QM 2021 Week 13: Wrap Up

Oliver Rittmann Viktoriia Semenova David Grundmanns

December 2 | 6 | 7, 2021

Contents

Today we will learn:

1. More on Interactions in GLMs
2. Robustness Tests
3. Formatting and RMarkdown (Revision)

In other words, the goals are to:

- do simulations with count models & interactions over the range of values
- think critically about the quantities of interest we produce
- explore the commonly used robustness checks
- review how to do citations and use chunk options in RMarkdown

---

```
# The first line sets an option for the final document that can be produced from
# the .Rmd file. Don't worry about it.
knitr::opts_chunk$set(echo = TRUE,
                      collapse = TRUE,
                      out.width="\\textwidth", # for larger figures
                      attr.output = 'style="max-height: 200px"',
                      tidy = 'styler' # styles the code in the output
                      )

# The next bit is quite powerful and useful.
# First you define which packages you need for your analysis and assign it to
# the p_needed object.
p_needed <-
  c("ggplot2", "viridis", "MASS", "optimx", "scales", "foreign",
    "patchwork", "stargazer", "janitor")

# Now you check which packages are already installed on your computer.
# The function installed.packages() returns a vector with all the installed
# packages.
packages <- rownames(installed.packages())
# Then you check which of the packages you need are not installed on your
# computer yet. Essentially you compare the vector p_needed with the vector
# packages. The result of this comparison is assigned to p_to_install.
p_to_install <- p_needed[!(p_needed %in% packages)]
# If at least one element is in p_to_install you then install those missing
# packages.
if (length(p_to_install) > 0) {
  install.packages(p_to_install, repos = "http://cran.us.r-project.org")
}
# installation from a different source
if ("countreg" %in% p_to_install) {
  install.packages("countreg", repos = "http://R-Forge.R-project.org")
}
# Now that all packages are installed on the computer, you can load them for
# this project. Additionally the expression returns whether the packages were
# successfully loaded.
sapply(p_needed, require, character.only = TRUE)

# This is an option for stargazer tables
# It automatically adapts the output to html or latex,
# depending on whether we want a html or pdf file
stargazer_opt <- ifelse(knitr::is_latex_output(), "latex", "html")

# Don't worry about this part: it ensures that if the file is knitted to html,
```

```r
# significance notes are depicted correctly
if (stargazer_opt == "html"){
  fargs <- formals(stargazer)
  fargs$notes.append = FALSE
  fargs$notes = c("<em>&#42;p&lt;0.1;&#42;&#42;p&lt;0.05;&#42;&#42;&#42;p&lt;0.01</em>")
  formals(stargazer) <- fargs
}

# only relevant for ggplot2 plotting
# setting a global ggplot theme for the entire document to avoid
# setting this individually for each plot
theme_set(theme_classic() + # start with classic theme
  theme(
    plot.background = element_blank(),# remove all background
    plot.title.position = "plot", # move the plot title start slightly
    legend.position = "bottom" # by default, put legend on the bottom
  ))

set.seed(2021)
```

GLMs Overview

We have worked with different GLMs in the second part of the course and as a review, you can find all the main information about them in the table below. As you can see, the models differ by their stochastic components and link functions. Both of these pieces of information depict the differences in the steps we follow when generating quantities of interest. In particular, we need to select the appropriate response function when transforming the linear predictor `XBeta` when working with the expected values and, when generating predicted values, to draw from the appropriate stochastic component.

Table 1: Note: $\Phi(.)$ is the cumulative distribution function of the standard normal distribution.

Model	Stochastic Component	Systematic Component: Linear Predictor	Systematic Component: Link Function	Inverse Link Function (Response function)
Linear	$Y \sim \mathcal{N}(\mu_\rangle, \sigma)$ `rnorm()`	$\mu_i = X_i\beta$	μ_i	μ_i
Logit	$Y \sim Bernoulli(\pi_i)$ `rbinom(size = 1)`	$\mu_i = X_i\beta$	$\mu_i = \log \frac{\pi_i}{1-\pi_i}$	$\pi_i = \frac{\exp(\mu_i)}{1+\exp(\mu_i)}$ `plogis()`
Probit	$Y \sim Bernoulli(\pi_i)$ `rbinom(size = 1)`	$\mu_i = X_i\beta$	$\mu_i = \Phi^{-1}(\pi_i)$	$\pi_i = \Phi(\mu_i)$ `pnorm()`
Poisson	$Y \sim Pois(\lambda_i)$ `rpois()`	$\mu_i = X_i\beta$	$\mu_i = \log \lambda_i$	$\lambda_i = e^{\mu_i}$ `exp()`
Negative Binomial	$Y \sim NegBin(\lambda_i, \theta)$ `rnbinom()`	$\mu_i = X_i\beta$	$\mu_i = \log \lambda_i$	$\lambda_i = e^{\mu_i}$ `exp()`

Interactions in Non-Linear Models

Unfortunately, the intuition about interaction terms from linear models does not extend to non-linear models. The marginal effects will not be linear any more. However, we have one really powerful tool in our toolbox that can help us to look at and interpret interactions in any model - simulation! The same logic applies to any other non-linear model.

Count Models

Let's have another look at the dataset from last week from Eck and Hultman (2007), who study direct and deliberate killings of civilians, called one-sided violence, in intrastate armed conflicts. Like last time, we'll exclude Rwanda from the analysis, and now we'll estimate the model with an interaction between the regime types and the prior one-sided killings.

```
dta <- read.dta("raw-data/eck_rep.dta")
dta$os_best[dta$os_best == 500000] <- NA # Rwanda

# model from last time
m1 <-
  glm.nb(
    os_best ~ intensity_dyad + auto + demo + govt + prior_os,
    data = dta,
    control = glm.control(maxit = 200)
  )

# model with an interaction
m2 <-
  glm.nb(
    os_best ~ intensity_dyad + auto + demo + govt + prior_os * auto + prior_os * demo,
    data = dta,
    control = glm.control(maxit = 200)
  )
summary(m2)
```

```
##
## Call:
## glm.nb(formula = os_best ~ intensity_dyad + auto + demo + govt +
##     prior_os * auto + prior_os * demo, data = dta, control = glm.control(maxit = 200),
##     init.theta = 0.04191725444, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8063  -0.7118  -0.7077  -0.6357   2.2185
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     0.624635   0.573917   1.088  0.27643
## intensity_dyad  1.015466   0.346589   2.930  0.00339 **
## auto            1.203475   0.436822   2.755  0.00587 **
## demo            1.133847   0.469202   2.417  0.01567 *
## govt            0.024951   0.301170   0.083  0.93397
## prior_os        0.016332   0.003723   4.387 1.15e-05 ***
```

```
## auto:prior_os  -0.007032   0.003773  -1.864  0.06233 .
## demo:prior_os  -0.010717   0.004045  -2.650  0.00806 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.0419) family taken to be 1)
##
##     Null deviance: 605.20  on 1158  degrees of freedom
## Residual deviance: 537.05  on 1151  degrees of freedom
##   (116 observations deleted due to missingness)
## AIC: 4689.5
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.04192
##           Std. Err.:  0.00288
##
##  2 x log-likelihood:  -4671.47000
```

Quantitites of Interest

Steps for Simulating Parameters (Estimation Uncertainty): (King, Tomz, and Wittenberg 2000)

1. Get the coefficients from the regression (`gamma_hat`)
2. Get the variance-covariance matrix (`V_hat`)
3. Set up a multivariate normal distribution N(`gamma_hat`,`V_hat`)
4. Draw from the distribution `nsim` times

```
nsim <- 1000
gamma_hat <- coef(m2)
V_hat <- vcov(m2)
S <- mvrnorm(nsim, gamma_hat, V_hat)
```

Set up interesting scenarios:

```
names(gamma_hat)
## [1] "(Intercept)"    "intensity_dyad" "auto"           "demo"
## [5] "govt"           "prior_os"       "auto:prior_os"  "demo:prior_os"
```

```
# create sequence for the continuous variable
prior_os_seq <- round(seq(
  min(m2$model$prior_os),
  quantile(m2$model$prior_os, 0.95), # why not max?
  length.out = 100
))

# set scenario for anocracies (baseline category)
scenario1 <- cbind(
  1, # Intercept
  median(dta$intensity_dyad, na.rm = TRUE), # median of civil war (dyadic)
  0, # autocracy
```

```
  0, # democracy
  median(dta$govt, na.rm = TRUE), # median of one-sided violence by government
  prior_os_seq, # mean of prior one-sided violence
  prior_os_seq * 0, # autocracy :prior_os
  prior_os_seq * 0 # democracy :prior_os
)
colnames(scenario1) <- names(gamma_hat)
head(scenario1)
```

```
##      (Intercept) intensity_dyad auto demo govt prior_os auto:prior_os
## [1,]           1              1    0    0    0        0             0
## [2,]           1              1    0    0    0        2             0
## [3,]           1              1    0    0    0        4             0
## [4,]           1              1    0    0    0        7             0
## [5,]           1              1    0    0    0        9             0
## [6,]           1              1    0    0    0       11             0
##      demo:prior_os
## [1,]             0
## [2,]             0
## [3,]             0
## [4,]             0
## [5,]             0
## [6,]             0
```

```
# copy existing scenario1 into new objects scenario2 & scenario3
scenario3 <- scenario2 <- scenario1

# switch only the changing values
# set scenario for democracies (baseline category)
scenario2[, which(colnames(scenario2) == "demo")] <- 1
scenario2[, which(colnames(scenario2) == "demo:prior_os")] <- prior_os_seq

# set scenario for autocracies (baseline category)
scenario3[, which(colnames(scenario3) == "auto")] <- 1
scenario3[, which(colnames(scenario3) == "auto:prior_os")] <- prior_os_seq
```

Now get the linear predictor:

```
Xbeta1 <- S %*% t(scenario1)
Xbeta2 <- S %*% t(scenario2)
Xbeta3 <- S %*% t(scenario3)
```

And then apply the response function to the linear predictor:

```
lambda1 <- exp(Xbeta1)
lambda2 <- exp(Xbeta2)
lambda3 <- exp(Xbeta3)
```

Additional Step: you can do the full procedure and calculate the many predicted values and average over fundamental uncertainty. The more draws you take, the more your confidence intervals will resemble the ones if you took the shortcut and just used `lambda` rights away. The calculation may take a little while, especially if you set the number of simulations to 10000 or larger.

Note that unlike last time, we are using `apply` and not `sapply` function for this, since `lambda`'s are now matrices and not vectors (unlike last time). In order to preserve the structure of the object and make our plotting easier, we use `apply` and specify both margins. This ensures that the function is preformed not across columns or rows, but to each value in the matrix.

```
theta <- m2$theta

exp_ano <-
  apply(lambda1, c(1, 2), function(x) {
    mean(rnbinom(100, size = theta, mu = x))
  })

exp_demo <-
  apply(lambda2, c(1, 2), function(x) {
    mean(rnbinom(100, size = theta, mu = x))
  })

exp_auto <-
  apply(lambda3, c(1, 2), function(x) {
    mean(rnbinom(100, size = theta, mu = x))
  })
```

Summarize the results: get 2.5%, 50%, and 97.5% percentiles.

```
quants_ano <- t(apply(exp_ano, 2, quantile, c(0.025, 0.5, 0.975)))
quants_demo <- t(apply(exp_demo, 2, quantile, c(0.025, 0.5, 0.975)))
quants_auto <- t(apply(exp_auto, 2, quantile, c(0.025, 0.5, 0.975)))
colnames(quants_ano)
## [1] "2.5%"  "50%"   "97.5%"
```

Plot it. We here show you to options to plot such quantities. Segment plots emphasize the discrete nature of the independent variable, while the plots with polygon arguably has somewhat better visibility with overlapping. The choice of the plot will depend on the nature of the independent variable as well as the aesthetics.

Base R

```
par(las = 1)
# segment plot
plot(
  prior_os_seq,
  quants_ano[, "50%"],
  type = "n",
  ylim = c(0, 1000),
  ylab = "One-sided Killings",
  xlab = "Prior One-sided Killings",
  bty = "n",
  main = "Expected One-sided Killings by Rebel Groups in Civil War Situation"
)

segments(
```

```
  x0 = prior_os_seq, x1 = prior_os_seq,
  y1 = quants_ano[, "97.5%"], y0 = quants_ano[, "2.5%"],
  col = viridis(3, 0.5)[1],
  lwd = 2
)
points(prior_os_seq, quants_ano[, 2], col = viridis(3, 0.5)[1], pch = 20)

segments(
  x0 = prior_os_seq, x1 = prior_os_seq,
  y1 = quants_demo[, "97.5%"], y0 = quants_demo[, "2.5%"],
  col = viridis(3, 0.5)[2],
  lwd = 2
)
points(prior_os_seq, quants_demo[, 2], col = viridis(3, 0.5)[2], pch = 20)

segments(
  x0 = prior_os_seq, x1 = prior_os_seq,
  y1 = quants_auto[, "97.5%"], y0 = quants_auto[, "2.5%"],
  col = viridis(3, 0.5)[3],
  lwd = 2
)
points(prior_os_seq, quants_auto[, 2], col = viridis(3, 0.5)[3], pch = 20)

# Add a "histogram" of actual X1-values.
axis(
  1,
  at = dta$prior_os,
  col.ticks = "gray30",
  labels = FALSE,
  tck = 0.02
)

legend(
  "topleft",
  legend = c(
    "Median & 95% CI:",
    "Anocracy",
    "Democracy",
    "Autocracy"
  ),
  col = c(
    "white",
    viridis(3, 0.5)
  ),
  lty = "solid",
  lwd = 2,
  pch = 20,
  pt.cex = 2,
  bty = "n"
)
```

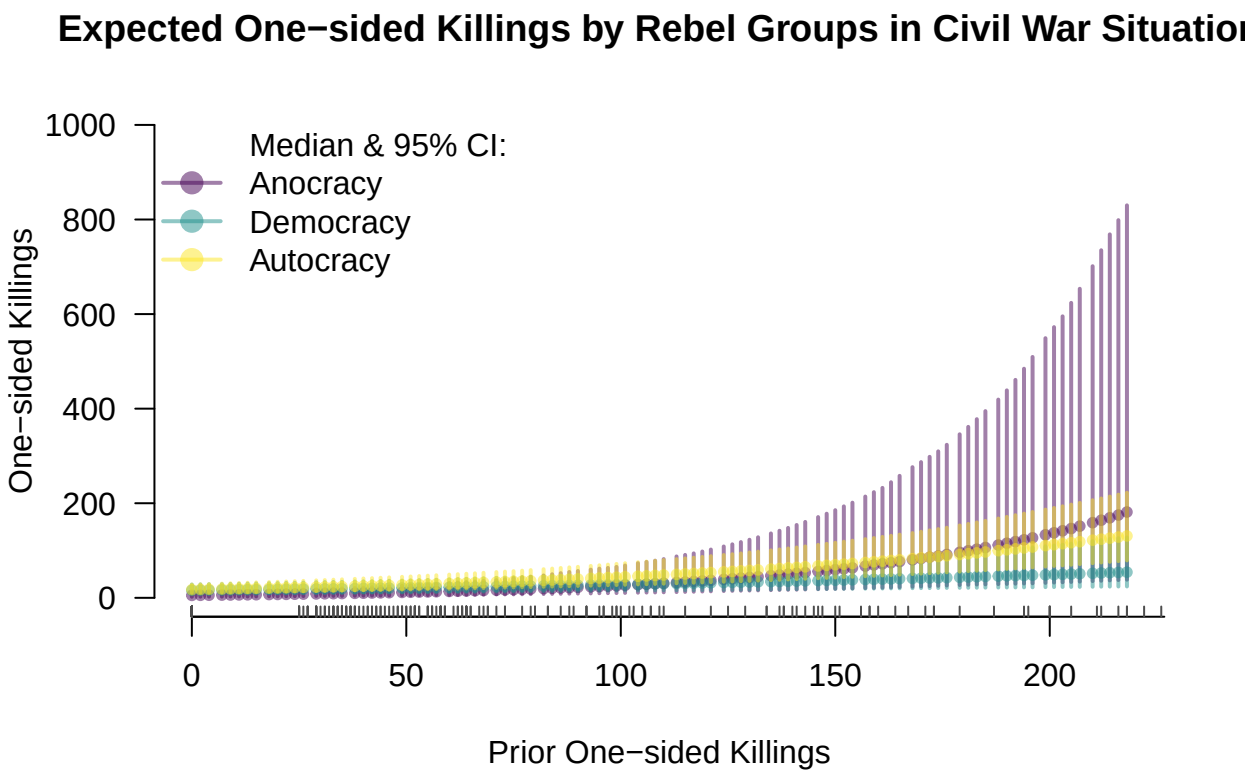


Figure 1: **Expected number of one-sided killings in autocracies, democracies, and anocracies over the range of prior one-sided killings**. Simulation based on model 2, other variables set to average value in dataset. Segments depict 95% confidence intervals. Data from Eck & Hultman (2007).

```r
# polygon plot
plot(
  prior_os_seq,
  quants_ano[, "50%"],
  type = "n",
  ylim = c(0, 1000),
  ylab = "One-sided Killings",
  xlab = "Prior one-sided Killings",
  bty = "n",
  main = "Expected One-sided Killings by Rebel Groups in Civil War Situation"
)

polygon(
  c(rev(prior_os_seq), prior_os_seq),
  c(rev(quants_auto[, "97.5%"]), quants_auto[, "2.5%"]),
  col = viridis(3, 0.2)[3],
  border = NA
)

polygon(
  c(rev(prior_os_seq), prior_os_seq),
  c(rev(quants_demo[, "97.5%"]), quants_demo[, "2.5%"]),
  col = viridis(3, 0.2)[2],
  border = NA
)

polygon(
  c(rev(prior_os_seq), prior_os_seq),
  c(rev(quants_ano[, "97.5%"]), quants_ano[, "2.5%"]),
  col = viridis(3, 0.2)[1],
  border = NA
)

lines(prior_os_seq, quants_auto[, 2], lwd = 2, lty = "dashed", col = viridis(3, 0.5)[3])
lines(prior_os_seq, quants_auto[, 1], lwd = 0.5, lty = "dashed", col = viridis(3, 0.5)[3])
lines(prior_os_seq, quants_auto[, 3], lwd = 0.5, lty = "dashed", col = viridis(3, 0.5)[3])

lines(prior_os_seq, quants_demo[, 2], lwd = 2, lty = "dotted", col = viridis(3, 0.5)[2])
lines(prior_os_seq, quants_demo[, 1], lwd = 0.5, lty = "dashed", col = viridis(3, 0.5)[2])
lines(prior_os_seq, quants_demo[, 3], lwd = 0.5, lty = "dashed", col = viridis(3, 0.5)[2])

lines(prior_os_seq, quants_ano[, 2], lwd = 2, col = viridis(3, 0.5)[1])
lines(prior_os_seq, quants_ano[, 1], lty = "dashed", col = viridis(3, 0.5)[1])
lines(prior_os_seq, quants_ano[, 3], lty = "dashed", col = viridis(3, 0.5)[1])


# Add a "histogram" of actual X1-values.
axis(
  1,
  at = dta$prior_os,
  col.ticks = "gray30",
  labels = FALSE,
  tck = 0.02
```

```
)

legend(
  "topleft",
  legend = c(
    "Median & 95% CI:",
    "Anocracy",
    "Democracy",
    "Autocracy"
  ),
  col = c(
    "white",
    viridis(3, 0.5)
  ),
  lty = c(NA, "solid", "dotted", "dashed"),
  lwd = c(NA, 2, 2, 2),
  # pch = c(NA, 15, NA, 15, NA, 15),
  pt.cex = 2,
  bty = "n"
)
```

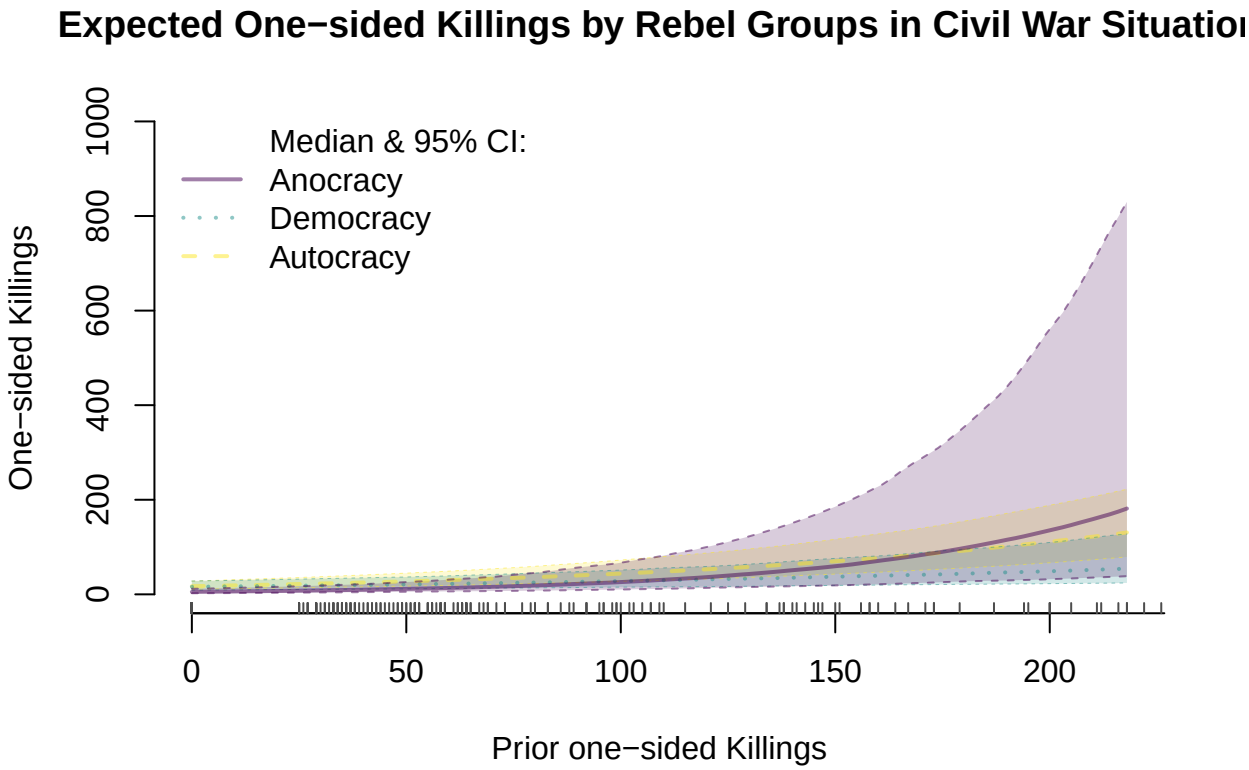
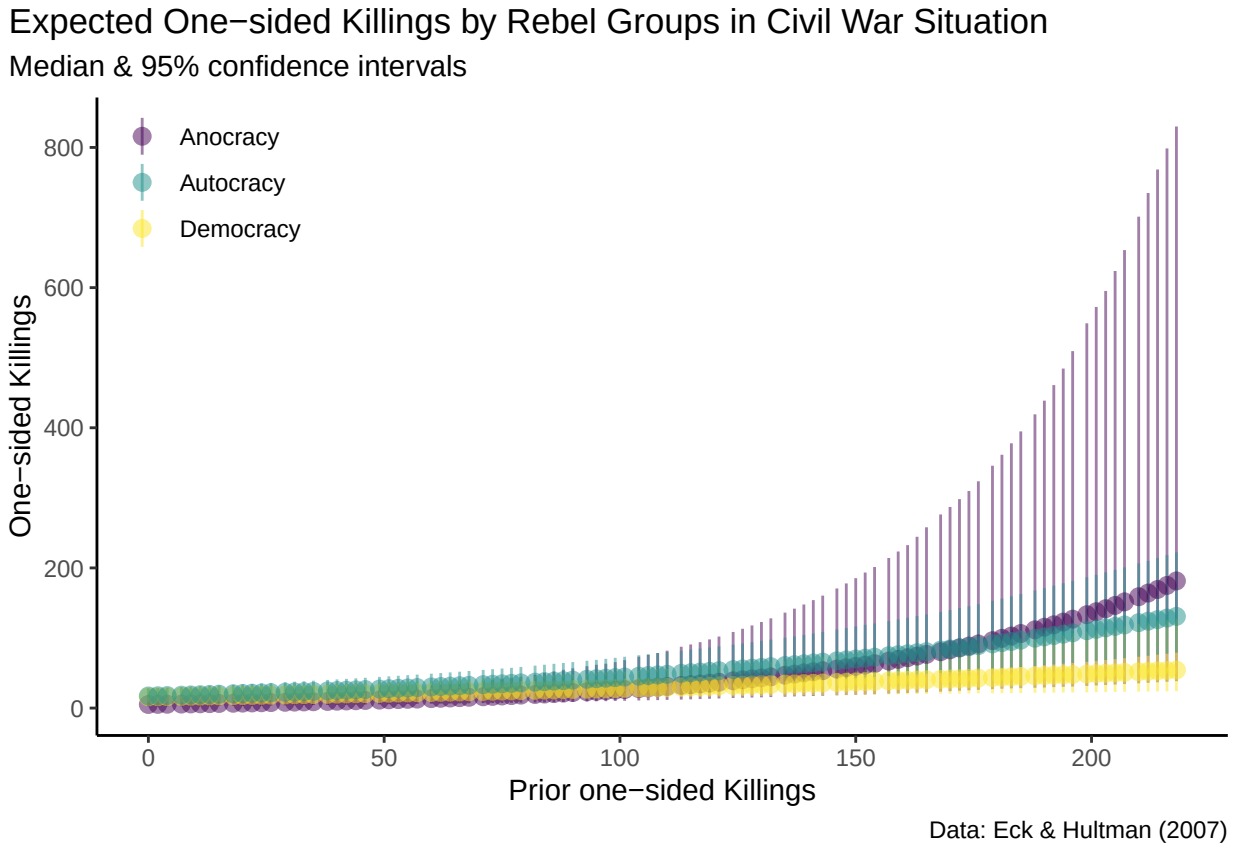


Figure 2: **Expected Number of One-sided Killings in Autocracies, Democracies, and Anocracies over the Range of Prior One-sided Killings.** Simulation based on model 2, other variables set to average value in dataset. Shaded areas depict 95% confidence intervals. Data from Eck & Hultman (2007)

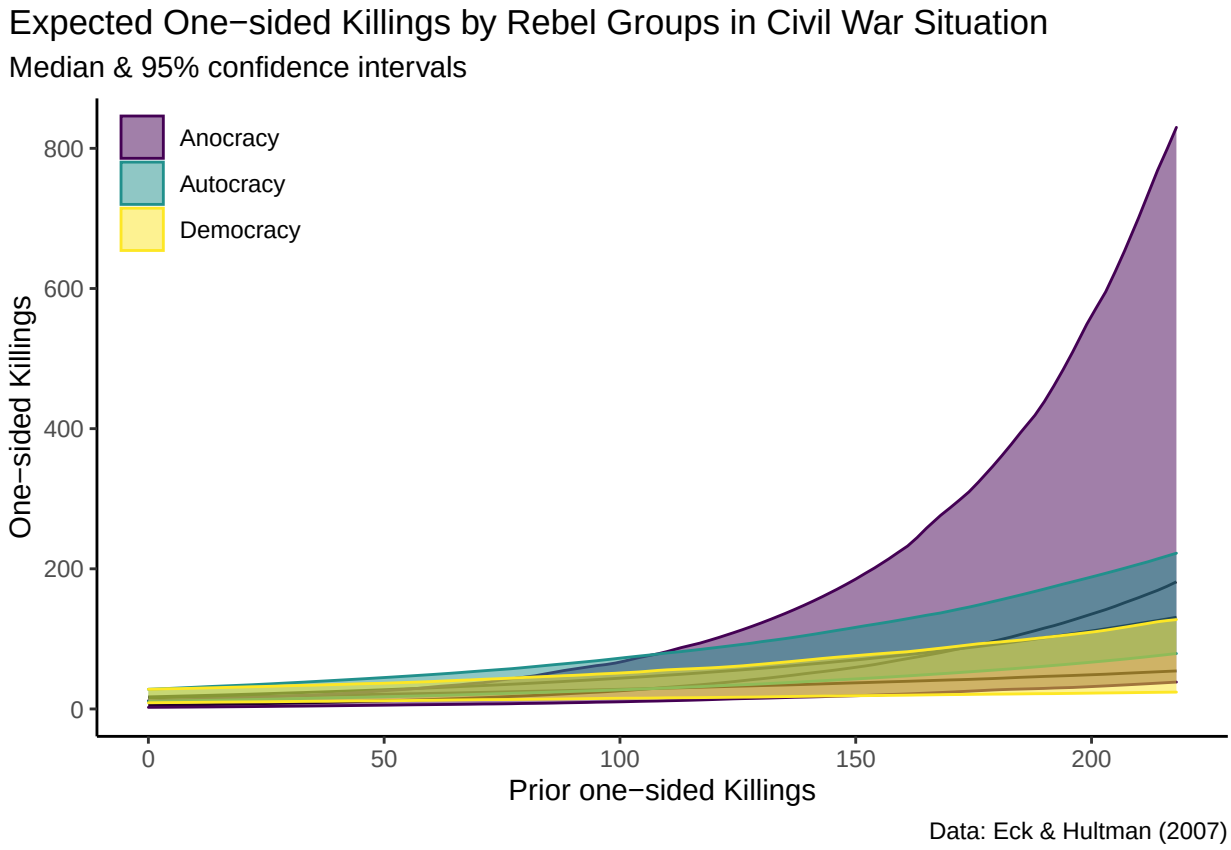
ggplot2

```
quants <- as.data.frame(rbind(quants_ano, quants_demo, quants_auto))
quants$regime <- rep(c("Anocracy", "Democracy", "Autocracy"), each = 100)
quants$prior_os_seq <- rep(prior_os_seq, times = 3)
quants <- clean_names(quants) # easier-to-work names
ggplot(
  data = quants,
  mapping = aes(x = prior_os_seq, y = x50_percent, group = regime)
) +
  geom_pointrange(aes(ymin = x2_5_percent, ymax = x97_5_percent, color = regime), alpha = 0.5) +
  scale_color_viridis_d() +
  labs(
    x = "Prior one-sided Killings",
    y = "One-sided Killings",
    color = "",
    title = "Expected One-sided Killings by Rebel Groups in Civil War Situation",
    subtitle = "Median & 95% confidence intervals",
    caption = "Data: Eck & Hultman (2007)"
  ) +
  theme(legend.position = c(0.1, 0.9))
```




```
ggplot(
  data = quants,
  mapping = aes(x = prior_os_seq, y = x50_percent, fill = regime)
```

```
) +
  geom_line() +
  geom_ribbon(aes(ymin = x2_5_percent, ymax = x97_5_percent, color = regime),
    alpha = 0.5
  ) +
  scale_color_viridis_d() +
  scale_fill_viridis_d() +
  labs(
    x = "Prior one-sided Killings",
    y = "One-sided Killings",
    color = "",
    fill = "",
    title = "Expected One-sided Killings by Rebel Groups in Civil War Situation",
    subtitle = "Median & 95% confidence intervals",
    caption = "Data: Eck & Hultman (2007)"
  ) +
  theme(legend.position = c(0.1, 0.9))
```



Meaningful Quantities of Interest

Let's start with exploring the differences in the effect of prior one-sided killings across the regimes and have a look at the respective first differences:

$$FD_{\text{Autocracy-Democracy}} = E(Y|X_{\text{Autocracy}}) - E(Y|X_{\text{Democracy}})$$

Base R

```
FD <- exp_auto - exp_demo
quants_FD <- t(apply(FD, 2, quantile, c(0.025, 0.5, 0.975)))

plot(
  prior_os_seq,
  quants_FD[, "50%"],
  type = "n",
  ylim = c(min(quants_FD[, "2.5%"]), max(quants_FD[, "97.5%"])),
  ylab = "Difference in Expected One-sided Killings",
  xlab = "Prior One-sided Killings",
  bty = "n",
  las = 1
)
segments(
  x0 = prior_os_seq, x1 = prior_os_seq,
  y1 = quants_FD[, "97.5%"], y0 = quants_FD[, "2.5%"],
  col = viridis(1, 0.5)
)
points(prior_os_seq, quants_FD[, 2], col = viridis(1, 0.5), pch = 20)

abline(h = 0, lty = "dashed")
```

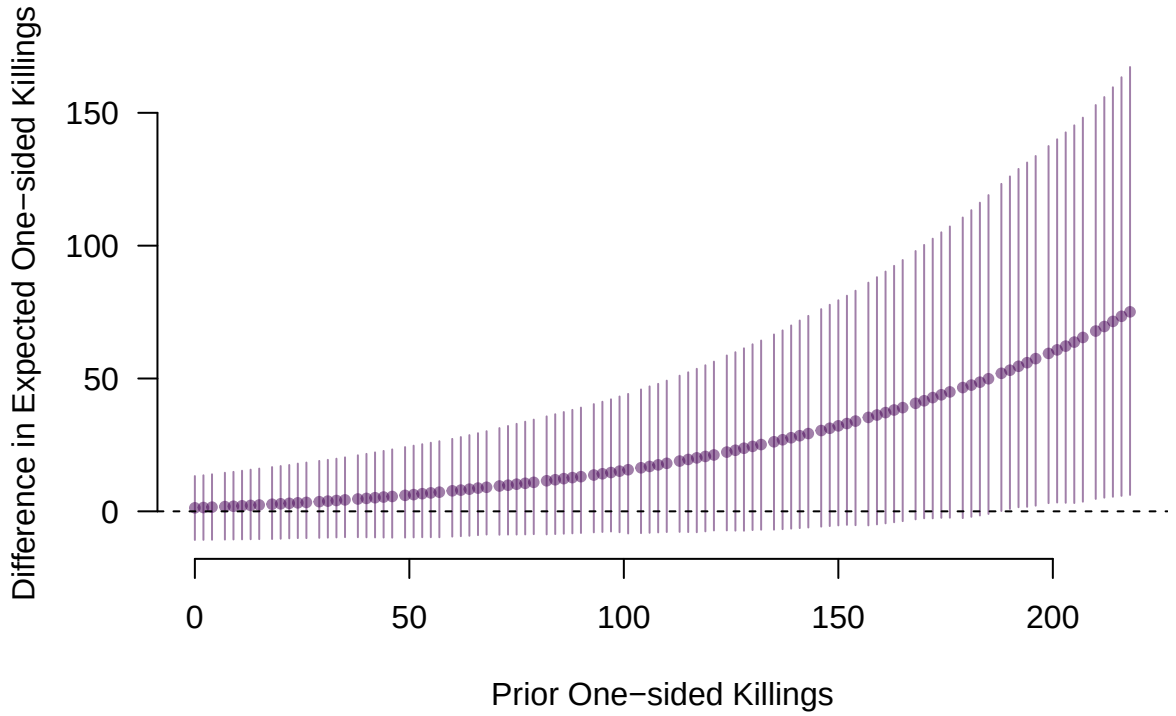


Figure 3: **Difference in expected number of one-sided killings in autocracies and democracies over the Range of prior one-sided killings**. Simulation based on model 2, other variables set to average value in dataset. Segments depict 95% confidence intervals. Data from Eck & Hultman (2007).

Here we see that there seems to be no significant difference between the effect of *prior one-sided killings* on the *one-sided killings* in the future **between** democracies and autocracies for the specified scenario at 5%

significance level when prior one-sided killings are below 188. However, when prior expected killings rise to 216, the difference between autocracies and democracies given our scenario is, on average, 73, yet it varies from 6 to 163 (based on the 95% confidence interval). In other words, we can only claim that for values above 188 of *prior one-sided killings* there is a significant difference in *one-sided killings* given our model estimates.

> Do these first differences allow us to make statements about the effect of *prior one-sided killings* on the *one-sided killings* **within** the regimes?

If our primary goal was to show that the *prior one-sided killings* has stronger effect on the *one-sided killings* in democracies than in autocracies, and we wanted to quantify this difference, we would be calculating a different quantity of interest.

```
FD <- cbind(
  exp_auto[, 95] - exp_auto[, 5],
  exp_demo[, 95] - exp_demo[, 5],
  exp_ano[, 95] - exp_ano[, 5]
)
quants_FD <- t(apply(FD, 2, quantile, c(0.025, 0.5, 0.975)))

plot(
  y = 1:3,
  x = quants_FD[, "50%"],
  type = "n",
  ylab = "",
  xlim = range(pretty(c(
    min(quants_FD[, "2.5%"]), max(quants_FD[, "97.5%"])
  ))),
  ylim = c(1, 2.6),
  xlab = "Difference in Expected One-sided Killings",
  bty = "n",
  las = 1,
  axes = F
)

segments(
  x0 = quants_FD[, "2.5%"],
  x1 = quants_FD[, "97.5%"],
  y0 = c(1.2, 1.8, 2.4)
)
points(
  x = quants_FD[, "50%"],
  y = c(1.2, 1.8, 2.4),
  pch = 19
)
axis(
  2,
  at =  c(1.2, 1.8, 2.4),
  las = 1,
  labels = c("Autocracy", "Democracy", "Anocracy"),
  tick = F,
  line = F,
  hadj = 0.7
```

```
)
axis(1)
abline(v = 0, lty = "dashed")
```

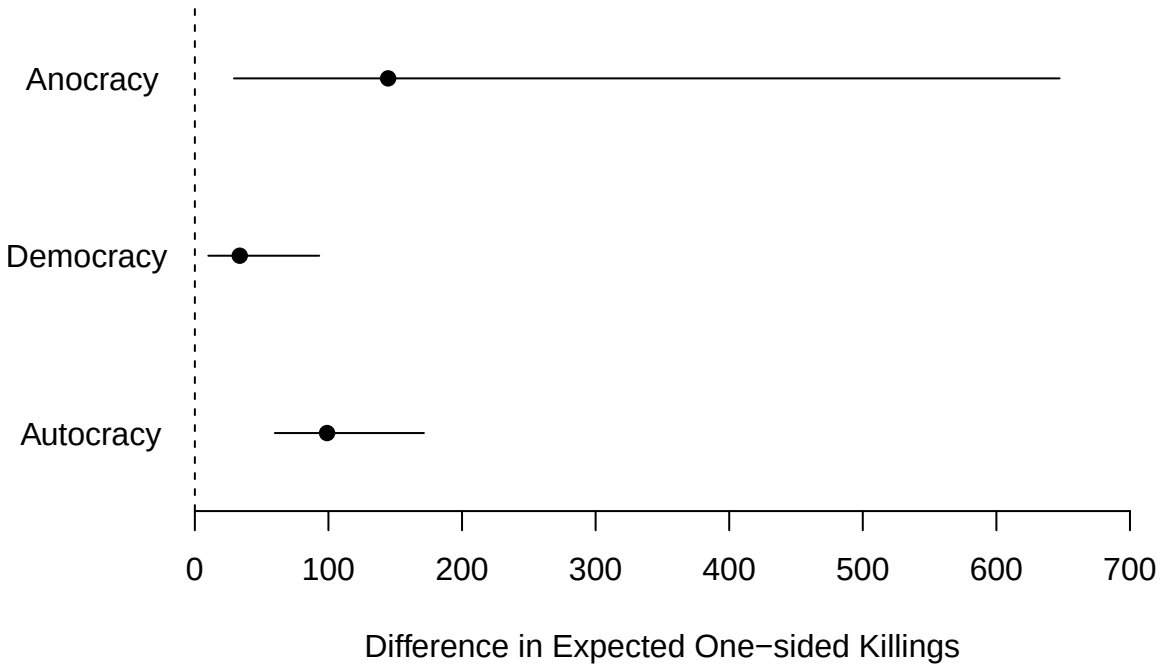


Figure 4: **Difference in expected number of one-sided killings bewteen high and low numbers of prior one-sided killings**. Simulation based on model 2, other variables set to average value in dataset. Segments depict 95% confidence intervals. Data from Eck & Hultman (2007).

What information is missing on this plot?

These quantities allow us to explore the effect of *Prior One-sided Killings* within the regimes. Depending on the theory you are testing, you may be more interested in this quantity rather than the difference across the regimes. Most often, we will want to distinguish between our main variable of interest and the moderator of it's effect. In general, it is the values of the key independent variable of interest that we should vary in calculating the first differences.

ggplot2

```
FD <- exp_auto - exp_demo
quants_FD <- t(apply(FD, 2, quantile, c(0.025, 0.5, 0.975)))
quants_FD <- clean_names(as.data.frame(quants_FD))
quants_FD$prior_os_seq <- prior_os_seq
ggplot(quants_FD, aes(x = prior_os_seq, y = x50_percent)) +
  geom_pointrange(aes(ymin = x2_5_percent, ymax = x97_5_percent), color = viridis(1)) +
  labs(
    y = "Difference in Expected One-sided Killings",
    x = "Prior One-sided Killings"
  ) +
  geom_hline(yintercept = 0, linetype = "dashed")
```

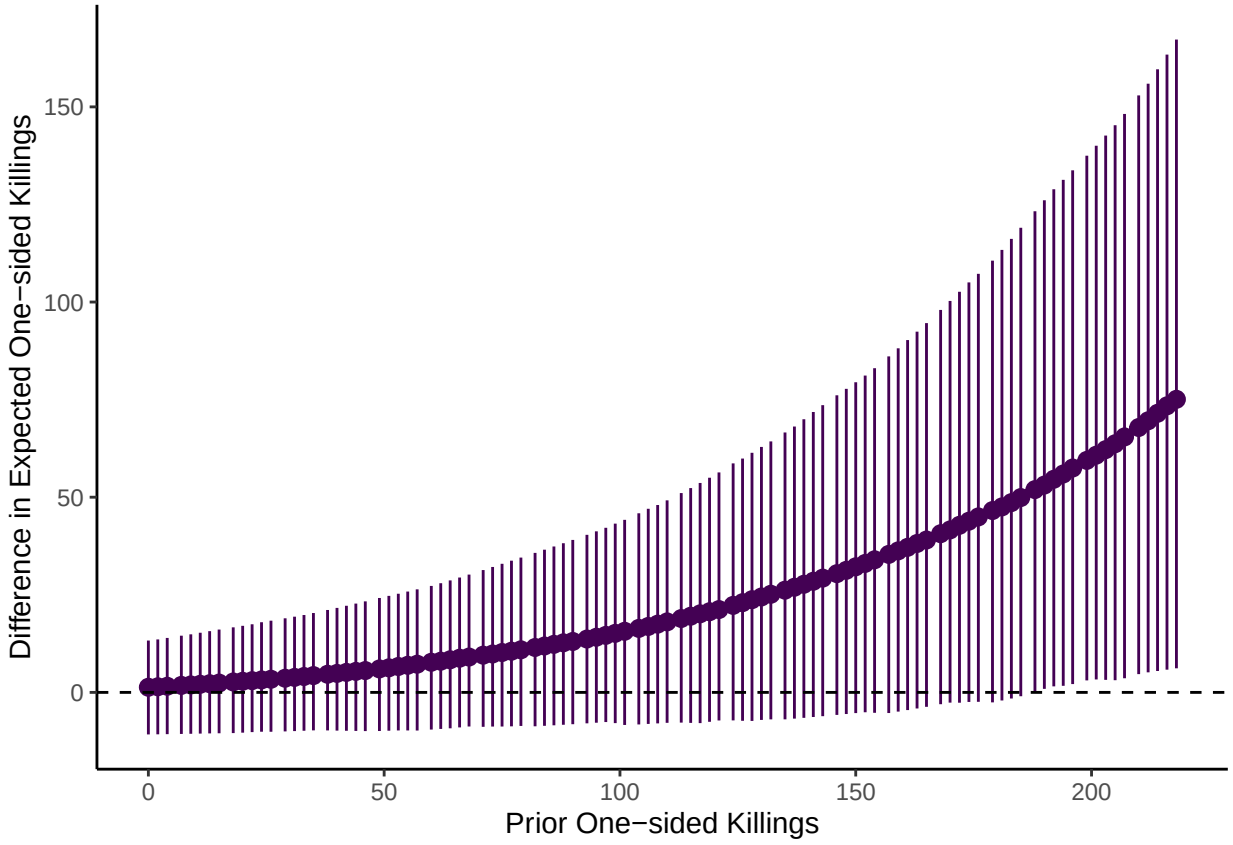


Figure 5: **Difference in expected number of one-sided killings in autocracies and democracies over the Range of prior one-sided killings**. Simulation based on model 2, other variables set to average value in dataset. Segments depict 95% confidence intervals. Data from Eck & Hultman (2007).

Here we see that there seems to be no significant difference between the effect of *prior one-sided killings* on the *one-sided killings* in the future **between** democracies and autocracies at 5% significance level when prior one-sided killings are below 200. In other words, we can only claim that for values above 200 of *prior one-sided killings* there is a significant difference in *one-sided killings* given our model estimates.

> Do these first differences allow us to make statements about the effect of *prior one-sided killings* on the *one-sided killings* **within** the regimes?

If our primary goal was to show that the *prior one-sided killings* has stronger effect on the *one-sided killings* in democracies than in autocracies, and we wanted to quantify this difference, we would be calculating a different quantity of interest.

```
FD <- cbind(
  exp_auto[, 95] - exp_auto[, 5],
  exp_demo[, 95] - exp_demo[, 5],
  exp_ano[, 95] - exp_ano[, 5]
)
quants_FD <- t(apply(FD, 2, quantile, c(0.025, 0.5, 0.975)))
quants_FD <- clean_names(as.data.frame(quants_FD))
quants_FD$regime <- c("Autocracy", "Democracy", "Anocracy")

ggplot(data = quants_FD, aes(y = regime, x = x50_percent)) +
  geom_pointrange(aes(xmin = x2_5_percent, xmax = x97_5_percent)) +
  geom_vline(xintercept = 0, linetype = "dashed") +
  labs(
    y = "",
    x = "Difference in Expected One-side Killings"
  ) +
  theme(
    axis.line.y = element_blank(),
    axis.ticks.y = element_blank()
  )
```

> What information is missing on this plot?

These quantities allow us to explore the effect of *Prior One-sided Killings* within the regimes. Depending on the theory you are testing, you may be more interested in this quantity rather than the difference across the regimes. Most often, we will want to distinguish between our main variable of interest and the moderator of it's effect. In general, it is the values of the key independent variable of interest that we should vary in calculating the first differences.

Robustness Checks

> "All models are wrong, but some are useful." – George Box

We cannot specify our models perfectly and correctly since the data generation process and causal relationships are very complex. Instead, when doing modeling, we make assumptions about DGP and select model specification based on these assumptions. Having a reasonable baseline model with a reasonable set of covariates, our best attempt of optimizing the specification of the empirical model, is not where we should stop. Once we have a good baseline model, we should try to see whether the results obtained by this model

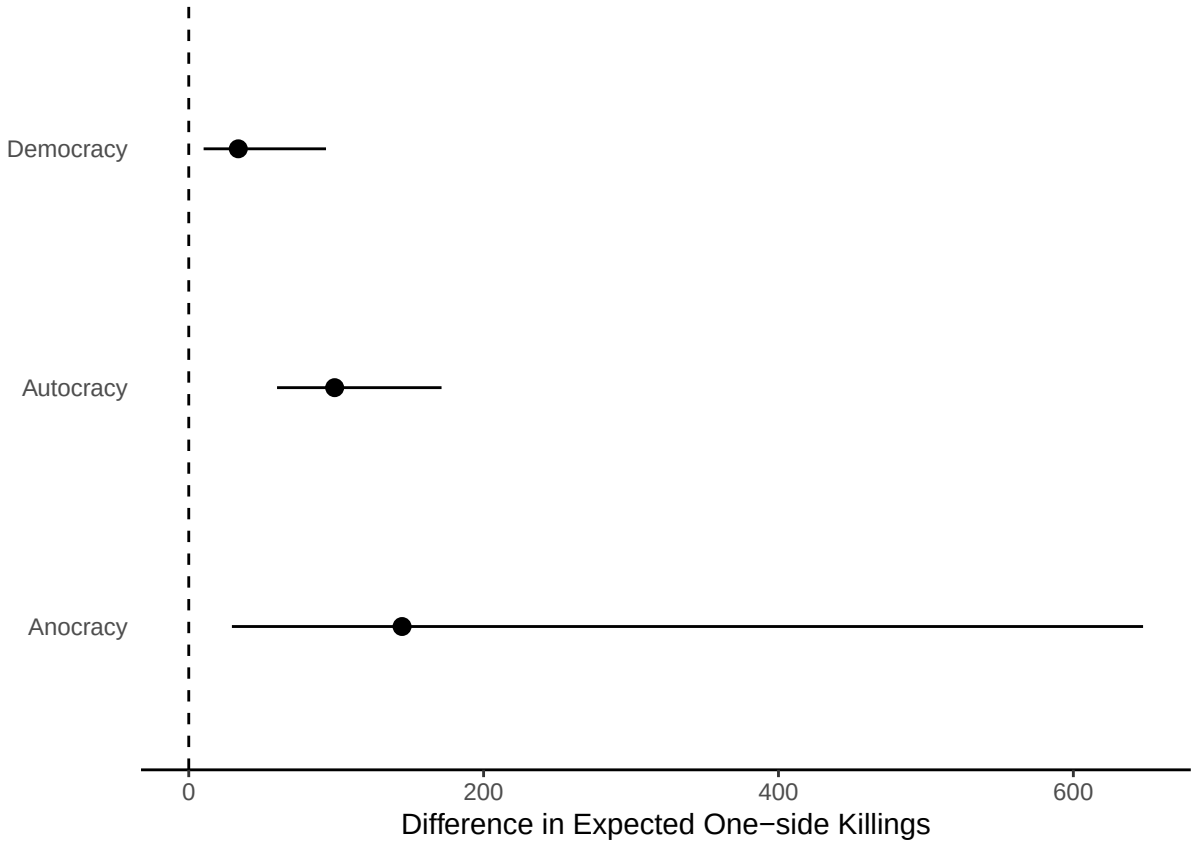


Figure 6: **Difference in expected number of one-sided killings bewteen high and low numbers of prior one-sided killings**. Simulation based on model 2, other variables set to average value in dataset. Segments depict 95% confidence intervals. Data from Eck & Hultman (2007).

hold when we substitute the baseline model specification with plausible alternatives. This is the practice of robustness testing.

In short, with robustness testing we analyze if the estimated effects of interest are sensitive to changes in model specifications. Robustness tests can increase the validity of inferences.

Population Definition and Sample Tests

A very common test is *outlier elimination*, where one essentially drops the outliers. Keep in mind, however, that if the model is strongly misspecified, outlier tests are more likely to pick up the consequences of model misspecification than to detect true outliers (cases that are not part of the population), thereby making bias potentially worse. We can try implementing this test for our model from before. Following the approach in (Hilbe 2009), we'll treat observations with so-called standardized deviance residuals (a generalization of $\hat{\epsilon}_i$ for GLMs) greater than 2 as potential outliers.

```
summary(m1)
```

```
##
## Call:
## glm.nb(formula = os_best ~ intensity_dyad + auto + demo + govt +
##     prior_os, data = dta, control = glm.control(maxit = 200),
##     init.theta = 0.04172548365, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8178  -0.7097  -0.7084  -0.6542   2.3510
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.9389889  0.5644146   1.664  0.09618 .
## intensity_dyad 1.0077767  0.3470595   2.904  0.00369 **
## auto           0.8881510  0.4187919   2.121  0.03394 *
## demo           0.7217879  0.4512491   1.600  0.10970
## govt           0.0209680  0.3011477   0.070  0.94449
## prior_os       0.0095023  0.0005744  16.543  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.0417) family taken to be 1)
##
##     Null deviance: 602.77  on 1158  degrees of freedom
## Residual deviance: 536.89  on 1153  degrees of freedom
##   (116 observations deleted due to missingness)
## AIC: 4687.4
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.04173
##           Std. Err.:  0.00286
##
##  2 x log-likelihood:  -4673.42500
```

```r
# remove outliers
m3 <- glm.nb(formula = os_best ~ intensity_dyad + auto + demo + govt +
  prior_os, data = m1$model[!rstandard(m1) > 2, ], control = glm.control(maxit = 200), )
summary(m3)
```

```
##
## Call:
## glm.nb(formula = os_best ~ intensity_dyad + auto + demo + govt +
##     prior_os, data = m1$model[!rstandard(m1) > 2, ], control = glm.control(maxit = 200),
##     init.theta = 0.04260388047, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9056  -0.7099  -0.7075  -0.6591   2.0168
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     1.2171546  0.5591007   2.177   0.0295 *
## intensity_dyad  0.7594002  0.3442984   2.206   0.0274 *
## auto            0.7796142  0.4146040   1.880   0.0601 .
## demo            0.4863755  0.4468627   1.088   0.2764
## govt           -0.0408280  0.2984301  -0.137   0.8912
## prior_os        0.0105034  0.0005685  18.476   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.0426) family taken to be 1)
##
##     Null deviance: 608.48  on 1156  degrees of freedom
## Residual deviance: 534.10  on 1151  degrees of freedom
## AIC: 4631.9
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.04260
##           Std. Err.:  0.00294
##
##  2 x log-likelihood:  -4617.85300
```

One can also consider gradually *expanding the sample size* and moving away from what they consider to be the sample "for which the theoretical framework most directly applies," as Scheve and Slaughter (2004) do in their analysis of whether economic integration increases worker insecurity in advanced economies. Authors point out:

> Our core results are for a sample of private-sector, full-time, not-self-employed workers: the labor-market participants for which the theoretical framework most directly applies. Our FDI-insecurity correlation of interest maintained in estimates of key specifications using broader samples.

One more approach related to samples is a **placebo test**, i.e. selecting a sample for which the theory should not apply and the effects should not be found. This is what Reuter and Szakonyi (2019) do when studying defections from the ruling party in an autocratic regime and showing that the effects are only found for the candidates in the ruling party and not other parties, as their theory argues.

Concept Validity and Measurement Tests

Another commonly used approach is to use an **alternative operationalization** of your dependent or key independent variables. For instance, Scheve and Slaughter (2004) uses various measures of their dependent variable, Foreign Direct Investment exposure (they have a great robustness tests section). Looking back at our Eck and Hultman (2007) example, showing that your results do not depend on, for instance, the definition of democracy that we use, could be an example of such test. For instance, if we are interested in the effect of regimes on one-sided killings, we would want to show that our results hold when we use the Polity score and the Democracy-Dictatorship measure (Cheibub, Gandhi, and Vreeland 2009) (although in this particular case since we require an intermediate category between autocracy and democracy, a dichotomous measure would not be appropriate).

If we keep working with the model from before, we can explore if the continuous measure of regime type produces similar results to the categorical one (yet this will not be the best test in this case):

```
# continuous Polity score instead of categories
# squared term for the inverse-U shape (democracy & autocracy have lower
# killings than anocracy, the middle category)
m5 <-
  glm.nb(
    formula = os_best ~ intensity_dyad + polity2 + polity_sq + govt + prior_os,
    data = dta,
    control = glm.control(maxit = 200),
  )
S <- mvrnorm(1000, coef(m5), vcov(m5))
polity_seq <- -10:10
Xbeta <-
  S %*% t(cbind(1, 1, polity_seq, polity_seq^2, 1, mean(dta$prior_os, na.rm = T)))
lambda <- exp(Xbeta)
quants <- t(apply(lambda, 2, quantile, c(0.025, 0.5, 0.975)))
plot(
  x = polity_seq,
  y = quants[, 2],
  type = "p",
  ylim = c(0, 80),
  pch = 19,
  las = 1,
  ylab = "One-sided Killings",
  xlab = "Polity2 Score"
)
segments(
  x0 = polity_seq,
  y0 = quants[, 1],
  y1 = quants[, 3]
)
axis(1,
  at = c(-10, 10),
  labels = c("Least democratic", "Most democratic"),
  padj = 1.2
)
```

As we see, the relationship between the variables does not seem to be U-shaped, as Eck and Hultman (2007) point out in the text. We should note, however, that using a 21-point scale implies a strong assumption on the effect of a variable: implicitly, the operationalization assumes that changes from, say, -10 to -5 represent

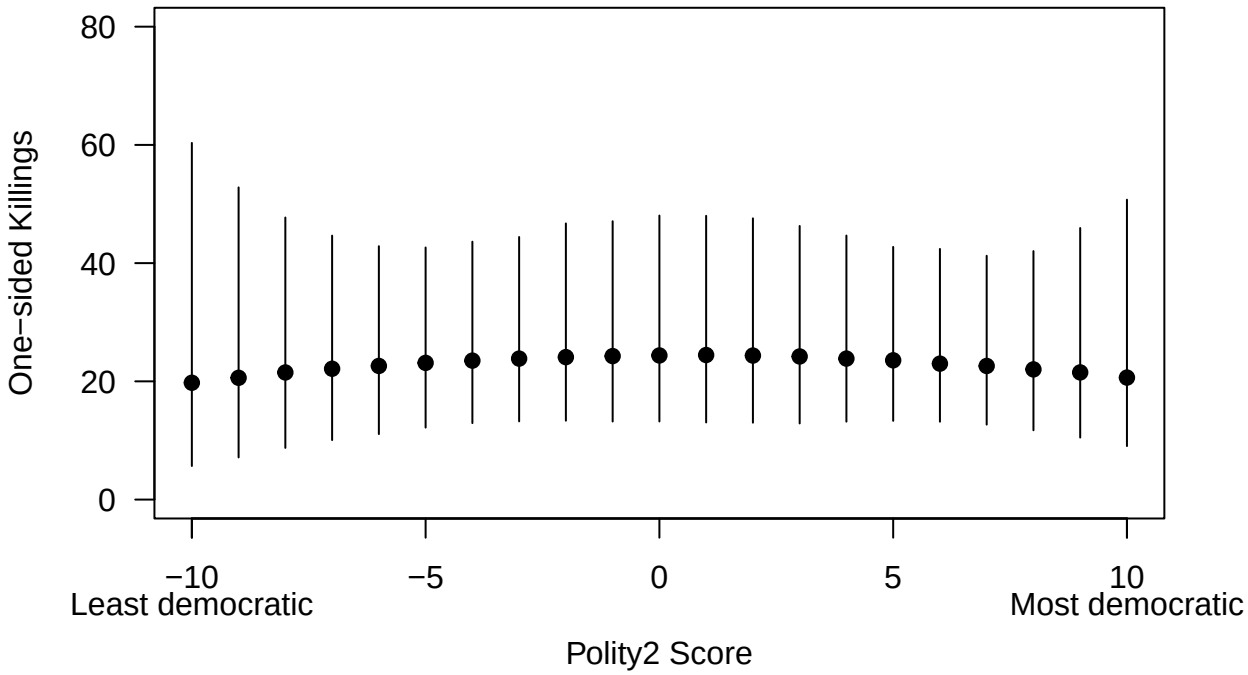


Figure 7: **Expected number of one-sided killings over the Range of Polity2 score**. Simulation based on model 5, other variables set to average value in dataset. Segments depict 95% confidence intervals. Data from Eck & Hultman (2007).

an equally strong move to a more democratic regime as a move from 5 to 10. This is a very strong assumption to make and usually, it will be a more appropriate choice to avoid making such an assumption. Hence, it would be a good test for robustness to use some aggregation of a continuous predictor of such kind.

We thus need to have a closer look at the values the authors used as thresholds. If we are transforming Polity score into a categorical variable, we'd want to show that our results do not depend on the arbitrary cut-off point we used for distinguishing between democracy, anocracy, and autocracy.

```
# check existing cutoffs
table(dta$auto, dta$polity2)
##
##      -9  -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5   6   7   8   9
##   0   0   0   0   0   0   0   0   0   0   0  58   6  24  29  43  20  66 134 119
##   1  27  31 184  90  20  51  60  22  34 141   0   0   0   0   0   0   0   0   0
##
##      10
##   0  40
##   1   0
table(dta$demo, dta$polity2)
##
##      -9  -8  -7  -6  -5  -4  -3  -2  -1   0   1   2   3   4   5   6   7   8   9
##   0  27  31 184  90  20  51  60  22  34 141  58   6  24  29  43  20   0   0   0
##   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0  66 134 119
##
##      10
##   0   0
##   1  40
```

```r
coefs_matrix <- NULL
# we can select some theoretically sensible values for cutoffs
for (demo_cutoff in 3:8) {
  for (auto_cutoff in -3:0) {
    dta$demo1 <- ifelse(dta$polity2 > demo_cutoff, 1,
      ifelse(is.na(dta$polity2), NA, 0)
    )
    dta$auto1 <- ifelse(dta$polity2 < auto_cutoff, 1,
      ifelse(is.na(dta$polity2), NA, 0)
    )
    m <- glm.nb(
      formula = os_best ~ intensity_dyad + auto1 + demo1 + govt + prior_os,
      data = dta,
      control = glm.control(maxit = 200),
    )
    coefs <- c(auto_cutoff, demo_cutoff, coef(m))
    coefs_matrix <- rbind(coefs, coefs_matrix)
  }
}
summary(coefs_matrix[, "auto1"])
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.43514 -0.18977 -0.06100 -0.07932  0.03404  0.19865
summary(coefs_matrix[, "demo1"])
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -0.5916 -0.2687 -0.1406 -0.1013 -0.0121  0.4644
```

For this particular model, it seems to be the case the estimates depend heavily on the cut-off points even if we look only at the signs of the estimates, hence questioning the robustness of findings from the main model.

Note, however, that should we select some meaningless cut-off points, we would expect it that the model estimates should not hold. Such an approach could be treated as a **placebo test** for the model.

Formatting and RMarkdown

For your data essay, you may choose to write the complete paper in RMarkdown (and this will be a very efficient option). You will be able to both do all analyses and write up the text in the same program. Here we will review the most relevant RMarkdown aspects when it comes to generating PDF output: YAML details, most commonly used chunk options, advice on tables and figures formatting, and doing citations. But before we move to them, make sure that you had a look at the Visual Editor interface, that makes formatting much easier in case, especially for tables and citations. Rstudio's guide contains lots of aspects that can help you work more efficiently.

YAML

Your standard YAML when creating a PDF will contain the title, author, date, and the output types:

```yaml
---
title: "Data Essay"
author: Anna Schmidt
date: December 8, 2021
output: pdf_document
---
```

You may want to add the table of contents, and `toc_depth` will define how many levels are depicted in TOC:

```
output:
  pdf_document:
    toc: true
    toc_depth: 2
```

You can also customize the size of the figures for the entire document (you will be able to change it for certain chunks with chunk options):

```
---
output:
  pdf_document:
    fig_width: 7
    fig_height: 6
    fig_caption: true
---
```

Should you choose to do so, you can customize the font size and margins like this:

```
---
title: "Data Essay"
output: pdf_document
fontsize: 11pt
geometry: "left=3cm,right=3cm,top=2.5cm,bottom=2.5cm"
---
```

Dealing with Unicode character error is a common problem., and it can be avoided with specifying the engine that will generate the PDF document. By default, PDF documents are rendered using `pdflatex`. You can specify an alternate engine using the `latex_engine` option. Available engines are `pdflatex`, `xelatex`, and `lualatex`. For example:

```
---
title: "Data Essay"
output:
  pdf_document:
    latex_engine: xelatex
---
```

You can also make your reports a little more flexible and print out the date of knitting rather than the predefined date argument. This can be done with inline coding in R: `format(Sys.Date(), "%B %d, %Y")` will give date in the format "December 3, 2021."

```
---
title: "Data Essay"
author: Anna Schmidt
date: December 08, 2021
output: pdf_document
---
```

> Make sure that if you use this option, the language of your R is the same as the language of the paper your are writing.

If you wish to store the figures your generated in your Rmd separately, you can use R commands to save them. But you can also set `keep_md: yes` and all the figures you generated will be stored in a new folder `_files`.

> Make sure to use meaningful chunk labels; otherwise, you will see a bunch of images named `unnamed-chunk-1` and navigating among these files will be harder than it should.

This works for both PDF and HTML outputs:

```
---
title: "Data Essay"
output:
  pdf_document:
    keep_md: yes
  html_document:
    keep_md: yes
---
```

Should you ever need to include any additional Latex packages, this is also straightforward:

```
---
title: "Data Essay"
header-includes:
- \usepackage{dcolumn}
output: pdf_document
---
```

This package ensures that `align=TRUE` argument in `stargazer` works correctly (yet this may still cause problems). If the error with `Missing $ inserted` arises, set `align=FALSE`.

Chunk Options

Here you can find an overview of the most often-used chunk options:

- `eval`: evaluate the code chunk?
- `echo`: display the source code in the output document?
- `include`: include the chunk output in the output document?
- `results='asis'`: write the raw text results directly into the output document without any markups (essential for `stargazer`)
- `collapse`: collapse all the source and output blocks from one code chunk into a single block?
- `warning:` preserve warnings in the output?
- `error:` preserve errors in the output?
- `messages:` preserve messages in the output?

While you can include some default options in the setup chunk inside the `knitr::opts_chunk$set`, you can also specify them directly for every chunk. For example:

```
knitr::opts_chunk$set(
  echo = TRUE,
  tidy = "styler" # styles the code in the output
)
```

For more on chunk options, please consult https://yihui.org/knitr/options/.

Citations

Here we will show you a way to add citations with the Visual Editor mode in Rstudio. If you'd like to learn a different way to do this, which does not require using Visual Editor mode, please consult our class website: Doing References in RMarkdown.

Bibiography in BibTex

To start with, you will need a file that contains all the bibliographic information about the texts you are using (you can add files there is necessary). We suggest you use the `bib` format, which is supported by most reference managers. The `bib` file will have one or many entries like this, one for each article/book/etc. you add:

```
@article{king2000making,
  title={Making the most of statistical analyses: Improving interpretation and presentation},
  author={King, Gary and Tomz, Michael and Wittenberg, Jason},
  journal={American journal of political science},
  pages={347--361},
  year={2000}
}
```

There is a unique identifier of the item, `king2000making` in this case, as well as the normal bibliographic information like the title and year of publication. There are various types of items, like articles or books.

Step 1: Add (empty) `bib` file

If you are already using any reference manager like Zotero or Mendeley, enter the texts you need into the manager as normal and export the `bib` (`Bibtex`) file into the project directory (where your `Rproj` file is located). Here is a way to do it in Zotero and in Mendeley.

If your project does not yet have a file with `bib` extension in your project directory, you can either copy-paste a file like this from any of our lab projects or create a new file with this extension inside Rstudio: *File -> New File -> Text File -> Save as > "citations.bib" .*

Step 2: Add Bibliography-related Parameters to YAML in `Rmd` file

Let's say you now have the `citations.bib` file in the folder of the project folder and your `Rmd` file is located there as well. Visual Editor will add the correct file name into `bibliography:`, but you will still need to select the style. In the YAML header, you will need to add the following lines and I want Chicago-style in-text citations:

```
bibliography: citations.bib
biblio-style: apsr
```

Style *apsr* is the style used in American Political Science Review, and this is the Chicago author-date style.

In case you need to use a very specific style that is not built-in, it will probably be available here: https://github.com/citation-style-language/styles. Styles are saved in `csl` files, so you will just need to download the file you need, put it in the project folder, and instead of `biblio-style` put `csl` parameter with the name of the respective file. For instance, if I wanted to use the style of *American Political Science Association*, I would write it like this if saved the `csl` file as `american-political-science-association.csl`:

```
bibliography: citations.bib
csl: american-political-science-association.csl
```

Step 3: Add *References* Section to the Document

The bibliography is typically placed at the end of the document, so your last heading should be something like `# References`.

Step 4: Reference Items in the Text

Now open the Visual Editor mode, and then click on: *Insert -> Citation* (or just use a shortcut `Ctrl + Shift + F8`/`Cmd + Shift+F8`). There, you can use the Crossref database and search by title (make sure to select the correct version of the text!) or search by DOI.

Once you found the text, select if you'd like it in format *Author (2000)* or *(Author 2000)* with *Use in-text citation* option and *Insert* the citation. Your bib entrance will be added to your `bib` file.

Citations go inside square brackets and are separated by semicolons. Each citation must have a key, composed of '@' + the citation identifier from the database, and may optionally have a prefix, a locator, and a suffix. Putting `[]` ensures that there are parenthesis around the citation.

```
Blah blah [see @doe99, pp. 33-35; also @smith04, ch. 1].

Blah blah [@smith04; @doe99].
```

A minus sign `-` before the `@` will suppress mention of the author in the citation. This can be useful when the author is already mentioned in the text and you only need to include the year:

```
Smith says blah [-@smith04].
```

This is how you get the in-text citations like *Smith (2004) says blah* and *Smith (2004, 33) says blah.*

```
@smith04 says blah

@smith04 [p. 33] says blah
```

Rstudio now also have a nice illustrated guide on the topic: https://rstudio.github.io/visual-markdown-editing/citations.html

Tables and Figures

Here is some general advice on how to make a good table and figures:

- Tables and figures should be clear, easily legible, and quickly understood by the reader
- Tables and figures should stand alone, and not require the reader to reference the text
- This requires a table/figure to minimally contain:
  - A title explaining the material concisely and clearly, with information about the outcome variable of other meaningful quantity of interest described

- Information on the sample time period and number of observations included in the graphic
- A note or notes that describe clearly what different cell entries or graphed material represents
- Meaningful variable names or labels, which clearly indicate meaning
- Clear and documented units of measurement
- Legends and captions that provide additional information when necessary

Let's look at an example:

Table 2: One-Sided Violence in Armed Conflict, 1989–2004

	Dependent variable:			
	Number killed in one-sided violence			
	(1)	(2)	(3)	(4)
Civil War	1.008***	1.015***	0.759**	1.042***
	(0.347)	(0.347)	(0.344)	(0.363)
Autocracy	0.888**	1.203***	0.780*	
	(0.419)	(0.437)	(0.415)	
Democracy	0.722	1.134**	0.486	
	(0.451)	(0.469)	(0.447)	
Polity Score				0.002
				(0.027)
Polity Score^2				−0.002
				(0.006)
Government	0.021	0.025	−0.041	−0.009
	(0.301)	(0.301)	(0.298)	(0.316)
One-sided Violence$_{t-1}$	0.010***	0.016***	0.011***	0.009***
	(0.001)	(0.004)	(0.001)	(0.001)
One-sided Violence$_{t-1}\times$ Autocracy		−0.007*		
		(0.004)		
One-sided Violence$_{t-1}\times$ Democracy		−0.011***		
		(0.004)		
Constant	0.939*	0.625	1.217**	1.716***
	(0.564)	(0.574)	(0.559)	(0.525)
Observations	1,159	1,159	1,157	1,102
Log Likelihood	−2,337.712	−2,336.735	−2,309.927	−2,172.654
θ	0.042*** (0.003)	0.042*** (0.003)	0.043*** (0.003)	0.040*** (0.003)
Akaike Inf. Crit.	4,687.425	4,689.470	4,631.853	4,357.308

Note: *p<0.1; **p<0.05; ***p<0.01
Standard errors in parenthesis. Excluding observation Rwanda 1994

Is there anything missing in this table?

Throwback Thursday/Monday/Tuesday

Remember your first lab exercises?

Exercise I

Create three objects:

1. `my_lucky_number` it should contain your lucky number
2. `my_firstname` it should contain your first name
3. `my_lastname` it should contain your last name

After you created the objects, call them separately. Don't forget to add comments to your code.

```
# We first create the objects
my_lucky_number <-
  my_firstname <-
  my_lastname <-
  # Now we want to call the objects
  my_lucky_number
my_firstname
my_lastname
```

Exercise II

Select and recode elements:

a) Create two vectors: `vec1` and `vec2`.

- `vec1` should contain 1, 56, 23, 89, -3 and 5 (in that order)
- `vec2` contains 24, 78, 32, 27, 8 and 1

b) Now select elements of `vec1` that are greater than 5 or smaller than 0

c) Next set `vec1` to zero if `vec2` is greater than 30 and smaller or equal to 32

Remember your first homework?

Quantitative Methods in Political Science - Homework 1

Your names go here (and percentages)

Due: September 21, 2021

Contents

Load the data set into R.

For this homework you will work with a data set on US Presidential Elections (`raw-data/uspresidentialelections.dta`). Since the data set comes in the `.dta` STATA format, you need to use the `foreign` library to load it into your environment. The following code chunk contains everything you need to successfully load the data into your environment.

```
library(foreign)
library(here)

# here() tells R to start looking for files in the project folder where .Rproj is
# Read the uspresidentialelections.dta file from the raw-data folder.
us_pres_data <- read.dta(here("raw-data/uspresidentialelections.dta"))
```

Now you have the object `us_pres_data` in your environment. It is your turn now to explore the data set and produce some interesting plots.

Describe the data set.

- What variables does it contain?
- How many observations are there?
- What time span does it cover?

Use the following code chunk to write code and explore the data. Please then write up your answers to the three questions after the code chunk.

The data set contains the following variables:

There are xxx observations in the data set.

It covers a time span from xxx to xxx.

Measures of central tendency and variability.

Compute measures of central tendency and variability of the variables `vote` and `growth` using the following code chunk. Use the numerical measures of central tendency and variability discussed in the lecture.

Describe the results in your own words and fill in the following table.

Measure	vote	growth
Mode		
Median		
Mean		
Variance		
Standard Deviation		
Range		
IQR		

Plot the distribution of both variables using a histogram and a density plot. Make sure to make your plots as nice-looking as possible. Especially, include a title and label the axes.

Use the following code chunk to produce all four plots.

```
library(viridis)

# Histogram for the variable vote


# Density plot for the variable vote


# Histogram for the variable growth


# Density plot for the variable growth
```

"It's the economy, stupid!"

During the presidential campaign in 1992, Bill Clinton's campaign coined the phrase "It's the economy, stupid!" Let's investigate the relationship between the economy and electoral success. Generate a nice-looking scatterplot of economic growth and vote share. Use the following code chunk to produce the scatterplot.

Describe the pattern that you see in your own words.

In the plot, I see ...

Styling your code

Once you are done with writing the code, may be a good idea to look over it and make sure it is well-formatted, readable, and commented out. While comments is something you would do when writing a code, little styling details such as extra spaces and line breaks may be hard to keep track of. Luckily, we have a package that allows us to format the code to follow a particular style very quickly. `styler`package formats the code you wrote to follow the guide we recommended you in the lab: https://style.tidyverse.org/.

We have already installed this package in our `setup` chunk, so now you can make use of it. The easiest way to do this would be with a so-called addin. Look for an Addin button in Rstudio and select *Style active file*. You can also select a few lines of code within a chunk and *Style the selection* to see the difference to your code better.

It took quite some time back then... How long do you think would it take you now?

It was so many commits ago!

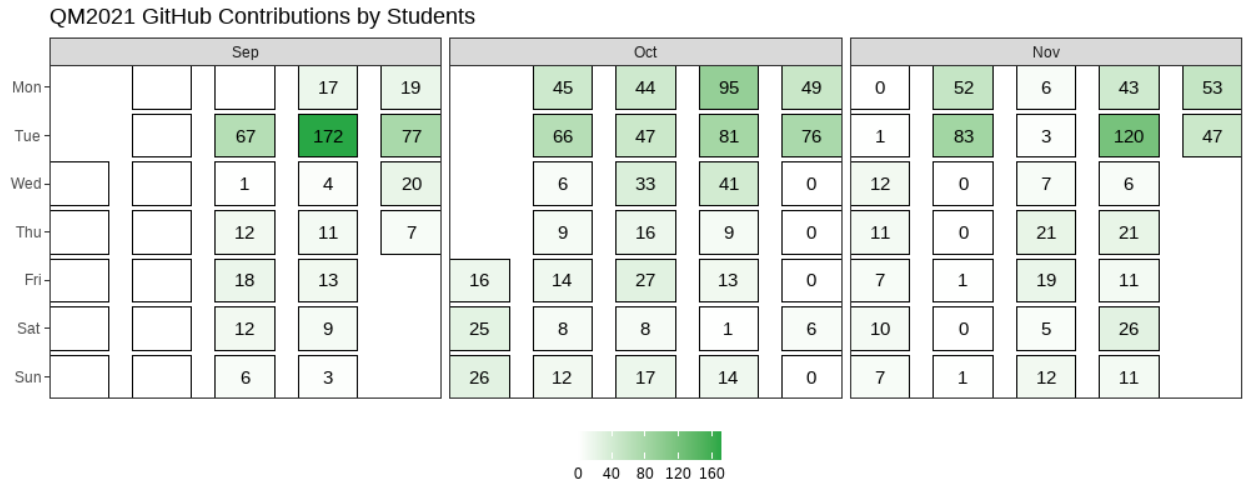


Figure 8: Github Activity in uni-mannheim-qm-2021 Organization

It's really amazing what you learnt this semester.

So we think you are well prepared to master the Data Essay!

References

Cheibub, José Antonio, Jennifer Gandhi, and James Raymond Vreeland. 2009. "Democracy and Dictatorship Revisited." *Public Choice* 143 (1-2): 67–101. https://doi.org/10.1007/s11127-009-9491-2.

Eck, Kristine, and Lisa Hultman. 2007. "One-Sided Violence Against Civilians in War: Insights from New Fatality Data." *Journal of Peace Research* 44 (2): 233–46.

Hilbe, Joseph M. 2009. "Negative Binomial Regression." https://doi.org/10.1017/cbo9780511973420.

King, Gary, Michael Tomz, and Jason Wittenberg. 2000. "Making the Most of Statistical Analyses: Improving Interpretation and Presentation." *American Journal of Political Science* 44 (2): 347. https://doi.org/10.2307/2669316.

Reuter, Ora John, and David Szakonyi. 2019. "Elite Defection Under Autocracy: Evidence from Russia." *American Political Science Review* 113 (2): 552–68. https://doi.org/10.1017/s0003055419000030.

Scheve, Kenneth, and Matthew J. Slaughter. 2004. "Economic Insecurity and the Globalization of Production." *American Journal of Political Science* 48 (4): 662–74. https://doi.org/10.1111/j.0092-5853.2004.00094.x.

Marking Scheme

	Max.	Points	Comment
Descriptive statistics	10		
Description of variables			
Presentation/Description			
Model Selection/Model Fit	25		
Justification			
- Chosen Model(s)			
- Selection of covariates			
Description			
- statistical significance			
Presentation			
- Table coef, se, loglik, N			
Model Fit			
Quantities of interest	25		
Some quantities of interest			
Sensible scenarios			
Substantial magnitude of findings			
Description			
Presentation			
Robustness	10		
Discussion of Robustness			
Robustness			
Conclusion	5		
Argument			
Limitations, further research			
R-Code	10		
Well-documented			
Runs Smoothly			
Overall Impression	15		
Presentation, Language, Coherence, Creativity			
Total			

Figure 9: Data Essay Marking