Models_AdrianBrenner_IR_ResearchDesign

Adrian Brenner

2023-12-13

This file is part of the IR Research Design Term Paper of Adrian Brenner.

```r
# This is the code for importing and cleaning the chosen Datasets.

if (!requireNamespace("readxl", quietly = TRUE)) {
  install.packages("readxl")
}
library(readxl)
```

```
## Warning: Paket 'readxl' wurde unter R Version 4.3.2 erstellt
```

```r
xlsx_file <- "C:/Users/AD/Desktop/data/BattleDeaths_v23_1.xlsx"


data <- read_xlsx(xlsx_file)

# In the first cleaning step I removed all conflict, not taking place in region = 4 (Africa)
new_data <- data[data$region == 4, ]
```

The second step involves removing all cases where there is only one belligerent on the second side, therefore not being supported by anyone else. Here the gwno classification is used, as it only picks up "real" countries in form of their country code. This will both be done for each warring side (in this Dataset represented by either side a or b).

```r
cleaned_data <- new_data[complete.cases(new_data$gwno_b_2nd), ]

cleaned_data_2 <- new_data[complete.cases(new_data$gwno_a_2nd), ]
```

The third step is to combine both data sets into one and remove the duplicates.

```r
library(dplyr)
```

```
##
## Attache Paket: 'dplyr'

## Die folgenden Objekte sind maskiert von 'package:stats':
##
##     filter, lag

## Die folgenden Objekte sind maskiert von 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Combine the datasets and remove duplicates
combined_data <- bind_rows(cleaned_data, cleaned_data_2) %>%
  distinct()
```

The final Dataset cleaned for the linear regression model of 4.2.1 now contains 238 distinct cases. I argue that I can proceed, as these are enough occurrences of foreign state intervention in conflict.

```
#TO create the opposite Dataset (the one with no intervention) I just subtract the combined_data from t
rest_data <- anti_join(data, combined_data)
```

```
## Joining with `by = join_by(conflict_id, dyad_id, location_inc, side_a,
## side_a_id, side_a_2nd, side_b, side_b_id, side_b_2nd, incompatibility,
## territory_name, year, bd_best, bd_low, bd_high, type_of_conflict,
## battle_location, gwno_a, gwno_a_2nd, gwno_b, gwno_b_2nd, gwno_loc, gwno_battle,
## region, version)`
```

In this next step I will create the regression model to include the battle related Deaths count of the combined_data Dataset in contrast to the count of the rest Dataset (no occurrence of classified state intervention).

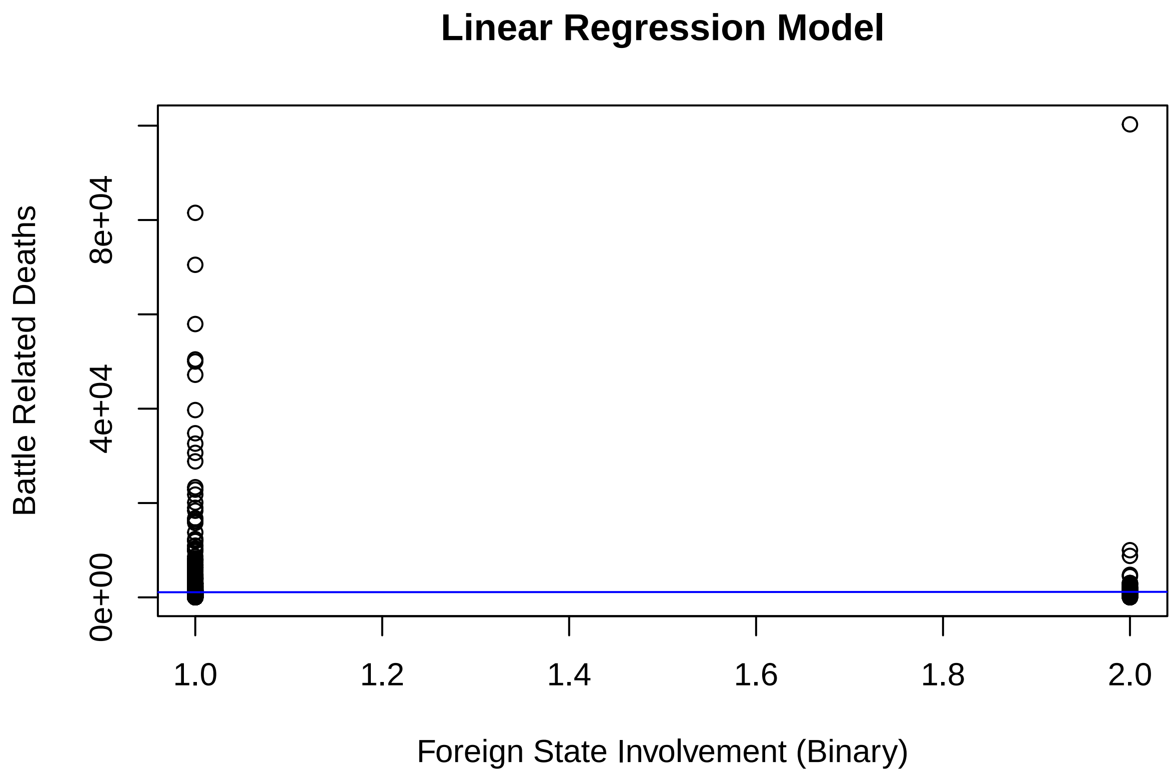
```
# Create a binary predictor variable
predictor_variable <- rep(c(0, 1), c(nrow(rest_data), nrow(combined_data)))

# Combine the datasets
all_data <- bind_rows(rest_data, combined_data)

# Create a linear regression model
model <- lm(bd_best ~ as.factor(predictor_variable), data = all_data)

# Plot the linear regression model
plot(all_data$bd_best ~ as.factor(predictor_variable),
     xlab = "Foreign State Involvement (Binary)",
     ylab = "Battle Related Deaths",
     main = "Linear Regression Model")

# Add the regression line to the plot
abline(model, col = "blue")
```




```r
# I now extract its coefficients, slope and intercept.

# Get coefficients of the linear regression model
coefficients <- coef(model)

# Extract slope and intercept
slope <- coefficients[2]  # Coefficient for the predictor variable
intercept <- coefficients[1]  # Intercept term

# Print the results
cat("Slope:", slope, "\n")
```

```
## Slope: 73.76704
```

```r
cat("Intercept:", intercept, "\n")
```

```
## Intercept: 1013.859
```

Here is a short summary of the models call.

```r
summary(model)
```

```
##
## Call:
## lm(formula = bd_best ~ as.factor(predictor_variable), data = all_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
```

```
##  -1062   -975   -898   -545  99194
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     1013.86     121.60   8.337   <2e-16 ***
## as.factor(predictor_variable)1    73.77     338.85   0.218    0.828
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4879 on 1846 degrees of freedom
## Multiple R-squared:  2.567e-05,  Adjusted R-squared:  -0.000516
## F-statistic: 0.04739 on 1 and 1846 DF,  p-value: 0.8277
```

Now for the T-test calculation.

```r
# T-test for the coefficient of the predictor variable
t_test <- summary(model)$coefficients["as.factor(predictor_variable)1", "t value"]

# Print the t-test result
cat("T-test for the coefficient of the predictor variable:", t_test, "\n")
```

```
## T-test for the coefficient of the predictor variable: 0.2176953
```

Now for the Chi-squared test calculation.

```r
# Extracting the residual standard error
residual_standard_error <- summary(model)$sigma

# Degrees of freedom
df <- length(model$coefficients) - 1  # Subtract 1 for the intercept term

# Chi-square test statistic
chi_square <- (t_test^2) / residual_standard_error^2

# P-value for chi-square test
p_value_chi_square <- 1 - pchisq(chi_square, df)

# Print the chi-square test results
cat("Chi-square test statistic:", chi_square, "\n")
```

```
## Chi-square test statistic: 1.990538e-09
```

```r
cat("Degrees of freedom:", df, "\n")
```

```
## Degrees of freedom: 1
```

```r
cat("P-value for chi-square test:", p_value_chi_square, "\n")
```

```
## P-value for chi-square test: 0.9999644
```

Adrian Brenner (12422308) for GSI LMU - IR Research Design (15103) - Dr. Oliver Pamp