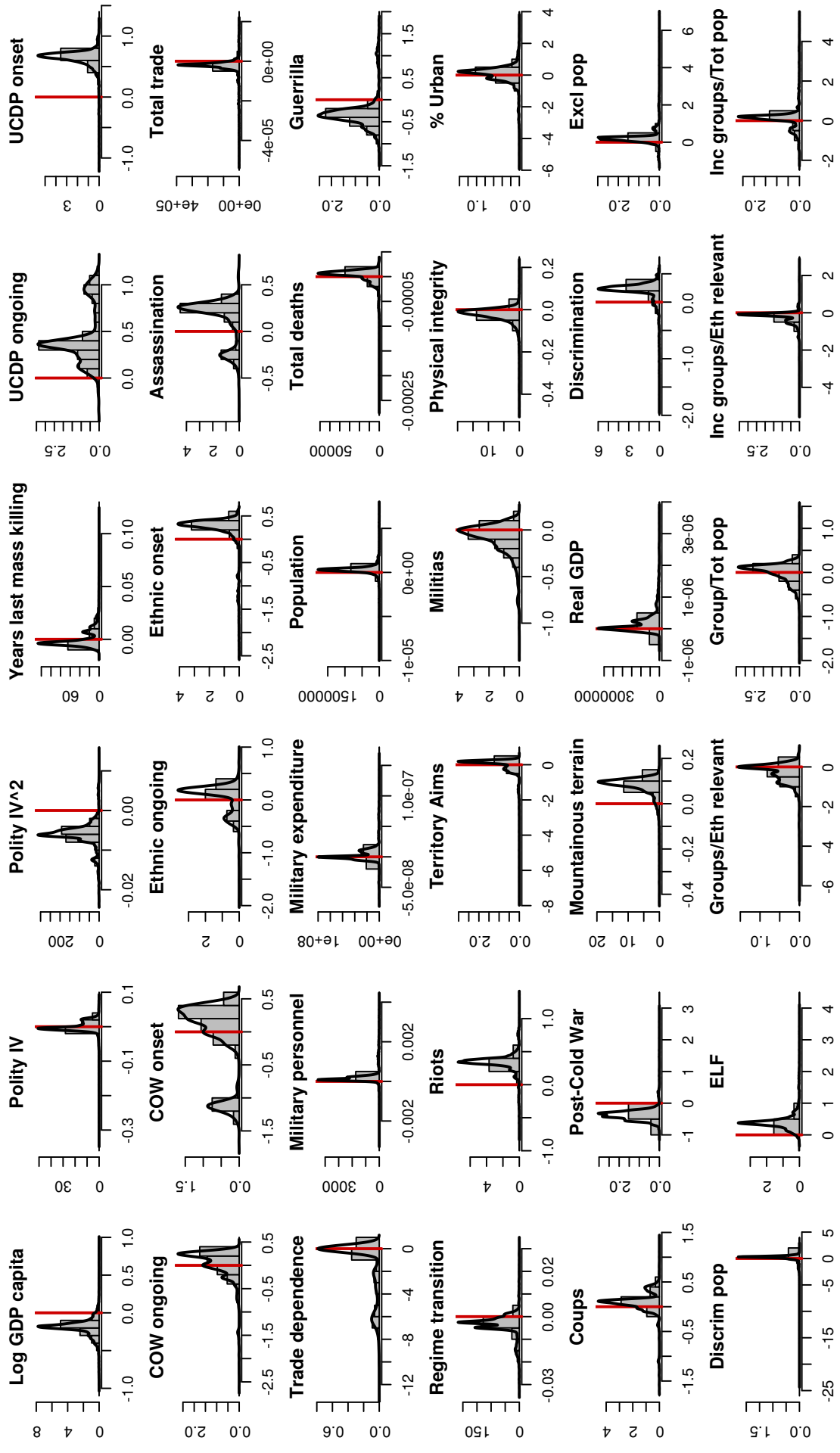

Figure 11: EBA – Probit Regression

1.4 Random Forest Extensions

1.4.1 Alternative Random Seeds

As noted in the article, we perform a grid search to optimise the hyperparameters of the random forest models. The grid search evaluates a wide range of parameter values at once, therefore it is generally unnecessary to run additional tests to assess the robustness of the results. Nevertheless, as random forests themselves are an approximation to a number of possible parameter combinations, changes in seed numbers may influence the output. Thus, we start the models with different random seed numbers to evaluate how sturdy are our original results.[2] The findings holds quite well: Although variable importance changes from one model to another, the most significant variables appear repeatedly in the estimations. The marginal plots also show that their effects on the outcome variable remains similar despite eventual nonlinearities. We show the six most significant predictors of mass killings and their respective partial dependence plots.

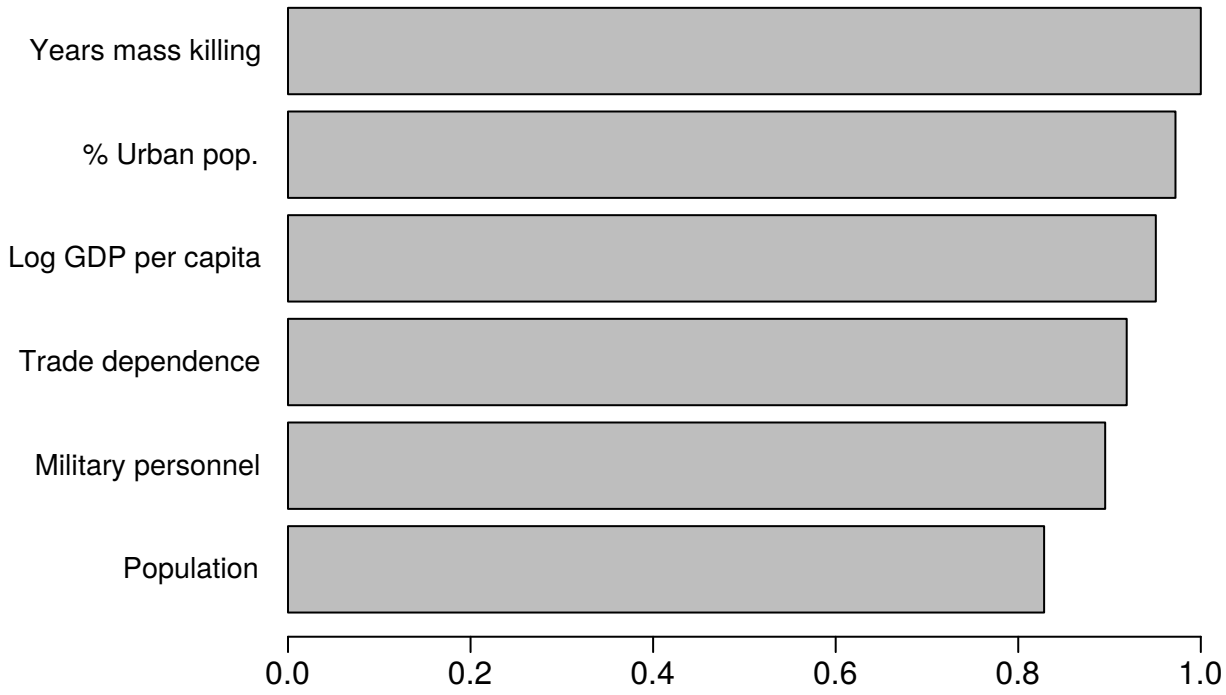

Figure 12: Variable Importance – Seed 44849999

[2]The numbers were generated at https://www.random.org/.

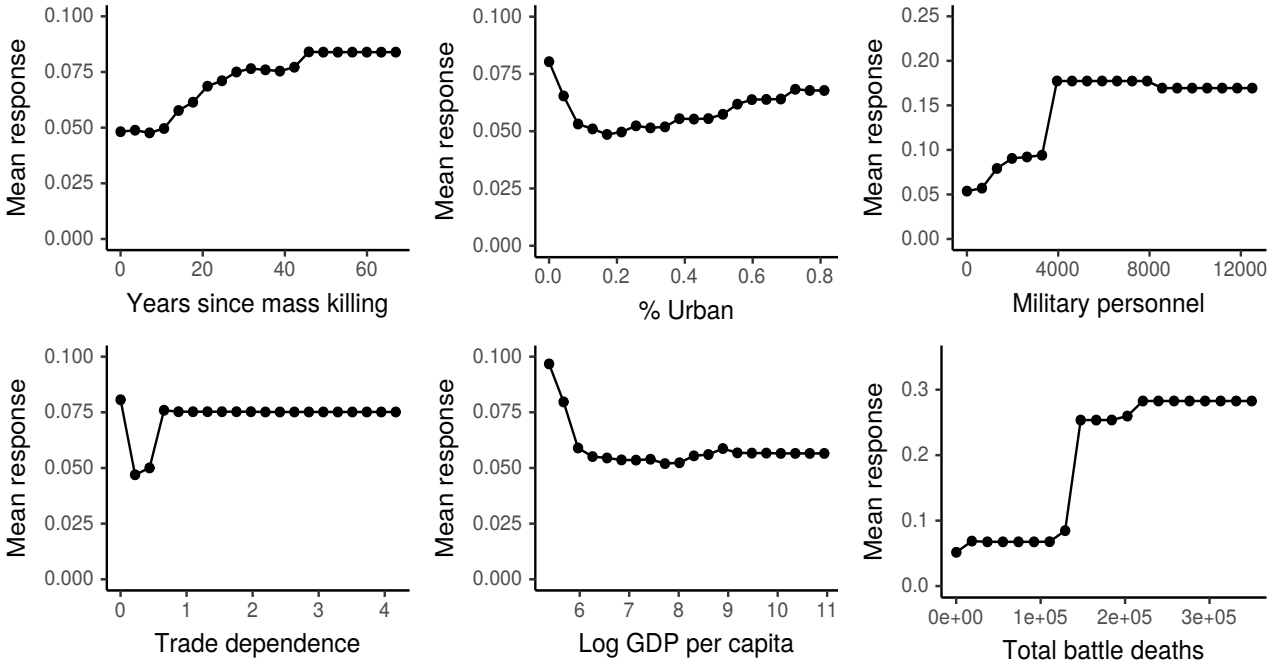

Figure 13: Partial Plots – Seed 44849999

Second model – seed 1502436:

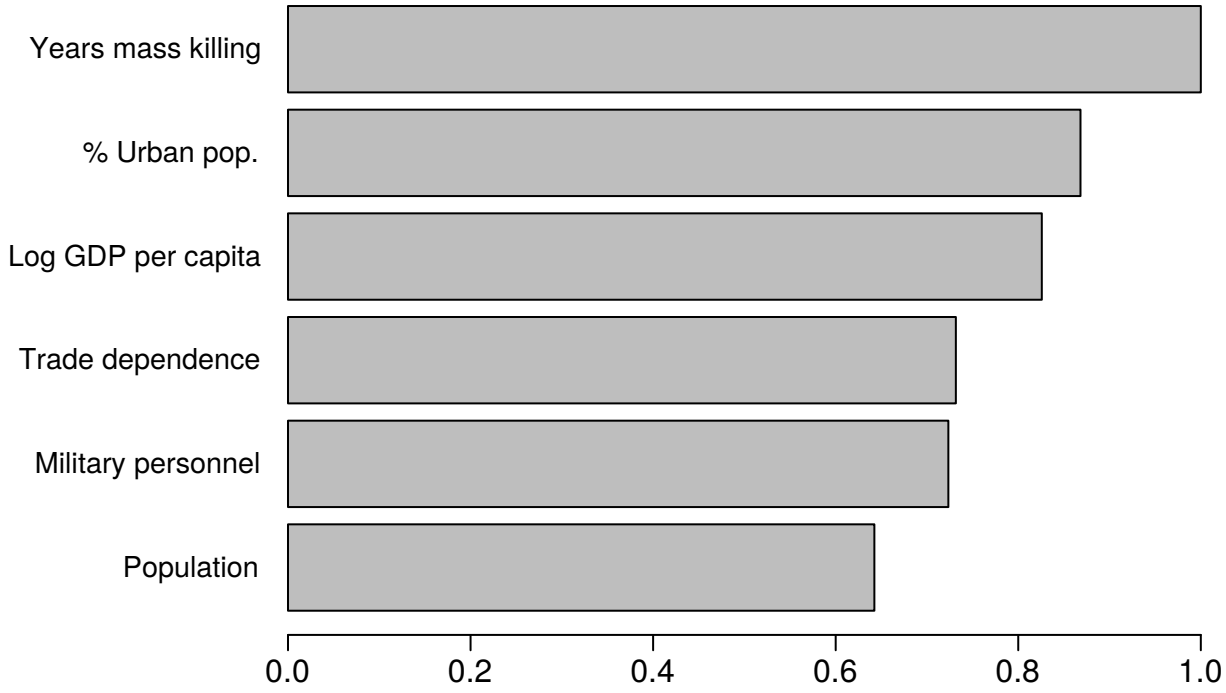

Figure 14: Variable Importance – Seed 1502436

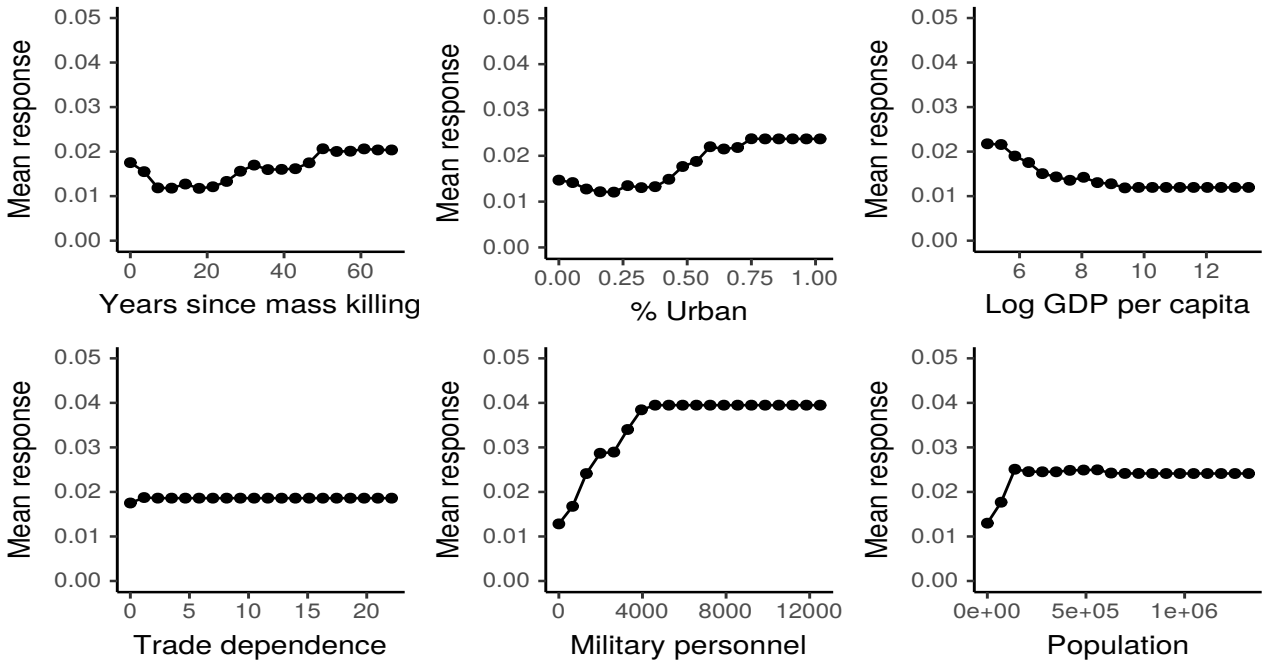

Figure 15: Partial Plots – Seed 1502436

1.5 Genocides/Politicides

In this section, we evaluate the models presented above with a measure of genocide and politicide by Harff (2003). The results show important contrasts with the previous analyses. First, no variable appear as significant in the main extreme bounds analysis. That is, none of the 36 predictors reached the threshold of CDF(0) > 0.9. The variable that came closest to significance was a dummy indicator of coups d'état, which has a CDF(0) of 0.897 and, as expected, is positively correlated with the onset of genocides. The distribution of the covariates' coefficients are available in figure 16.

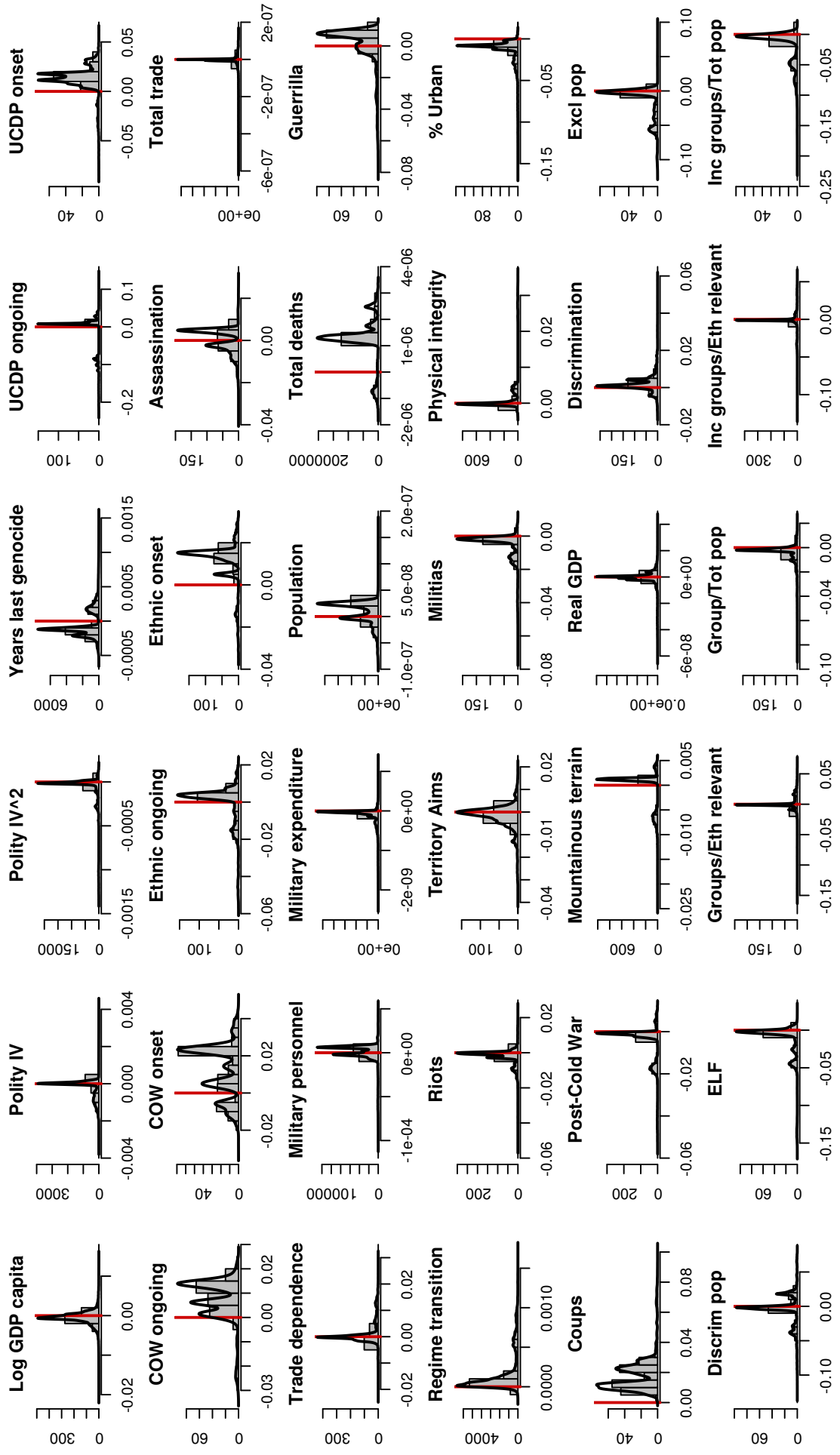

Figure 16: EBA – Genocides/Politicides

1.5.1 Genocides/Politicides during Civil Wars

Next, we evaluate what covariates are robust when considering only genocides and politicide that occur during civil conflicts. Post-Cold War years again appear as a significant variable and with a negative sign; excluded population also has a negative impact on the outcome variable in two analyses.

Variable	**Avg. β**	**Avg. SE**	**% Sig.**	**CDF(0)**	**Models**
UCDP data					
Excluded population	-0.037	0.022	64.524	0.9176	8758
COW data					
Excluded population	-0.057	0.031	65.703	0.9570	8820
Discriminated population	-0.050	0.029	53.850	0.93.67	8767
Post-Cold War years	-0.019	0.013	42.531	0.9203	8904
Cederman et al. data					
Assassination dummy	-0.009	0.006	47.723	0.9232	8828

Table 12: EBA – Genocides/Politicides

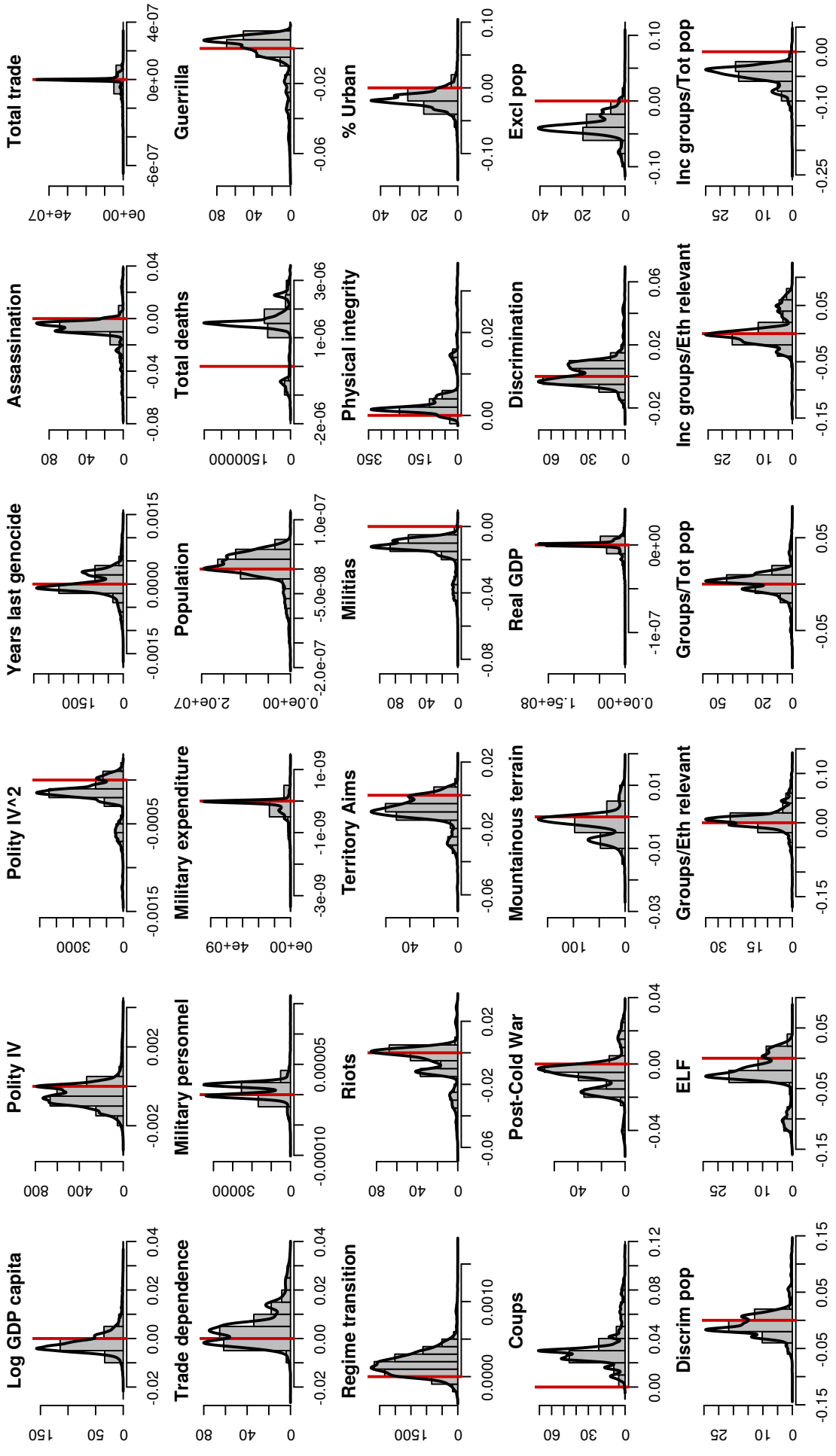

Figure 17: EBA – Genocides and Politicides during Civil Wars (UCDP Data)

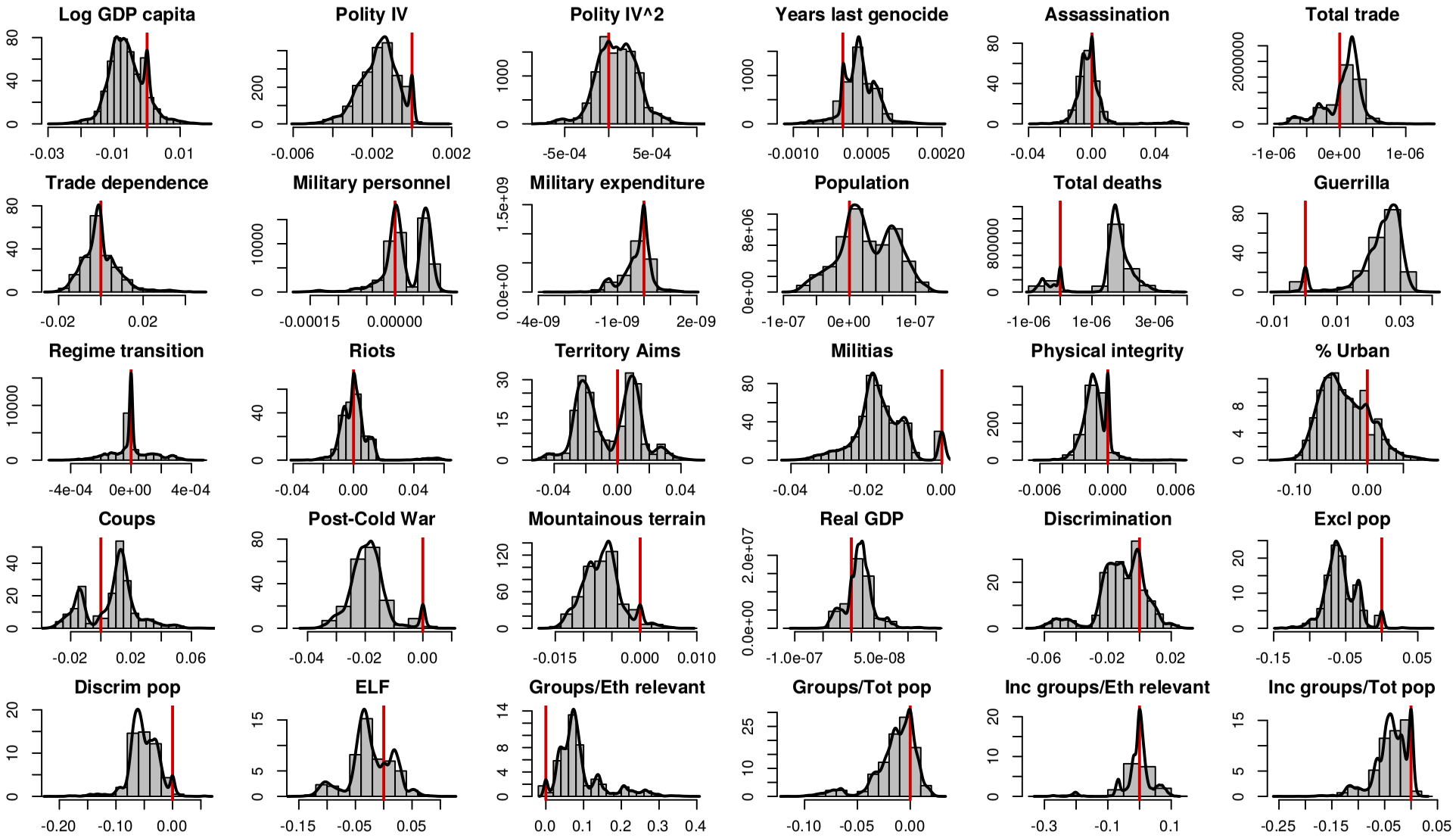

Figure 18: EBA – Genocides and Politicides during Civil Wars (COW Data)

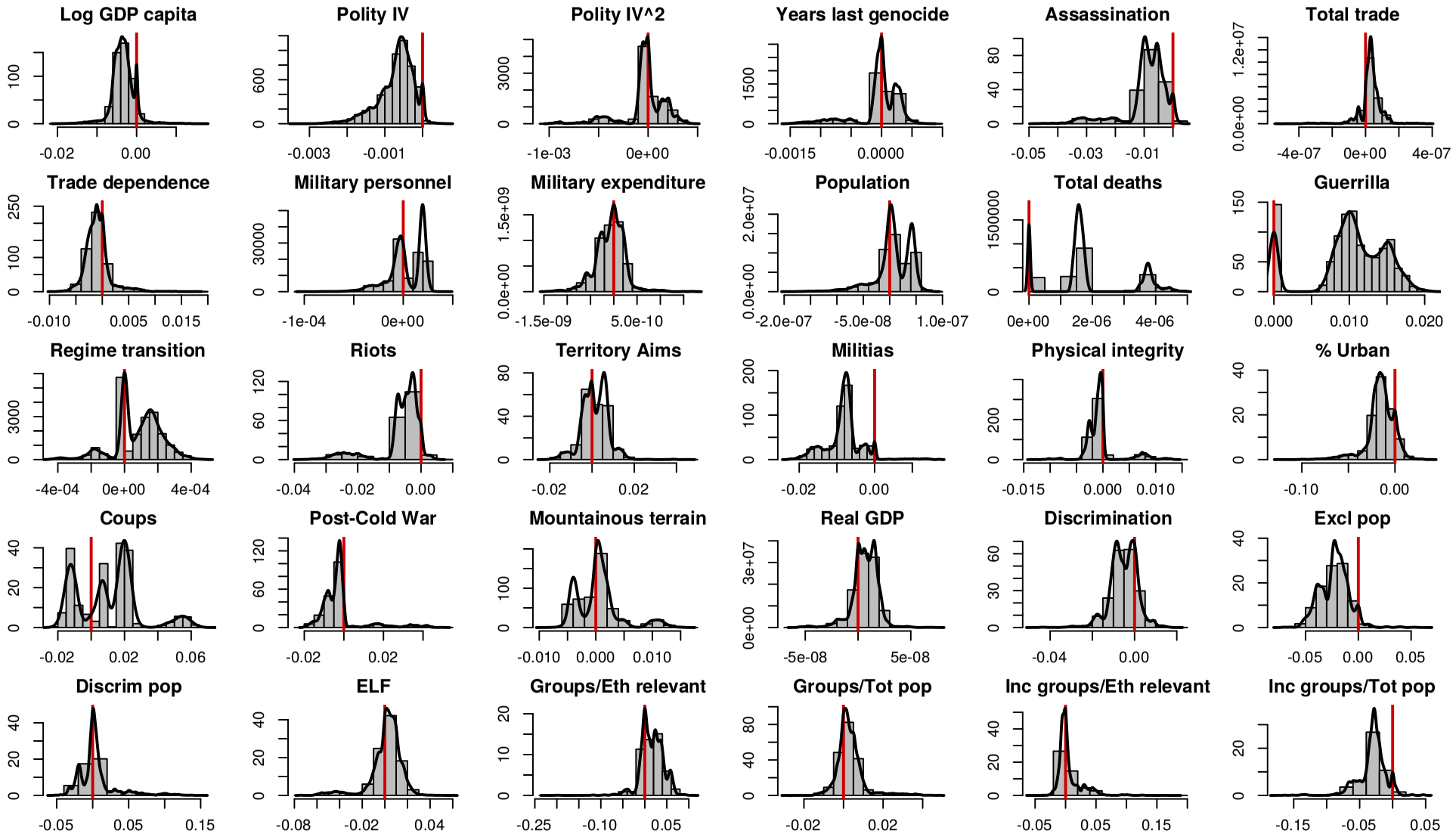

Figure 19: EBA – Genocides and Politicides during Ethnic Civil Wars (Cederman et al. Data)

1.6 Genocides/Politicides – Random Forests

Lastly, we present three models using the distributed random forest algorithm (The H2O.ai Team 2017). The results are in line with those obtained with the mass killing variable by Ulfelder and Valentino (2008). Again, we see that CINC, the percentage of urban population, and variables concerning the military are some of the most important predictors of state-led violence. The results confirm the overall finding of the chapter: Poor countries are more likely to experience mass killings, and states with a stronger army see a significant upward shift in genocide risk.

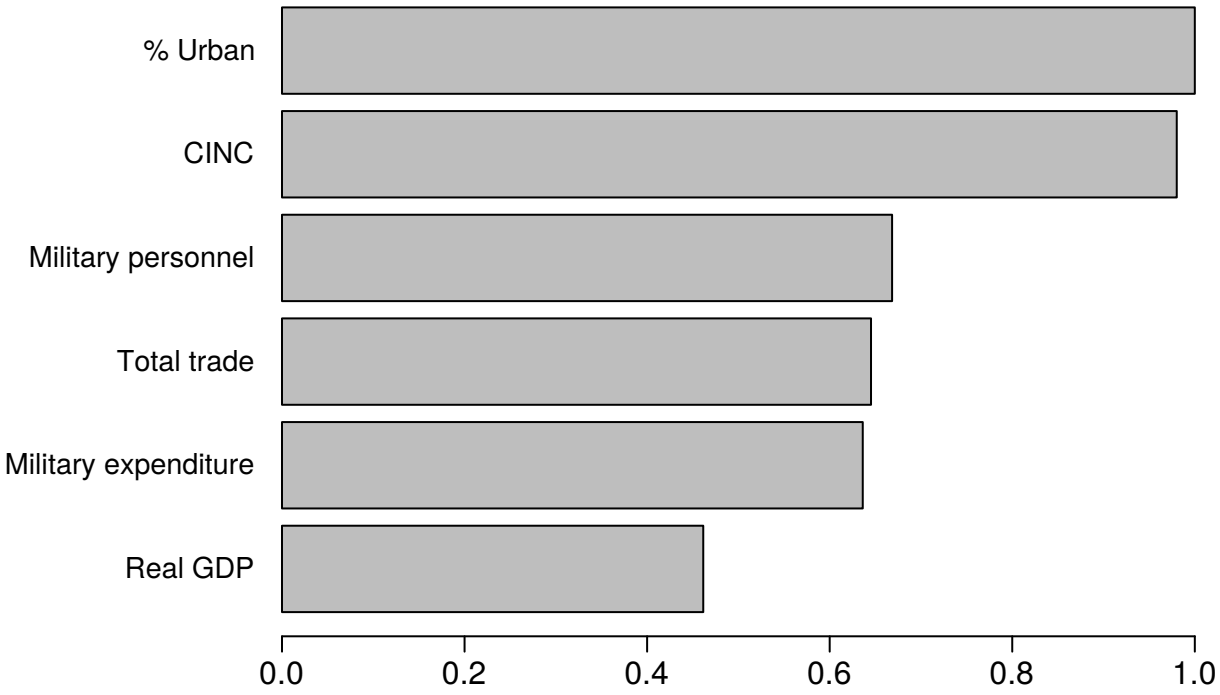

Figure 20: Variable Importance – Genocides/Politicides

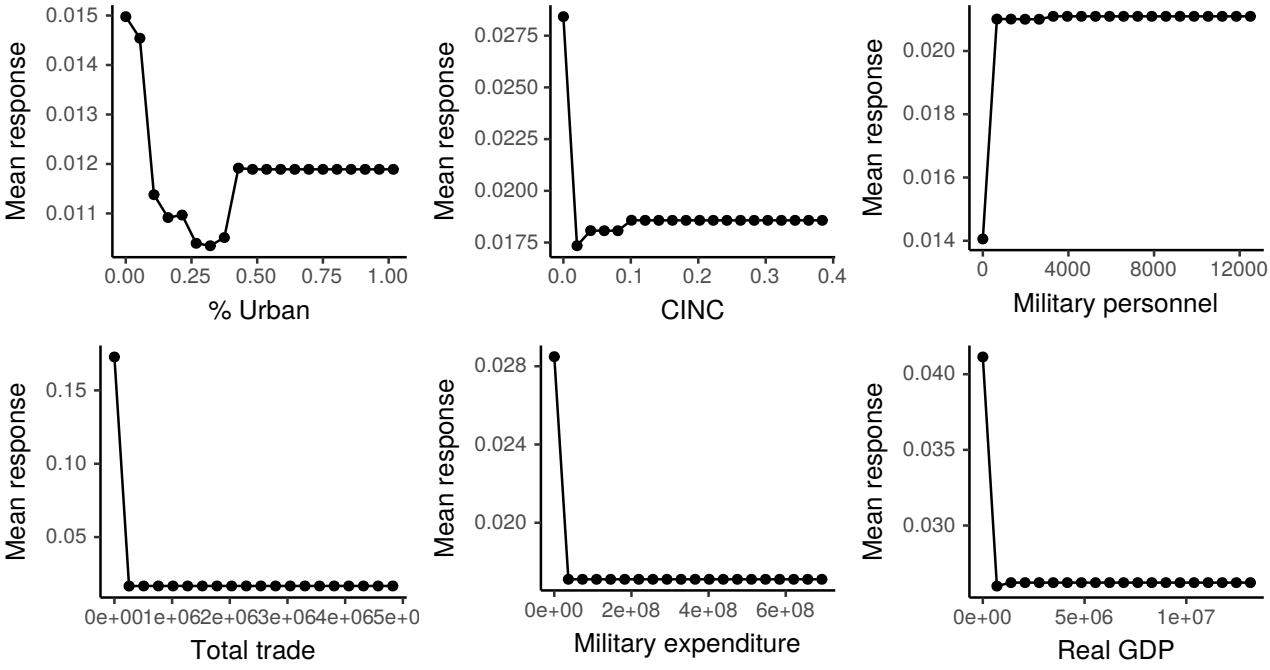

Figure 21: Partial Plots – Genocides/Politicides

The next graphs show the most important predictors of genocides that occur during civil wars and their respective partial dependence plots.

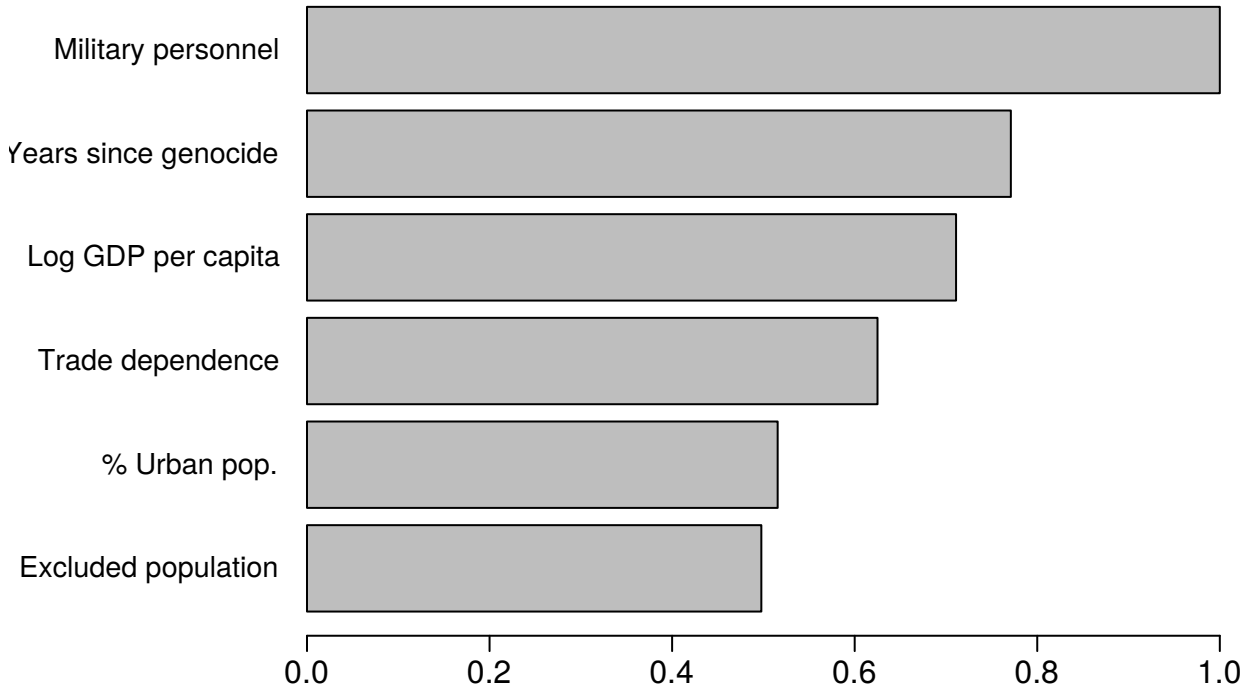

Figure 22: Variable Importance – Genocides/Politicides (UCDP Data)

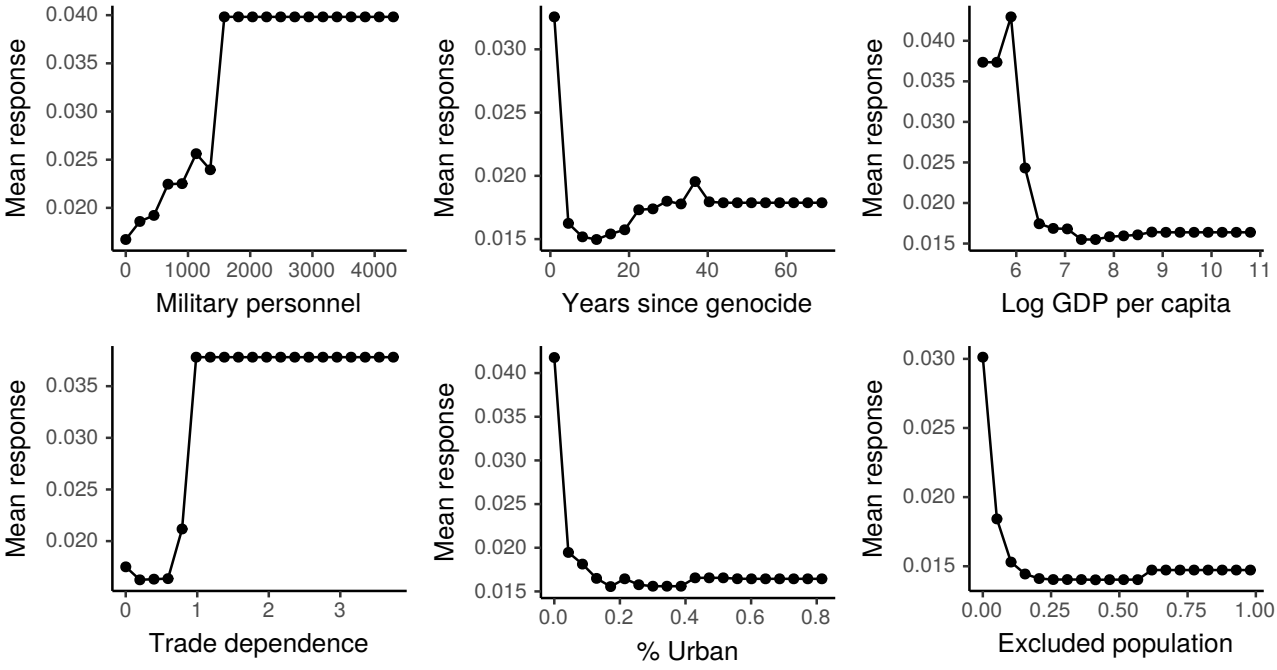

Figure 23: Partial Plots – Genocides/Politicides (UCDP Data)

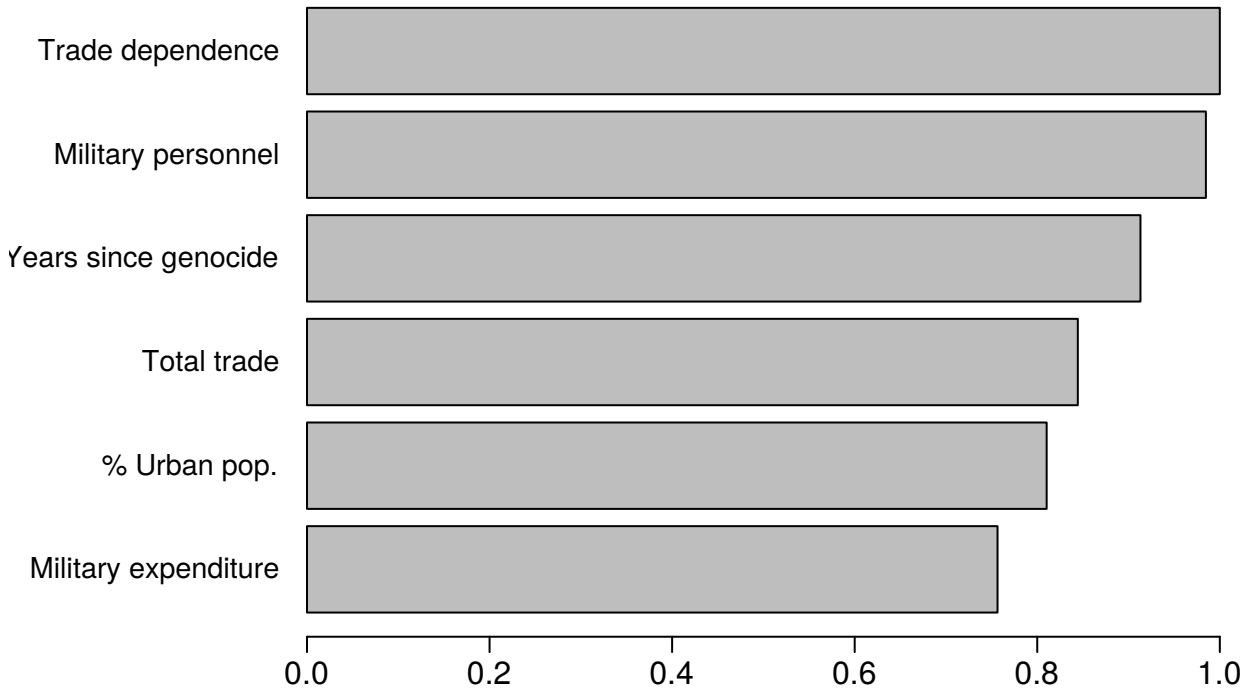

Figure 24: Variable Importance – Genocides/Politicides (COW Data)

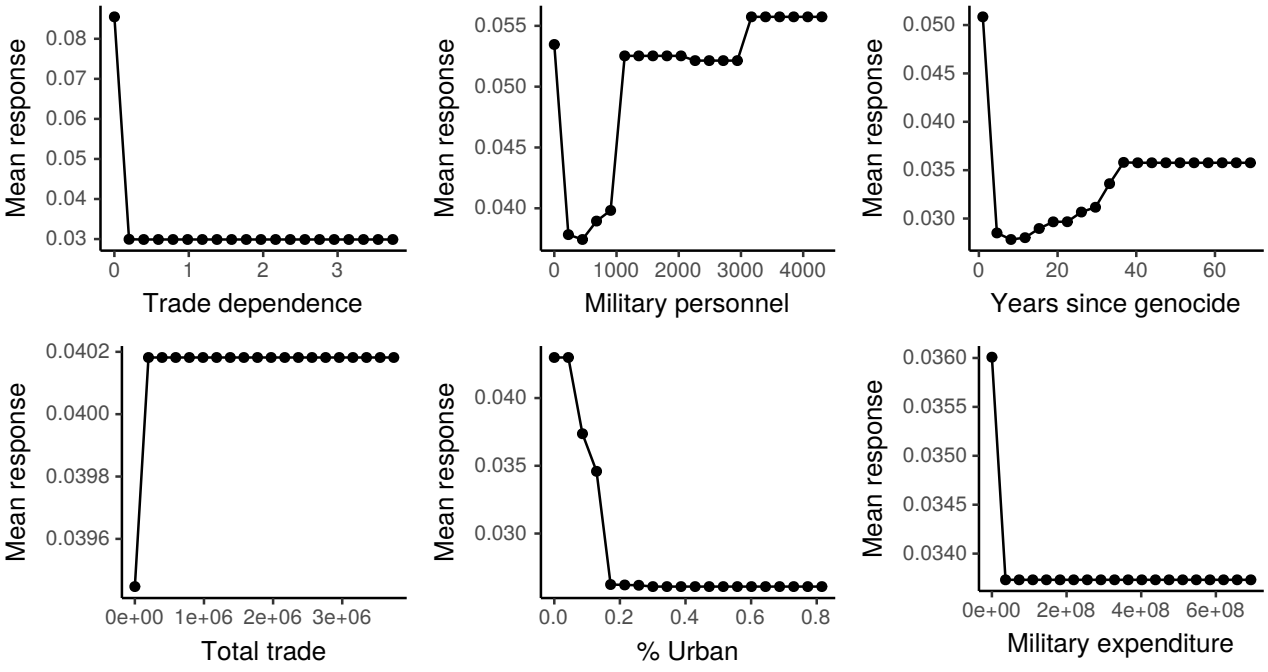

Figure 25: Partial Plots – Genocides/Politicides (COW Data)

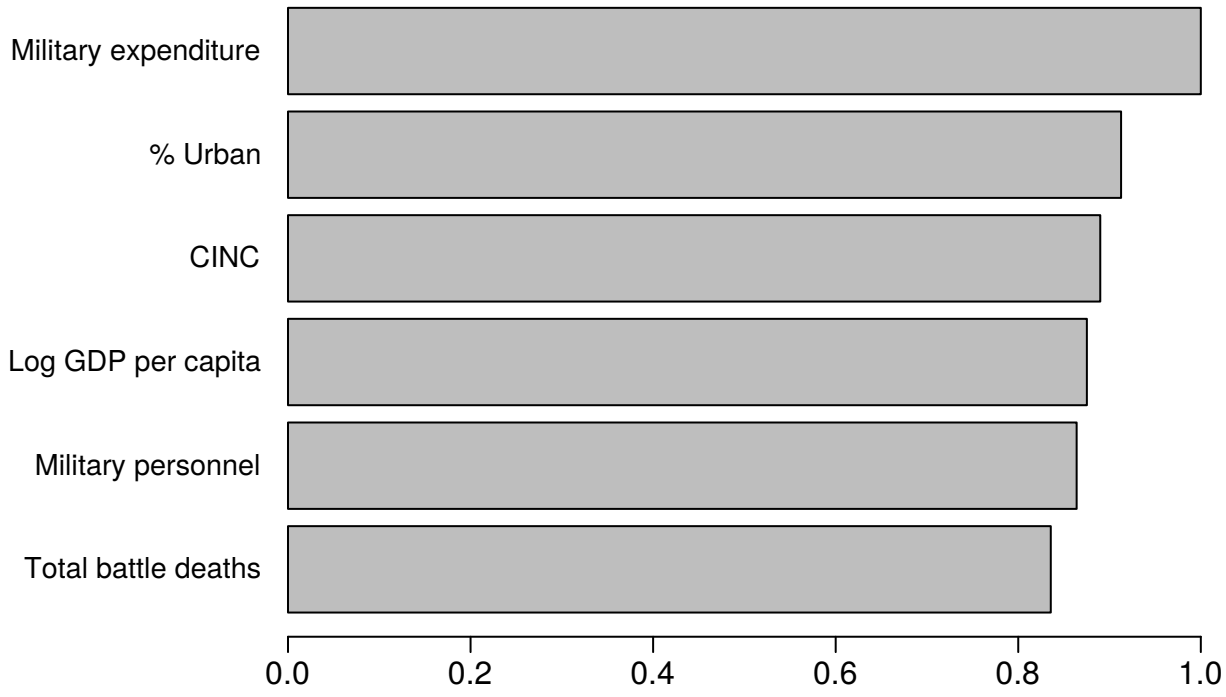

Figure 26: Variable Importance – Genocides/Politicides (Cederman et al. Data)

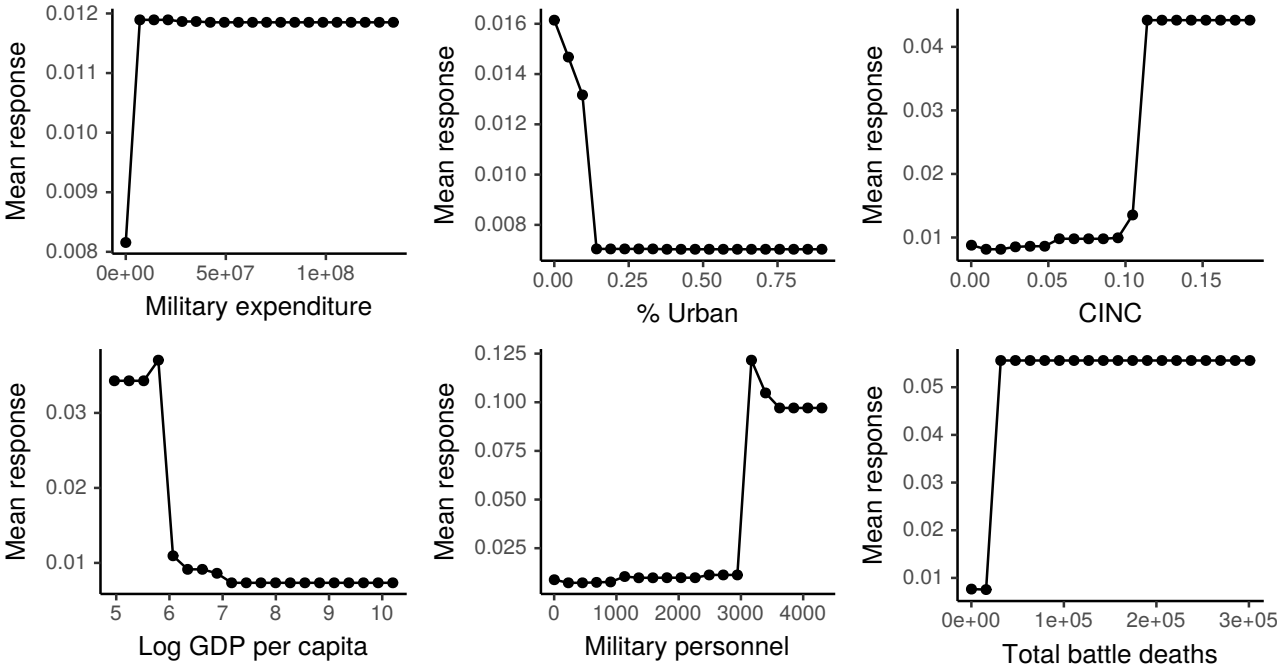

Figure 27: Partial Plots – Genocides/Politicides (Cederman et al. Data)

1.7 R Code

The R code below reproduces the analyses presented in the article.

```
######################
### Data Wrangling ###
######################

## Install and load required packages
if (!require("tidyverse")) {
        install.packages("tidyverse")
}
if (!require("data.table")) {
        install.packages("data.table")
}
if (!require("ExtremeBounds")) {
        install.packages("ExtremeBounds")
}
if (!require("h2o")) {
        install.packages("h2o")
}
if (!require("sandwich")) {
        install.packages("sandwich")
}
if (!require("arm")) {
        install.packages("arm")
}
if (!require("stargazer")) {
        install.packages("stargazer")
}

## Load dataset
df <- haven::read_dta("data/base variables.dta") %>% setDT()

## Select and lag variables
sd.cols <- c("UCDPcivilwarstart", "UCDPcivilwarongoing", "COWcivilwarstart",
             "COWcivilwarongoing", "ethnowarstart", "ethnowarongoing",
             "assdummy", "demdummy", "elf", "lmtnest", "pop", "realgdp",
             "rgdppc", "polity2", "exclpop", "discpop", "polrqnew",
             "poltrqnew", "egiptpolrqnew", "egippolrqnew", "discrim",
             "elf2", "interstatewar", "milex", "milper", "percentpopurban",
             "postcoldwar", "coupdummy", "riotdummy", "territoryaims",
             "totaltrade", "tradedependence", "militias", "physint", "cinc",
             "totalbeaths", "change", "guerrilladummy", "sf", "regtrans")

df1 <- cbind(df, df[, shift(.SD, 1, give.names = TRUE),
                    by = ccode, .SDcols = sd.cols])

# Remove the second `ccode` variable
df1 <- as.data.frame(df1[, -c(70)])

# Add new variables
df1$logrgdppc_lag_1 <- log(df1$rgdppc_lag_1)
df1$polity2sq_lag_1 <- df1$polity2_lag_1^2

# UCDP civil war == 1
df.ucdp <- df1 %>% filter(UCDPcivilwarongoing == 1)
df.ucdp <- as.data.frame(df.ucdp[, c(1:7, 76:111)])
names(df.ucdp) <- sub("_.*","", names(df.ucdp))
```

```
# COW civil war == 1
df.cow <- df1 %>% filter(COWcivilwarongoing == 1)
df.cow <- as.data.frame(df.cow[, c(1:7, 76:111)])
names(df.cow) <- sub("_.*","", names(df.cow))

# Ethnic civil war == 1
df.eth <- df1 %>% filter(ethnowarongoing == 1)
df.eth <- as.data.frame(df.eth[, c(1:7, 76:111)])
names(df.eth) <- sub("_.*","", names(df.eth))

# Regular model
df2 <- as.data.frame(df1[, c(1:7, 70:111)])
names(df2) <- sub("_.*","", names(df2))


################################
### Extreme Bounds Analyses ###
################################

## Classifying a few variables as mutually exclusive.
## "Change" was removed because it was correlated at 0.99 with "regtrans".
free.variables <- c("logrgdppc", "polity2", "mksyr")
civilwar.variables <- c("UCDPcivilwarongoing", "UCDPcivilwarstart",
                        "COWcivilwarongoing", "COWcivilwarstart",
                        "ethnowarongoing", "ethnowarstart")
doubtful.variables <- c("UCDPcivilwarongoing", "UCDPcivilwarstart",
                        "COWcivilwarongoing", "COWcivilwarstart",
                        "ethnowarongoing", "ethnowarstart", "assdummy",
                        "totaltrade", "tradedependence", "milper", "milex",
                        "pop", "totalbeaths", "guerrilladummy", "regtrans",
                        "riotdummy", "territoryaims", "militias",
                        "physint", "percentpopurban", "coupdummy",
                        "postcoldwar",  "lmtnest", "realgdp", "discrim",
                        "exclpop", "discpop", "elf",  "polrqnew",
                        "egippolrqnew", "poltrqnew", "egiptpolrqnew",
                        "polity2sq")

# Cluster-robust standard errors
se.clustered.robust <- function(model.object){
        model.fit <- vcovHC(model.object, type = "HC", cluster = "country")
        out <- sqrt(diag(model.fit))
        return(out)
}

# Main model
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:4,
          data = df2, vif = 7, level = 0.9,
          se.fun = se.clustered.robust)

summary(m1)
hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "mksyr",
                       "UCDPcivilwarongoing",
                       "UCDPcivilwarstart", "COWcivilwarongoing",
                       "COWcivilwarstart", "ethnowarongoing", "ethnowarstart",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
                       "riotdummy", "territoryaims", "militias", "physint",
                       "percentpopurban", "coupdummy", "postcoldwar",
```

```
                        "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                        "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                        "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last mass killing",
              "UCDP ongoing", "UCDP onset", "COW ongoing", "COW onset",
              "Ethnic ongoing", "Ethnic onset", "Assassination", "Total trade",
          "Trade dependence", "Military personnel", "Military expenditure", "Population",
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Group/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

# Mass killings during civil war
# UCDP civil conflicts == 1
doubtful.variables <- c("assdummy", "totaltrade", "tradedependence",
                        "milper", "milex", "pop", "totalbeaths",
                        "guerrilladummy", "regtrans", "riotdummy",
                        "territoryaims", "militias", "physint",
                        "percentpopurban", "coupdummy", "postcoldwar",
                        "lmtnest", "realgdp", "discrim", "exclpop",
                        "discpop", "elf",  "polrqnew", "egippolrqnew",
                        "poltrqnew", "egiptpolrqnew", "polity2sq")

m1 <- eba(y = "MKstart", free = free.variables,
          doubtful = doubtful.variables, k = 0:4,
          data = df.ucdp, vif = 7,
          level = 0.9, se.fun = se.clustered.robust)

summary(m1)
hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "mksyr",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
                       "riotdummy", "territoryaims", "militias", "physint",
                       "percentpopurban", "coupdummy", "postcoldwar",
                       "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                       "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                       "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last mass killing",
              "Assassination", "Total trade",
          "Trade dependence", "Military personnel", "Military expenditure", "Population",
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Group/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

# COW civil wars == 1
doubtful.variables <- c("assdummy", "totaltrade", "tradedependence",
                        "milper", "milex", "pop", "totalbeaths",
                        "guerrilladummy", "regtrans", "riotdummy",
                        "territoryaims", "militias", "physint",
                        "percentpopurban", "coupdummy", "postcoldwar",
                        "lmtnest", "realgdp", "discrim", "exclpop",
                        "discpop", "elf",  "polrqnew", "egippolrqnew",
                        "poltrqnew", "egiptpolrqnew", "polity2sq")

m1 <- eba(y = "MKstart", free = free.variables,
```

```
          doubtful = doubtful.variables, k = 0:4,
          data = df.cow, vif = 7,
          level = 0.9, se.fun = se.clustered.robust)

summary(m1)
hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "mksyr",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
                       "riotdummy", "territoryaims", "militias", "physint",
                       "percentpopurban", "coupdummy", "postcoldwar",
                       "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                       "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                       "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last mass killing",
              "Assassination", "Total trade",
          "Trade dependence", "Military personnel", "Military expenditure", "Population",
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Group/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

# Ethnic civil war == 1
doubtful.variables <- c("assdummy", "totaltrade", "tradedependence",
                        "milper", "milex", "pop", "totalbeaths",
                        "guerrilladummy", "regtrans", "riotdummy",
                        "territoryaims", "militias", "physint",
                        "percentpopurban", "coupdummy", "postcoldwar",
                        "lmtnest", "realgdp", "discrim", "exclpop",
                        "discpop", "elf",  "polrqnew", "egippolrqnew",
                        "poltrqnew", "egiptpolrqnew", "polity2sq")

m1 <- eba(y = "MKstart", free = free.variables,
          doubtful = doubtful.variables, k = 0:4,
          data = df.eth, vif = 7,
          level = 0.9, se.fun = se.clustered.robust)

summary(m1)
hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "mksyr",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
                       "riotdummy", "territoryaims", "militias", "physint",
                       "percentpopurban", "coupdummy", "postcoldwar",
                       "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                       "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                       "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last mass killing",
              "Assassination", "Total trade",
          "Trade dependence", "Military personnel", "Military expenditure", "Population",
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Group/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

## Different values of k
## Code for the histogram not included as it is the same as that of the main model.
```

```
# 3 variables per model
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:3,
          data = df2, vif = 7, level = 0.9, draws = 50000,
          se.fun = se.clustered.robust)

# 5 variables per model
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:5,
          data = df2, vif = 7, draws = 50000,
          level = 0.9, se.fun = se.clustered.robust)

## Alternative VIFs

# VIF = 10
# Low VIF
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:4,
          data = df2, vif = 10, level = 0.9, draws = 50000,
          se.fun = se.clustered.robust)

# VIF = 2.5
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:4,
          data = df2, vif = 2.5, level = 0.9, draws = 50000,
          se.fun = se.clustered.robust)

# No VIF
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:4,
          data = df2, level = 0.9, draws = 50000,
          se.fun = se.clustered.robust)

## Generalised linear models

# Logit
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:4,
          data = df2, level = 0.9, vif = 7, draws = 50000,
          reg.fun = bayesglm, family = binomial(link = "logit"))

# Probit
m1 <- eba(y = "MKstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:4,
          data = df2, level = 0.9, vif = 7, draws = 50000,
          reg.fun = bayesglm, family = binomial(link = "probit"))


###################################
### Distributed Random Forests ###
###################################

## Random Forests
```

```
## Prepare the data set
h2o.init(nthreads = -1, max_mem_size = "20g") # memory size

df2a <- as.h2o(df2)

df2a$MKstart <- as.factor(df2a$MKstart)  # encode the binary response as a factor
h2o.levels(df2a$MKstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df2a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility

train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "MKstart"
x <- setdiff(names(df2), c(y, "ccode", "year", "rgdppc",
                           "mksyr2", "mksyr3", "sf", "country",
                           "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf01",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

# Saving the most accurate model
rf.grid <- h2o.getGrid(grid_id = "gridrf01",
                       sort_by = "auc",
                       decreasing = TRUE)

rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

# Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf01_model_21")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))
```

```
par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Polity IV",
                      "Military personnel",
                      "Ethnic polarisation",
                      "% Urban pop.",
                      "Years mass killing",
                      "Log GDP per capita"),
        main = "")

logrgdppc <- h2o.partialPlot(object = a, data = train, cols = c("logrgdppc"))
p1 <- qplot(logrgdppc$logrgdppc, logrgdppc$mean_response) + geom_line() + theme_classic() + ylim(0,
        xlab("Log GDP per capita") + ylab("Mean response")

mksyr <- h2o.partialPlot(object = a, data = train, cols = c("mksyr"))
p2 <- qplot(mksyr$mksyr, mksyr$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Years since mass killing") +  ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p3 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
        theme_classic() + ylim(0, 0.05) + xlab("% Urban") + ylab("Mean response")

egiptpolrqnew <- h2o.partialPlot(object = a, data = train, cols = c("egiptpolrqnew"))
p4 <- qplot(egiptpolrqnew$egiptpolrqnew, egiptpolrqnew$mean_response) + geom_line() +
     theme_classic() + ylim(0, 0.05) + xlab("Ethnic polarisation") + ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
p5 <- qplot(milper$milper, milper$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Military personnel") + ylab("Mean response")

polity2 <- h2o.partialPlot(object = a, data = train, cols = c("polity2"))
p6 <- qplot(polity2$polity2, polity2$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Polity IV") + ylab("Mean response")

# Multiplot function: http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_(ggplot2)/
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
        library(grid)

        # Make a list from the ... arguments and plotlist
        plots <- c(list(...), plotlist)

        numPlots = length(plots)

        # If layout is NULL, then use 'cols' to determine layout
        if (is.null(layout)) {
                # Make the panel
                # ncol: Number of columns of plots
                # nrow: Number of rows needed, calculated from # of cols
                layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                                 ncol = cols, nrow = ceiling(numPlots/cols))
        }

        if (numPlots==1) {
                print(plots[[1]])

        } else {
                # Set up the page
                grid.newpage()
               pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))
```

```
                # Make each plot, in the correct location
                for (i in 1:numPlots) {
                  # Get the i,j matrix positions of the regions that contain this subplot
                        matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

                        print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                                        layout.pos.col = matchidx$col))
                }
        }
}

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)

## Different seeds

rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf01b",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 44849999))

# Saving the most accurate model
rf.grid <- h2o.getGrid(grid_id = "gridrf01b",
                       sort_by = "auc",
                       decreasing = TRUE)

rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

# Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf01b_model_8")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Population",
                      "Military personnel",
                      "Trade dependence",
                      "Log GDP per capita",
                      "% Urban pop.",
```

```
                      "Years mass killing"),
        main = "")

mksyr <- h2o.partialPlot(object = a, data = train, cols = c("mksyr"))
p1 <- qplot(mksyr$mksyr, mksyr$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Years since mass killing") +  ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p2 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
        theme_classic() + ylim(0, 0.05) + xlab("% Urban") + ylab("Mean response")

logrgdppc <- h2o.partialPlot(object = a, data = train, cols = c("logrgdppc"))
p3 <- qplot(logrgdppc$logrgdppc, logrgdppc$mean_response) + geom_line() + theme_classic() + ylim(0,
        xlab("Log GDP per capita") + ylab("Mean response")

tradedependence <- h2o.partialPlot(object = a, data = train, cols = c("tradedependence"))
p4 <- qplot(tradedependence$tradedependence, tradedependence$mean_response) + geom_line() +
      theme_classic() + ylim(0, 0.05) + xlab("Trade dependence") + ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
p5 <- qplot(milper$milper, milper$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Military personnel") + ylab("Mean response")

pop <- h2o.partialPlot(object = a, data = train, cols = c("pop"))
p6 <- qplot(pop$pop, pop$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Population") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)


rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf01c",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 1502436))

# Saving the most accurate model
rf.grid <- h2o.getGrid(grid_id = "gridrf01c",
                       sort_by = "auc",
                       decreasing = TRUE)

rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
```

```
h2o.performance(rf2, newdata = test)

# Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf01c_model_58")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Population",
                      "Military personnel",
                      "Trade dependence",
                      "Log GDP per capita",
                      "% Urban pop.",
                      "Years mass killing"),
        main = "")

mksyr <- h2o.partialPlot(object = a, data = train, cols = c("mksyr"))
p1 <- qplot(mksyr$mksyr, mksyr$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Years since mass killing") +  ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p2 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
        theme_classic() + ylim(0, 0.05) + xlab("% Urban") + ylab("Mean response")

logrgdppc <- h2o.partialPlot(object = a, data = train, cols = c("logrgdppc"))
p3 <- qplot(logrgdppc$logrgdppc, logrgdppc$mean_response) + geom_line() + theme_classic() + ylim(0,
        xlab("Log GDP per capita") + ylab("Mean response")

tradedependence <- h2o.partialPlot(object = a, data = train, cols = c("tradedependence"))
p4 <- qplot(tradedependence$tradedependence, tradedependence$mean_response) + geom_line() +
       theme_classic() + ylim(0, 0.05) + xlab("Trade dependence") + ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
p5 <- qplot(milper$milper, milper$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Military personnel") + ylab("Mean response")

pop <- h2o.partialPlot(object = a, data = train, cols = c("pop"))
p6 <- qplot(pop$pop, pop$mean_response) + geom_line() + theme_classic() + ylim(0, 0.05) +
        xlab("Population) + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)

## Genocides in civil wars

## UCDP data
df.ucdpa <- as.h2o(df.ucdp)

df.ucdpa$MKstart <- as.factor(df.ucdpa$MKstart)  # encode the binary response as a factor
h2o.levels(df.ucdpa$MKstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.ucdpa,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility

train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")
```

```
y <- "MKstart"
x <- setdiff(names(df.ucdp), c(y, "ccode", "year", "rgdppc",
                           "mksyr2", "mksyr3", "sf", "country",
                           "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf02",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf02",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

# Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf02_model_34")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Total battle deaths",
                      "Log GDP per capita",
                       "Trade dependence",
                      "Military personnel",
                      "% Urban pop.",
                      "Years mass killing"),
        main = "")

mksyr <- h2o.partialPlot(object = a, data = train, cols = c("mksyr"))
p1 <- qplot(mksyr$mksyr, mksyr$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.1) + xlab("Years since mass killing") +  ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p2 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
        ylim(0, 0.1) + theme_classic() + xlab("% Urban") + ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
```

```
p3 <- qplot(milper$milper, milper$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.25) + xlab("Military personnel") + ylab("Mean response")

tradedependence <- h2o.partialPlot(object = a, data = train, cols = c("tradedependence"))
p4 <- qplot(tradedependence$tradedependence, tradedependence$mean_response) + geom_line() + theme_c
        ylim(0, 0.1) + xlab("Trade dependence") + ylab("Mean response")

logrgdppc <- h2o.partialPlot(object = a, data = train, cols = c("logrgdppc"))
p5 <- qplot(logrgdppc$logrgdppc, logrgdppc$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.1) + xlab("Log GDP per capita") + ylab("Mean response")

totalbeaths <- h2o.partialPlot(object = a, data = train, cols = c("totalbeaths"))
p6 <- qplot(totalbeaths$totalbeaths, totalbeaths$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.35) + xlab("Total battle deaths") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)

## COW data
df.cowa <- as.h2o(df.cow)

df.cowa$MKstart <- as.factor(df.cowa$MKstart)
h2o.levels(df.cowa$MKstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.cowa,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)

train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "MKstart"
x <- setdiff(names(df.ucdp), c(y, "ccode", "year", "rgdppc",
                               "mksyr2", "mksyr3", "sf", "country",
                               "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf03",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf03",
                       sort_by = "auc",
```

```
                        decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

## Graphs
a <- h2o.loadModel("/root/Documents/mk/gridrf03_model_3")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))
par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Total battle deaths",
                      "Excluded population",
                      "Yrs since mass killing",
                      "Log GDP per capita",
                      "Ethnic polarisation",
                      "Physical integrity"),
        main = "")

physint <- h2o.partialPlot(object = a, data = train, cols = c("physint"))
p1 <- qplot(physint$physint, physint$mean_response) + geom_line() + theme_classic() +
        xlab("Physical integrity") +  ylab("Mean response")

egiptpolrqnew <- h2o.partialPlot(object = a, data = train, cols = c("egiptpolrqnew"))
p2 <- qplot(egiptpolrqnew$egiptpolrqnew, egiptpolrqnew$mean_response) + geom_line() +
        theme_classic() + xlab("Ethnic polarisation") + ylab("Mean response")

logrgdppc <- h2o.partialPlot(object = a, data = train, cols = c("logrgdppc"))
p3 <- qplot(logrgdppc$logrgdppc, logrgdppc$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.1) + xlab("Log GDP per capita") + ylab("Mean response")

mksyr <- h2o.partialPlot(object = a, data = train, cols = c("mksyr"))
p4 <- qplot(mksyr$mksyr, mksyr$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.1) + xlab("Years since mass killing") +  ylab("Mean response")

exclpop <- h2o.partialPlot(object = a, data = train, cols = c("exclpop"))
p5 <- qplot(exclpop$exclpop, exclpop$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.1) + xlab("Excluded population") + ylab("Mean response")

totalbeaths <- h2o.partialPlot(object = a, data = train, cols = c("totalbeaths"))
p6 <- qplot(totalbeaths$totalbeaths, totalbeaths$mean_response) + geom_line() + theme_classic() +
        ylim(0, 0.1) + xlab("Total battle deaths") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)

## Ethnic war
df.etha <- as.h2o(df.eth)

df.etha$MKstart <- as.factor(df.etha$MKstart)
h2o.levels(df.etha$MKstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.etha,
                         ratios = c(0.7, 0.15),
                         seed = 42)

train <- h2o.assign(splits[[1]], "train.hex")
```

```
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "MKstart"
x <- setdiff(names(df.eth), c(y, "ccode", "year", "rgdppc",
                              "mksyr2", "mksyr3", "sf", "country",
                              "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf04",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf04",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

# Graphs
a <- h2o.loadModel("/root/Documents/mk/gridrf04_model_14")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Military expenditure",
                      "Total trade",
                      "% Urban",
                      "Trade dependence",
                      "Military personnel",
                      "CINC"),
        main = "")


cinc <- h2o.partialPlot(object = a, data = train, cols = c("cinc"))
p1 <- qplot(cinc$cinc, cinc$mean_response) + geom_line() + theme_classic() +
        xlab("CINC") +  ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
```

```
p2 <- qplot(milper$milper, milper$mean_response) + geom_line() +
        theme_classic() + xlab("Military personnel") + ylab("Mean response")

tradedependence <- h2o.partialPlot(object = a, data = train, cols = c("tradedependence"))
p3 <- qplot(tradedependence$tradedependence, tradedependence$mean_response) + geom_line() + theme_c
        xlab("Trade dependence") + ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p4 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
       theme_classic() + xlab("% Urban") + ylab("Mean response")

totaltrade <- h2o.partialPlot(object = a, data = train, cols = c("totaltrade"))
p5 <- qplot(totaltrade$totaltrade, totaltrade$mean_response) + geom_line() +
        theme_classic() + xlab("Total trade") + ylab("Mean response")

milex <- h2o.partialPlot(object = a, data = train, cols = c("milex"))
p6 <- qplot(milex$milex, milex$mean_response) + geom_line() +
        theme_classic() + xlab("Military expenditure") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)

####################################
### Genocide/Politicide Variable ###
####################################

## The code below replicates the same analyses presented above
## but using a measure of genocide/politicide coded by Harff (2003).

## Data wrangling
df3 <- haven::read_dta("data/uamkstart.dta") %>% setDT()

sd.cols <- c("UCDPcivilwarstart", "UCDPcivilwarongoing", "COWcivilwarstart",
             "COWcivilwarongoing", "ethnowarstart", "ethnowarongoing",
             "assdummy", "demdummy", "elf", "lmtnest", "pop", "realgdp",
             "rgdppc", "polity2", "exclpop", "discpop", "polrqnew",
             "poltrqnew", "egiptpolrqnew", "egippolrqnew", "discrim",
             "elf2", "interstatewar", "milex", "milper", "percentpopurban",
             "postcoldwar", "coupdummy", "riotdummy", "territoryaims",
             "totaltrade", "tradedependence", "militias", "physint", "cinc",
             "totalbeaths", "change", "guerrilladummy", "sf", "regtrans")

df4 <- cbind(df3, df3[, shift(.SD, 1, give.names = TRUE),
                      by = ccode, .SDcols = sd.cols])

# Remove the second `ccode` variable
df4 <- as.data.frame(df4[, -c(75)])

# Add new variables
df4$logrgdppc_lag_1 <- log(df4$rgdppc_lag_1)
df4$polity2sq_lag_1 <- df4$polity2_lag_1^2

# Renaming variables
df5 <- as.data.frame(df4[, c(1:4, 72:116)])
names(df5) <- sub("_.*","", names(df5))

################################################
### Distributed Random Forests - Harff (2003) ###
################################################

df5a <- as.h2o(df5)
```

```
df5a$uamkstart <- as.factor(df5a$uamkstart)  #encode the binary repsonse as a factor
h2o.levels(df5a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df5a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
x <- setdiff(names(df5), c(y, "ccode", "year", "rgdppc",
                           "uamkyr2", "uamkyr3", "sf", "country",
                           "elf2", "polity2sq"))
# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5,
               grid_id = "gridrf05",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

# Saving the most accurate model
rf.grid <- h2o.getGrid(grid_id = "gridrf05",
                       sort_by = "auc",
                       decreasing = TRUE)

rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

## Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf05_model_27")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Real GDP",
```

```
                      "Military expenditure",
                      "Total trade",
                      "Military personnel",
                      "CINC",
                      "% Urban"),
        main = "")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p1 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
        theme_classic() + xlab("% Urban") + ylab("Mean response")

cinc <- h2o.partialPlot(object = a, data = train, cols = c("cinc"))
p2 <- qplot(cinc$cinc, cinc$mean_response) + geom_line() + theme_classic() +
        xlab("CINC") +  ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
p3 <- qplot(milper$milper, milper$mean_response) + geom_line() +
        theme_classic() + xlab("Military personnel") + ylab("Mean response")

totaltrade <- h2o.partialPlot(object = a, data = train, cols = c("totaltrade"))
p4 <- qplot(totaltrade$totaltrade, totaltrade$mean_response) + geom_line() + theme_classic() +
        xlab("Total trade") + ylab("Mean response")

milex <- h2o.partialPlot(object = a, data = train, cols = c("milex"))
p5 <- qplot(milex$milex, milex$mean_response) + geom_line() +
        theme_classic() + xlab("Military expenditure") + ylab("Mean response")

realgdp <- h2o.partialPlot(object = a, data = train, cols = c("realgdp"))
p6 <- qplot(realgdp$realgdp, realgdp$mean_response) + geom_line() +
        theme_classic() + xlab("Real GDP") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)


## UCDP == 1
df.ucdp2 <- df5 %>% filter(UCDPcivilwarongoing == 1)
df.ucdp2a <- as.h2o(df.ucdp2)

df.ucdp2a$uamkstart <- as.factor(df.ucdp2a$uamkstart)  #encode the binary repsonse as a factor
h2o.levels(df.ucdp2a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.ucdp2a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
x <- setdiff(names(df.ucdp2), c(y, "ccode", "year", "rgdppc",
                           "uamkyr2", "uamkyr3", "sf", "country",
                           "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf06",
               fold_assignment = "Stratified",
```

```
                hyper_params = list(ntrees = c(256, 512, 1024),
                                    max_depth = c(10, 20, 40),
                                    mtries = c(5, 6, 7),
                                    balance_classes = c(TRUE, FALSE),
                                    sample_rate = c(0.5, 0.632, 0.95),
                                    col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                    histogram_type = c("UniformAdaptive",
                                                       "Random",
                                                       "QuantilesGlobal",
                                                       "RoundRobin")),
                search_criteria = list(strategy = "RandomDiscrete",
                                       max_models = 1000,
                                       stopping_metric = "auc",
                                       stopping_tolerance = 0.01,
                                       stopping_rounds = 5,
                                       seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf06",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

## Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf06_model_43")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Excluded population",
                      "% Urban pop.",
                      "Trade dependence",
                      "Log GDP per capita",
                      "Years since genocide",
                      "Military personnel"),
        main = "")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
p1 <- qplot(milper$milper, milper$mean_response) + geom_line() +
        theme_classic() + xlab("Military personnel") + ylab("Mean response")

uamkyr <- h2o.partialPlot(object = a, data = train, cols = c("uamkyr"))
p2 <- qplot(uamkyr$uamkyr, uamkyr$mean_response) + geom_line() + theme_classic() +
        xlab("Years since genocide") +  ylab("Mean response")

logrgdppc <- h2o.partialPlot(object = a, data = train, cols = c("logrgdppc"))
p3 <- qplot(logrgdppc$logrgdppc, logrgdppc$mean_response) + geom_line() + theme_classic() +
        xlab("Log GDP per capita") + ylab("Mean response")

tradedependence <- h2o.partialPlot(object = a, data = train, cols = c("tradedependence"))
p4 <- qplot(tradedependence$tradedependence, tradedependence$mean_response) + geom_line() +
        theme_classic() + xlab("Trade dependence") + ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p5 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
```

```
        theme_classic() + xlab("% Urban") + ylab("Mean response")

exclpop <- h2o.partialPlot(object = a, data = train, cols = c("exclpop"))
p6 <- qplot(exclpop$exclpop, exclpop$mean_response) + geom_line() +
        theme_classic() + xlab("Excluded population") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)

## COW == 1
df.cow2 <- df5 %>% filter(COWcivilwarongoing == 1)

df.cow2a$uamkstart <- as.factor(df.cow2a$uamkstart) #encode the binary repsonse as a factor
h2o.levels(df.cow2a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.cow2a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
x <- setdiff(names(df.cow2), c(y, "ccode", "year", "rgdppc",
                               "uamkyr2", "uamkyr3", "sf", "country",
                               "elf2", "polity2sq"))

## Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf07",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf07",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)
```

```
## Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf07_model_27")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Military expenditure",
                      "% Urban pop.",
                      "Total trade",
                      "Years since genocide",
                      "Military personnel",
                      "Trade dependence"),
        main = "")

uamkyr <- h2o.partialPlot(object = a, data = train, cols = c("uamkyr"))
p3 <- qplot(uamkyr$uamkyr, uamkyr$mean_response) + geom_line() + theme_classic() +
        xlab("Years since genocide") +  ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
p2 <- qplot(milper$milper, milper$mean_response) + geom_line() +
        theme_classic() + xlab("Military personnel") + ylab("Mean response")

totaltrade <- h2o.partialPlot(object = a, data = train, cols = c("totaltrade"))
p4 <- qplot(totaltrade$totaltrade, totaltrade$mean_response) + geom_line() + theme_classic() +
        xlab("Total trade") + ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p5 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
        theme_classic() + xlab("% Urban") + ylab("Mean response")

tradedependence <- h2o.partialPlot(object = a, data = train, cols = c("tradedependence"))
p1 <- qplot(tradedependence$tradedependence, tradedependence$mean_response) + geom_line() +
        theme_classic() + xlab("Trade dependence") + ylab("Mean response")

milex <- h2o.partialPlot(object = a, data = train, cols = c("milex"))
p6 <- qplot(milex$milex, milex$mean_response) + geom_line() +
        theme_classic() + xlab("Military expenditure") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)

## ETHONSET == 1
df.eth2 <- df5 %>% filter(ethnowarongoing == 1)

df.eth2a <- as.h2o(df.eth2)

df.eth2a$uamkstart <- as.factor(df.eth2a$uamkstart)  #encode the binary repsonse as a factor
h2o.levels(df.eth2a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.eth2a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
```

```
x <- setdiff(names(df.eth2), c(y, "ccode", "year", "rgdppc",
                               "uamkyr2", "uamkyr3", "sf", "country",
                               "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf08",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 1000,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf08",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

## Graphs
a <- h2o.loadModel("/home/sussa/Documents/GitHub/mk/gridrf08_model_55")
print(va <- a %>% h2o.varimp() %>% as.data.frame() %>% head(., 6))

par(mgp=c(2.2,0.45,0), tcl=-0.4, mar=c(2,7.5,1,1))
barplot(va$scaled_importance[6:1],
        horiz = TRUE, las = 1, cex.names=0.9,
        names.arg = c("Total battle deaths",
                      "Military personnel",
                      "Log GDP per capita",
                      "CINC",
                      "% Urban",
                      "Military expenditure"),
        main = "")

cinc <- h2o.partialPlot(object = a, data = train, cols = c("cinc"))
p3 <- qplot(cinc$cinc, cinc$mean_response) + geom_line() + theme_classic() +
        xlab("CINC") +  ylab("Mean response")

milper <- h2o.partialPlot(object = a, data = train, cols = c("milper"))
p5 <- qplot(milper$milper, milper$mean_response) + geom_line() +
        theme_classic() + xlab("Military personnel") + ylab("Mean response")

logrgdppc <- h2o.partialPlot(object = a, data = train, cols = c("logrgdppc"))
p4 <- qplot(logrgdppc$logrgdppc, logrgdppc$mean_response) + geom_line() + theme_classic() +
```

```
        xlab("Log GDP per capita") + ylab("Mean response")

percentpopurban <- h2o.partialPlot(object = a, data = train, cols = c("percentpopurban"))
p2 <- qplot(percentpopurban$percentpopurban, percentpopurban$mean_response) + geom_line() +
        theme_classic() + xlab("% Urban") + ylab("Mean response")

totalbeaths <- h2o.partialPlot(object = a, data = train, cols = c("totalbeaths"))
p6 <- qplot(totalbeaths$totalbeaths, totalbeaths$mean_response) + geom_line() +
        theme_classic() + xlab("Total battle deaths") + ylab("Mean response")

milex <- h2o.partialPlot(object = a, data = train, cols = c("milex"))
p1 <- qplot(milex$milex, milex$mean_response) + geom_line() +
        theme_classic() + xlab("Military expenditure") + ylab("Mean response")

multiplot(p1, p4, p2, p5, p3, p6, cols = 3)


##############################################
### Genocides and Politicides (Harff 2003) ###
##############################################

# Preparing the dataset
df3 <- haven::read_dta("data/uamkstart.dta") %>% setDT()
sd.cols <- c("UCDPcivilwarstart", "UCDPcivilwarongoing", "COWcivilwarstart",
             "COWcivilwarongoing", "ethnowarstart", "ethnowarongoing",
             "assdummy", "demdummy", "elf", "lmtnest", "pop", "realgdp",
             "rgdppc", "polity2", "exclpop", "discpop", "polrqnew",
             "poltrqnew", "egiptpolrqnew", "egippolrqnew", "discrim",
             "elf2", "interstatewar", "milex", "milper", "percentpopurban",
             "postcoldwar", "coupdummy", "riotdummy", "territoryaims",
             "totaltrade", "tradedependence", "militias", "physint", "cinc",
             "totalbeaths", "change", "guerrilladummy", "sf", "regtrans")

df4 <- cbind(df3, df3[, shift(.SD, 1, give.names = TRUE),
                    by = ccode, .SDcols = sd.cols])

# Remove the second `ccode` variable
df4 <- as.data.frame(df4[, -c(75)])

# Add new variables
df4$logrgdppc_lag_1 <- log(df4$rgdppc_lag_1)
df4$polity2sq_lag_1 <- df4$polity2_lag_1^2

# Renaming variables
df5 <- as.data.frame(df4[, c(1:4, 72:116)])
names(df5) <- sub("_.*","", names(df5))

## Extreme Bounds
free.variables <- c("logrgdppc", "polity2", "uamkyr")
civilwar.variables <- c("UCDPcivilwarongoing", "UCDPcivilwarstart",
                        "COWcivilwarongoing", "COWcivilwarstart",
                        "ethnowarongoing", "ethnowarstart")
doubtful.variables <- c("UCDPcivilwarongoing", "UCDPcivilwarstart",
                        "COWcivilwarongoing", "COWcivilwarstart",
                        "ethnowarongoing", "ethnowarstart", "assdummy",
                        "totaltrade", "tradedependence", "milper", "milex",
                        "pop", "totalbeaths", "guerrilladummy", "regtrans",
                        "riotdummy", "territoryaims", "militias",
                        "physint", "percentpopurban", "coupdummy",
                        "postcoldwar",  "lmtnest", "realgdp", "discrim",
```

```
                        "exclpop", "discpop", "elf",  "polrqnew",
                        "egippolrqnew", "poltrqnew", "egiptpolrqnew",
                        "polity2sq")
m1 <- eba(y = "uamkstart", free = free.variables,
          exclusive = list(civilwar.variables),
          doubtful = doubtful.variables, k = 0:4,
          data = df5, vif = 7, level = 0.9,
          se.fun = se.clustered.robust)

hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "uamkyr",
                       "UCDPcivilwarongoing",
                       "UCDPcivilwarstart", "COWcivilwarongoing",
                       "COWcivilwarstart", "ethnowarongoing", "ethnowarstart",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
                       "riotdummy", "territoryaims", "militias", "physint",
                       "percentpopurban", "coupdummy", "postcoldwar",
                       "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                       "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                       "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last genocide",
              "UCDP ongoing", "UCDP onset", "COW ongoing", "COW onset",
              "Ethnic ongoing", "Ethnic onset", "Assassination", "Total trade",
         "Trade dependence", "Military personnel", "Military expenditure", "Population",
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Group/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

### Ongoing Civil Wars

# UCDPcivilwarongoing == 1
df.ucdp2 <- df5 %>% filter(UCDPcivilwarongoing == 1)
doubtful.variables <- c("assdummy", "totaltrade", "tradedependence",
                        "milper", "milex", "pop", "totalbeaths",
                        "guerrilladummy", "regtrans", "riotdummy",
                        "territoryaims", "militias", "physint",
                        "percentpopurban", "coupdummy", "postcoldwar",
                        "lmtnest", "realgdp", "discrim", "exclpop",
                        "discpop", "elf",  "polrqnew", "egippolrqnew",
                        "poltrqnew", "egiptpolrqnew", "polity2sq")

m1 <- eba(y = "uamkstart", free = free.variables,
          doubtful = doubtful.variables, k = 0:4,
          data = df.ucdp2, vif = 7, draws = 50000,
          level = 0.9, se.fun = se.clustered.robust)

hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "uamkyr",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
                       "riotdummy", "territoryaims", "militias", "physint",
                       "percentpopurban", "coupdummy", "postcoldwar",
                       "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                       "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                       "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last genocide",
              "Assassination", "Total trade",
         "Trade dependence", "Military personnel", "Military expenditure", "Population",
```

```
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Groups/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

# COWcivilwarongoing == 1
df.cow2 <- df5 %>% filter(COWcivilwarongoing == 1)
doubtful.variables <- c("assdummy", "totaltrade", "tradedependence",
                        "milper", "milex", "pop", "totalbeaths",
                        "guerrilladummy", "regtrans", "riotdummy",
                        "territoryaims", "militias", "physint",
                        "percentpopurban", "coupdummy", "postcoldwar",
                        "lmtnest", "realgdp", "discrim", "exclpop",
                        "discpop", "elf",  "polrqnew", "egippolrqnew",
                        "poltrqnew", "egiptpolrqnew", "polity2sq")

m1 <- eba(y = "uamkstart", free = free.variables,
          doubtful = doubtful.variables, k = 0:4,
          data = df.cow2, vif = 7, draws = 50000,
          level = 0.9, se.fun = se.clustered.robust)

hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "uamkyr",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
                       "riotdummy", "territoryaims", "militias", "physint",
                       "percentpopurban", "coupdummy", "postcoldwar",
                       "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                       "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                       "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last genocide",
              "Assassination", "Total trade",
         "Trade dependence", "Military personnel", "Military expenditure", "Population",
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Groups/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

# ethnic civil war == 1
df.eth2 <- df5 %>% filter(ethnowarongoing == 1)
doubtful.variables <- c("assdummy", "totaltrade", "tradedependence",
                        "milper", "milex", "pop", "totalbeaths",
                        "guerrilladummy", "regtrans", "riotdummy",
                        "territoryaims", "militias", "physint",
                        "percentpopurban", "coupdummy", "postcoldwar",
                        "lmtnest", "realgdp", "discrim", "exclpop",
                        "discpop", "elf",  "polrqnew", "egippolrqnew",
                        "poltrqnew", "egiptpolrqnew", "polity2sq")

m1 <- eba(y = "uamkstart", free = free.variables,
          doubtful = doubtful.variables, k = 0:4,
          data = df.eth2, vif = 7, draws = 50000,
          level = 0.9, se.fun = se.clustered.robust)

hist(m1, variables = c("logrgdppc", "polity2", "polity2sq", "uamkyr",
                       "assdummy", "totaltrade", "tradedependence", "milper",
                       "milex","pop", "totalbeaths", "guerrilladummy", "regtrans",
```

```
                        "riotdummy", "territoryaims", "militias", "physint",
                        "percentpopurban", "coupdummy", "postcoldwar",
                        "lmtnest", "realgdp", "discrim", "exclpop", "discpop",
                        "elf", "polrqnew", "egippolrqnew", "poltrqnew",
                        "egiptpolrqnew"),
     main = c("Log GDP capita", "Polity IV", "Polity IV^2", "Years last genocide",
              "Assassination", "Total trade",
          "Trade dependence", "Military personnel", "Military expenditure", "Population",
              "Total deaths", "Guerrilla", "Regime transition", "Riots",
              "Territory Aims", "Militias", "Physical integrity", "% Urban",
              "Coups", "Post-Cold War", "Mountainous terrain", "Real GDP",
             "Discrimination", "Excl pop", "Discrim pop", "ELF", "Groups/Eth relevant",
              "Groups/Tot pop", "Inc groups/Eth relevant", "Inc groups/Tot pop"),
     density.col = "black", mu.col = "red3")

#########################################################
### Genocide/Politicide -- Distributed Random Forest ###
#########################################################

df5a <- as.h2o(df5)

df5a$uamkstart <- as.factor(df5a$uamkstart)  #encode the binary repsonse as a factor
h2o.levels(df5a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df5a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
x <- setdiff(names(df5), c(y, "ccode", "year", "rgdppc",
                           "uamkyr2", "uamkyr3", "sf", "country",
                           "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5,
               grid_id = "gridrf05",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 500,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))
```

```
# Saving the most accurate model
rf.grid <- h2o.getGrid(grid_id = "gridrf05",
                       sort_by = "auc",
                       decreasing = TRUE)

rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

## UCDP == 1
df.ucdp2a <- as.h2o(df.ucdp2)

df.ucdp2a$uamkstart <- as.factor(df.ucdp2a$uamkstart)  #encode the binary repsonse as a factor
h2o.levels(df.ucdp2a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.ucdp2a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
x <- setdiff(names(df.ucdp2), c(y, "ccode", "year", "rgdppc",
                           "uamkyr2", "uamkyr3", "sf", "country",
                           "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf06",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 100,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf06",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
```

```
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

## COW == 1
df.cow2a <- as.h2o(df.cow2)

df.cow2a$uamkstart <- as.factor(df.cow2a$uamkstart) #encode the binary repsonse as a factor
h2o.levels(df.cow2a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.cow2a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
x <- setdiff(names(df.cow2), c(y, "ccode", "year", "rgdppc",
                           "uamkyr2", "uamkyr3", "sf", "country",
                           "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf07",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 100,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf07",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)

## ETHONSET == 1
df.eth2a <- as.h2o(df.eth2)
```

```
df.eth2a$uamkstart <- as.factor(df.eth2a$uamkstart) #encode the binary repsonse as a factor
h2o.levels(df.eth2a$uamkstart)

# Partition the data into training, validation and test sets
splits <- h2o.splitFrame(data = df.eth2a,
                         ratios = c(0.7, 0.15),  # 70%, 15%, 15%
                         seed = 42)  # reproducibility


train <- h2o.assign(splits[[1]], "train.hex")
valid <- h2o.assign(splits[[2]], "valid.hex")
test <- h2o.assign(splits[[3]], "test.hex")

y <- "uamkstart"
x <- setdiff(names(df.eth2), c(y, "ccode", "year", "rgdppc",
                           "uamkyr2", "uamkyr3", "sf", "country",
                           "elf2", "polity2sq"))

# Running the model
rf <- h2o.grid("randomForest", x = x, y = y, training_frame = train,
               validation_frame = valid, nfolds = 5, grid_id = "gridrf08",
               fold_assignment = "Stratified",
               hyper_params = list(ntrees = c(256, 512, 1024),
                                   max_depth = c(10, 20, 40),
                                   mtries = c(5, 6, 7),
                                   balance_classes = c(TRUE, FALSE),
                                   sample_rate = c(0.5, 0.632, 0.95),
                                   col_sample_rate_per_tree = c(0.5, 0.9, 1.0),
                                   histogram_type = c("UniformAdaptive",
                                                      "Random",
                                                      "QuantilesGlobal",
                                                      "RoundRobin")),
               search_criteria = list(strategy = "RandomDiscrete",
                                      max_models = 100,
                                      stopping_metric = "auc",
                                      stopping_tolerance = 0.01,
                                      stopping_rounds = 5,
                                      seed = 26227709))

rf.grid <- h2o.getGrid(grid_id = "gridrf08",
                       sort_by = "auc",
                       decreasing = TRUE)
rf2 <- h2o.getModel(rf.grid@model_ids[[1]])
h2o.saveModel(rf2, path = "/root/Documents/mk/")
summary(rf2)
varimp <- as.data.frame(h2o.varimp(rf2))
h2o.varimp_plot(rf2)
h2o.performance(rf2, newdata = test)
```

References

Allansson, M., Melander, E., and Themnér, L. (2017). Organized violence, 1989–2016. *Journal of Peace Research*, 54(4):574–587.

Anderton, C. H. and Carter, J. R. (2015). A new look at weak state conditions and genocide risk. *Peace Economics, Peace Science and Public Policy*, 21(1):1–36.

Balcells, L. (2010). Rivalry and revenge: Violence against civilians in conventional civil wars. *International Studies Quarterly*, 54(2):291–313.

Balcells, L. (2011). Continuation of politics by two means: Direct and indirect violence in civil war. *Journal of Conflict Resolution*, 55(3):397–422.

Balcells, L. and Kalyvas, S. N. (2014). Does warfare matter? severity, duration, and outcomes of civil wars. *Journal of Conflict Resolution*, 58(8):1390–1418.

Banks, A. S. (1999). *Cross-National Time-Series Data Archive User's Manual*. Center for Social Analysis, State University of New York at Binghamton.

Besançon, M. L. (2005). Relative resources: Inequality in ethnic wars, revolutions, and genocides. *Journal of Peace Research*, 42(4):393–415.

Bulutgil, H. Z. (2015). Social cleavages, wartime experience, and ethnic cleansing in europe. *Journal of Peace Research*, 52(5):577–590.

Bundervoet, T. (2009). Livestock, land and political power: The 1993 killings in burundi. *Journal of Peace Research*, 46(3):357–376.

Carey, S. C., Mitchell, N. J., and Lowe, W. (2013). States, the security sector, and the monopoly of violence: A new database on pro-government militias. *Journal of Peace Research*, 50(2):249–258.

Cederman, L.-E., Wimmer, A., and Min, B. (2010). Why do ethnic groups rebel? new data and analysis. *World Politics*, 62(1):87–119.

Cingranelli, D. L. and Richards, D. L. (2010). The cingranelli and richards (ciri) human rights data project. *Human Rights Quarterly*, 32(2):401–424.

Clayton, G. and Thomson, A. (2016). Civilianizing civil conflict: Civilian defense militias and the logic of violence in intrastate conflict. *International Studies Perspectives*, 60(3):499–510.

Colaresi, M. and Carey, S. (2008). To kill or to protect: Security forces, domestic institutions, and genocide. *Journal of Conflict Resolution*, 52(1):39–67.

Downes, A. B. (2006). Desperate times, desperate measures: The causes of civilian victimization in war. *International Security*, 30(4):152–195.

Downes, A. B. (2007). Restraint or propellant? democracy and civilian fatalities in interstate wars. *Journal of Conflict Resolution*, 51(6):872–904.

Easterly, W., Gatti, R., and Kurlat, S. (2006). Development, democracy, and mass killings. *Journal of Economic Growth*, 11(2):129–156.

Eck, K. and Hultman, L. (2007). One-sided violence against civilians in war: Insights from new fatality data. *Journal of Peace Research*, 44(2):233–246.

Esteban, J., Morelli, M., and Rohner, D. (2015). Strategic mass killings. *Journal of Political Economy*, 123(5):1087–1132.

Fazal, T. M. and Greene, B. C. (2015). A particular difference: European identity and civilian targeting. *British Journal of Political Science*, 45(4):829–851.

Fearon, J. D. and Laitin, D. D. (2003). Ethnicity, Insurgency, and Civil War. *American Political Science Review*, 97(01):75–90.

Fjelde, H. and Hultman, L. (2014). Weakening the enemy: A disaggregated study of violence against civilians in africa. *Journal of Conflict Resolution*, 58(7):1230–1257.

Gelman, A., Jakulin, A., Pittau, M. G., and Su, Y.-S. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, pages 1360–1383.

Gelman, A. and Su, Y.-S. (2016). *arm: Data Analysis Using Regression and Multilevel/Hierarchical Models*. R package version 1.9-3.

Gleditsch, K. S. (2002). Expanded trade and gdp data. *Journal of Conflict Resolution*, 46(5):712–724.

Gleditsch, N. P., Wallensteen, P., Eriksson, M., Sollenberg, M., and Strand, H. (2002). Armed conflict 1946–2001: A new dataset. *Journal of peace research*, 39(5):615–637.

Goldsmith, B. E., Butcher, C. R., Semenovich, D., and Sowmya, A. (2013). Forecasting the onset of genocide and politicide: Annual out-of-sample forecasts on a global dataset, 1988–2003. *Journal of Peace Research*, 50(4):437–452.

Harff, B. (2003). No lessons learned from the holocaust? assessing risks of genocide and political mass murder since 1955. *American Political Science Review*, 97(1):57–73.

Joshi, M. and Quinn, J. M. (2017). Who kills whom? the micro-dynamics of civilian targeting in civil war. *Social Science Research*, 63:227–241.

Kim, D. (2010). What makes state leaders brutal? examining grievances and mass killing during civil war. *Civil Wars*, 12(3):237–260.

Kim, N. K. (2016). Revolutionary leaders and mass killing. *Journal of Conflict Resolution*, page 0022002716653658.

Kisangani, E. and Wayne Nafziger, E. (2007). The political economy of state terror. *Defence and Peace Economics*, 18(5):405–414.

Koren, O. (2017). Means to an end: Pro-government militias as a predictive indicator of strategic mass killing. *Conflict Management and Peace Science*, 34(5):461–484.

Krain, M. (1997). State-Sponsored Mass Murder: The Onset and Severity of Genocides and Politicides. *Journal of Conflict Resolution*, 41(3):331–360.

Lacina, B. and Gleditsch, N. P. (2005). Monitoring trends in global combat: A new dataset of battle deaths. *European Journal of Population/Revue Européenne de Démographie*, 21(2-3):145–166.

Manekin, D. (2013). Violence against civilians in the second intifada: The moderating effect of armed group structure on opportunistic violence. *Comparative Political Studies*, 46(10):1273–1300.

Marshall, M. G., Gurr, T. R., and Harff, B. (2017). Pitf state failure problem set, 1955-2016.

McDoom, O. S. (2013). Who killed in rwanda's genocide? micro-space, social influence and individual participation in intergroup violence. *Journal of Peace Research*, 50(4):453–467.

McDoom, O. S. (2014). Predicting violence within genocide: A model of elite competition and ethnic segregation from rwanda. *Political Geography*, 42:34–45.

Melander, E., Öberg, M., and Hall, J. (2009). Are 'new wars' more atrocious? battle severity, civilians killed and forced migration before and after the end of the cold war. *European Journal of International Relations*, 15(3):505–536.

Montalvo, J. G. and Reynal-Querol, M. (2008). Discrete polarisation with an application to the determinants of genocides. *The Economic Journal*, 118(533):1835–1865.

O'Brien, R. M. (2007). A caution regarding rules of thumb for variance inflation factors. *Quality & Quantity*, 41(5):673–690.

Pilster, U., Böhmelt, T., and Tago, A. (2016). The differentiation of security forces and the onset of genocidal violence. *Armed Forces & Society*, 42(1):26–50.

Querido, C. M. (2009). State-sponsored mass killing in african wars–greed or grievance? *International Advances in Economic Research*, 15(3):351.

Raleigh, C. (2012). Violence against civilians: A disaggregated analysis. *International Interactions*, 38(4):462–481.

Rost, N. (2013). Will it happen again? on the possibility of forecasting the risk of genocide. *Journal of Genocide Research*, 15(1):41–67.

Rummel, R. J. (1995). Democracy, power, genocide, and mass murder. *Journal of Conflict Resolution*, 39(1):3–26.

Sala-i-Martin, X., Doppelhofer, G., and Miller, R. I. (2004). Determinants of long-term growth: A bayesian averaging of classical estimates (bace) approach. *American Economic Review*, 94(4):813–835.

Sarkees, M. R. and Wayman, F. W. (2010). *Resort to War*. Washington DC: CQ Press.

Schneider, G. and Bussmann, M. (2013). Accounting for the dynamics of one-sided violence: Introducing kosved. *Journal of Peace Research*, 50(5):635–644.

Singer, J. D. (1988). Reconstructing the correlates of war dataset on material capabilities of states, 1816–1985. *International Interactions*, 14(2):115–132.

Singer, J. D., Bremer, S., and Stuckey, J. (1972). Capability distribution, uncertainty, and major power war, 1820-1965. In Russett, B., editor, *Peace, War, and Numbers*, pages 19–48. Beverly Hills: Sage.

Siroky, D. and Dzutsati, V. (2015). The empire strikes back: Ethnicity, terrain, and indiscriminate violence in counterinsurgencies. *Social Science Quarterly*, 96(3):807–829.

Stanton, J. (2015). Regulating militias: Governments, militias, and civilian targeting in civil war. *Journal of Conflict Resolution*, 59(5):899–923.

Sullivan, C. M. (2012). Blood in the village: A local-level investigation of state massacres. *Conflict Management and Peace Science*, 29(4):373–396.

The H2O.ai Team (2017). *h2o: R Interface for H2O*. R package version 3.14.0.3.

Tir, J. and Jasinski, M. (2008). Domestic-level diversionary theory of war: Targeting ethnic minorities. *Journal of Conflict Resolution*, 52(5):641–664.

Ulfelder, J. (2012). Forecasting onsets of mass killing. Technical report. Accessed: January 2018.

Ulfelder, J. and Valentino, B. (2008). Assessing risks of state-sponsored mass killing. Technical report. Accessed: January 2018.

Uzonyi, G. (2015). Civil war victory and the onset of genocide and politicide. *International Interactions*, 41(2):365–391.

Uzonyi, G. (2016). Domestic unrest, genocide and politicide. *Political Studies*, 64(2):315–334.

Valentino, B., Huth, P., and Balch-Lindsay, D. (2004). "draining the sea": Mass killing and guerrilla warfare. *International Organization*, 58(2):375–407.

Valentino, B., Huth, P., and Croco, S. (2006). Covenants without the sword international law and the protection of civilians in times of war. *World Politics*, 58(3):339–377.

Verpoorten, M. (2012). Leave none to claim the land: A malthusian catastrophe in rwanda? *Journal of Peace Research*, 49(4):547–563.

Wayman, F. W. and Tago, A. (2010). Explaining the onset of mass killing, 1949–87. *Journal of Peace Research*, 47(1):3–13.

Wig, T. and Tollefsen, A. F. (2016). Local institutional quality and conflict violence in africa. *Political geography*, 53:30–42.

Yanagizawa-Drott, D. (2014). Propaganda and conflict: Evidence from the rwandan genocide. *The Quarterly Journal of Economics*, 129(4):1947–1994.