Conducting EDA on Corruption Data

Andre Fernandes, Keenan Szulik, and Erik Hou

09/25/2018

Introduction

This analysis is motivated by the following research question:

"Is there any relationship between corruption and parking violations both before and after 2002? If so, are there any other relevant explanatory variables?"

The question will be addressed by exploratory data analysis techniques. We have been asked to imagine that we were hired by the World Bank to explore the influence cultural norms and legal enforcement have on controlling corruption. To operationalize the analysis, the assignment looks at the parking behavior of United Nations officials in Manhattan.

Until 2002, UN diplomats were protected by diplomatic immunity: they were not subject to parking enforcement actions and their actions were solely constrained by cultural norms. In 2002, the parking authority acquired the right to confiscate diplomatic license plates of the violators. As a result, their parking behavior was constrained by both cultural norms and the legal penalties.

We have been given a dataset of a selection of UN diplomatic missions, which includes a target variable, *violations*; a label for before and after the parking enforcement change, *prepost*; and other essential variables, like corruption index and continent regions.

Setup

First, we load some of the packages we will need for the analysis and the data into R.

Load the data:

```
source("utils/functions.R")

## Loading required package: pacman

df <- load_rda('data/Corrupt.Rdata')
```

Overview of the data structure

We have 364 observations.

```
nrow(df)

## [1] 364
```

We have 28 variables in the dataset:

```
str(df)

## 'data.frame':    364 obs. of  28 variables:
##  $ wbcode        : chr  "AFG" "AGO" "AGO" "ALB" ...
##  $ prepost       : chr  "" "pre" "pos" "pre" ...
##  $ violations    : num  NA 744.38 15.37 256.63 5.56 ...
##  $ fines         : num  NA 40294 1208 13970 610 ...
##  $ mission       : int  NA 1 1 1 1 1 1 1 1 1 ...
##  $ staff         : int  NA 9 9 3 3 3 3 19 19 4 ...
##  $ spouse        : int  NA 4 4 3 3 2 2 10 10 1 ...
##  $ gov_wage_gdp  : num  NA 1.3 1.3 1.3 1.3 ...
##  $ pctmuslim     : num  NA 0.01 0.01 0.7 0.7 ...
##  $ majoritymuslim: int  NA 0 0 1 1 1 1 0 0 -1 ...
##  $ trade         : num  NA 2.61e+09 2.61e+09 2.72e+07 2.72e+07 ...
##  $ cars_total    : int  NA 24 24 4 4 13 13 15 15 3 ...
##  $ cars_personal : int  NA 3 3 0 0 6 6 14 14 1 ...
##  $ cars_mission  : int  NA 21 21 4 4 7 7 1 1 2 ...
##  $ pop1998       : num  NA 11739390 11739390 3101330 3101330 ...
##  $ gdppcus1998   : num  NA 731 731 1008 1008 ...
##  $ ecaid         : num  NA 92.3 92.3 62.8 62.8 ...
##  $ milaid        : num  NA 0 0 2.2 2.2 ...
##  $ region        : int  NA 6 6 3 3 7 7 2 2 4 ...
##  $ corruption    : num  NA 1.048 1.048 0.921 0.921 ...
##  $ totaid        : num  NA 92.3 92.3 65 65 ...
##  $ r_africa      : int  NA 1 1 0 0 0 0 0 0 0 ...
##  $ r_middleeast  : int  NA 0 0 0 0 1 1 0 0 0 ...
##  $ r_europe      : int  NA 0 0 1 1 0 0 0 0 0 ...
##  $ r_southamerica: int  NA 0 0 0 0 0 0 1 1 0 ...
##  $ r_asia        : int  NA 0 0 0 0 0 0 0 0 1 ...
##  $ country       : chr  "AFGANISTAN" "ANGOLA" "ANGOLA" "ALBANIA" ...
##  $ distUNplz     : num  0.445 1.554 1.554 1.775 1.775 ...
```

Look at the summary of the dataset:

```
summary(df)
```

```
##     wbcode            prepost            violations
##  Length:364         Length:364         Min.   :   0.000
##  Class :character   Class :character   1st Qu.:   0.654
##  Mode  :character   Mode  :character   Median :   5.724
##                                        Mean   : 100.879
##                                        3rd Qu.:  51.915
##                                        Max.   :3392.961
##                                        NA's   :66
##      fines              mission           staff           spouse
##  Min.   :     0.00   Min.   :0.0000   Min.   : 0.00   Min.   : 0.000
##  1st Qu.:    65.41   1st Qu.:1.0000   1st Qu.: 5.00   1st Qu.: 3.000
##  Median :   579.72   Median :1.0000   Median : 9.00   Median : 5.000
##  Mean   :  5579.60   Mean   :0.9868   Mean   :11.65   Mean   : 7.656
##  3rd Qu.:  2999.05   3rd Qu.:1.0000   3rd Qu.:14.00   3rd Qu.:10.000
##  Max.   :186163.17   Max.   :1.0000   Max.   :86.00   Max.   :81.000
##  NA's   :66          NA's   :62       NA's   :62      NA's   :62
##   gov_wage_gdp      pctmuslim      majoritymuslim         trade
##  Min.   : 0.100   Min.   :0.0000   Min.   :-1.0000   Min.   :0.000e+00
##  1st Qu.: 1.300   1st Qu.:0.0060   1st Qu.: 0.0000   1st Qu.:9.532e+07
##  Median : 1.900   Median :0.0500   Median : 0.0000   Median :5.443e+08
##  Mean   : 2.828   Mean   :0.2766   Mean   : 0.2416   Mean   :1.034e+10
##  3rd Qu.: 3.625   3rd Qu.:0.5400   3rd Qu.: 1.0000   3rd Qu.:4.904e+09
##  Max.   :11.800   Max.   :0.9990   Max.   : 1.0000   Max.   :3.290e+11
##  NA's   :180      NA's   :66       NA's   :66        NA's   :68
##    cars_total     cars_personal     cars_mission       pop1998
##  Min.   :  1.00   Min.   : 0.000   Min.   :  0.000   Min.   :5.308e+05
##  1st Qu.:  3.00   1st Qu.: 1.000   1st Qu.:  2.000   1st Qu.:3.879e+06  
##  Median :  7.00   Median : 2.000   Median :  3.000   Median :9.488e+06  
##  Mean   : 10.47   Mean   : 5.324   Mean   :  5.144   Mean   :1.174e+08  
##  3rd Qu.: 12.00   3rd Qu.: 6.000   3rd Qu.:  6.000   3rd Qu.:3.019e+07  
##  Max.   :116.00   Max.   :64.000   Max.   :116.000   Max.   :5.900e+09  
##  NA's   :86       NA's   :86       NA's   :86        NA's   :42         
##   gdppcus1998           ecaid             milaid            region     
##  Min.   :   95.45   Min.   :   0.00   Min.   :   0.00   Min.   :1.000  
##  1st Qu.:  418.20   1st Qu.:   0.00   1st Qu.:   0.00   1st Qu.:3.000  
##  Median : 1430.67   Median :   8.75   Median :   0.20   Median :4.000  
##  Mean   : 5236.42   Mean   :  49.44   Mean   :  32.85   Mean   :4.347  
##  3rd Qu.: 5132.01   3rd Qu.:  40.70   3rd Qu.:   0.80   3rd Qu.:6.000  
##  Max.   :36485.64   Max.   :1026.10   Max.   :3120.00   Max.   :7.000  
##  NA's   :42         NA's   :68        NA's   :68        NA's   :64     
##    corruption           totaid            r_africa       r_middleeast    
##  Min.   :-2.58299   Min.   :   0.000   Min.   :0.0000   Min.   :0.00000  
##  1st Qu.:-0.46186   1st Qu.:   0.375   1st Qu.:0.0000   1st Qu.:0.00000  
##  Median : 0.32292   Median :   9.100   Median :0.0000   Median :0.00000  
##  Mean   :-0.00932   Mean   :  82.293   Mean   :0.2857   Mean   :0.09317  
##  3rd Qu.: 0.71516   3rd Qu.:  43.000   3rd Qu.:1.0000   3rd Qu.:0.00000  
##  Max.   : 1.58281   Max.   :4069.100   Max.   :1.0000   Max.   :1.00000  
##  NA's   :61         NA's   :68         NA's   :42       NA's   :42       
##     r_europe      r_southamerica       r_asia          country         
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Length:364         
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   Class :character   
##  Median :0.0000   Median :0.0000   Median :0.0000   Mode  :character   
##  Mean   :0.2174   Mean   :0.1118   Mean   :0.1615                      
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000                      
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000                      
##  NA's   :42       NA's   :42       NA's   :42                          
##    distUNplz      
##  Min.   : 0.0000  
##  1st Qu.: 0.2218  
##  Median : 0.2956  
##  Mean   : 0.5864  
##  3rd Qu.: 0.4610  
##  Max.   :15.0552  
##  NA's   :33
```

Examine the first ten rows of the dataset:

```
head(df, 10)
```

```
##    wbcode prepost violations      fines mission staff spouse gov_wage_gdp
## 1     AFG               NA         NA      NA    NA     NA           NA
## 2     AGO     pre 744.381226 40293.8125       1     9      4          1.3
## 3     AGO     pos  15.371863  1208.4901       1     9      4          1.3
## 4     ALB     pre 256.634308 13970.0615       1     3      3          1.3
## 5     ALB     pos   5.560036   609.9686       1     3      3          1.3
## 6     ARE     pre   0.000000     0.0000       1     3      2           NA
## 7     ARE     pos   0.000000     0.0000       1     3      2           NA
## 8     ARG     pre  75.957268  4106.7563       1    19     10          2.4
## 9     ARG     pos   6.868279   695.0045       1    19     10          2.4
## 10    ARM     pre  40.915649  1986.0294       1     4      1          0.8
##    pctmuslim majoritymuslim      trade cars_total cars_personal
## 1         NA             NA         NA         NA            NA
## 2      0.010              0 2605844736         24             3
## 3      0.010              0 2605844736         24             3
## 4      0.700              1   27228056          4             0
## 5      0.700              1   27228056          4             0
## 6      0.760              1 3030428160         13             6
## 7      0.760              1 3030428160         13             6
## 8      0.015              0 8137429504         15            14
## 9      0.015              0 8137429504         15            14
## 10     0.000             -1   68119280          3             1
##    cars_mission  pop1998 gdppcus1998 ecaid milaid region corruption totaid
## 1            NA       NA          NA    NA     NA     NA         NA     NA
## 2            21 11739390    731.2249  92.3    0.0      6  1.0475056   92.3
## 3            21 11739390    731.2249  92.3    0.0      6  1.0475056   92.3
## 4             4  3101330   1008.3250  62.8    2.2      3  0.9210790   65.0
## 5             4  3101330   1008.3250  62.8    2.2      3  0.9210790   65.0
## 6             7  2834000  21143.5391    NA     NA      7 -0.7794677     NA
## 7             7  2834000  21143.5391    NA     NA      7 -0.7794677     NA
## 8             1 36005390   8234.9307   0.0    1.2      2  0.2235667    1.2
## 9             1 36005390   8234.9307   0.0    1.2      2  0.2235667    1.2
## 10            2  3181000    548.8061  93.0    0.0      4  0.7100782   93.0
##    r_africa r_middleeast r_europe r_southamerica r_asia    country
## 1        NA           NA       NA             NA     NA AFGANISTAN
## 2         1            0        0              0      0     ANGOLA
## 3         1            0        0              0      0     ANGOLA
## 4         0            0        1              0      0    ALBANIA
## 5         0            0        1              0      0    ALBANIA
## 6         0            1        0              0      0
## 7         0            1        0              0      0
## 8         0            0        0              1      0  ARGENTINA
## 9         0            0        0              1      0  ARGENTINA
## 10        0            0        0              0      1    ARMENIA
##    distUNplz
## 1  0.4451198
## 2  1.5536108
## 3  1.5536108
## 4  1.7754116
## 5  1.7754116
## 6  0.3338862
## 7  0.3338862
## 8  0.0000000
## 9  0.0000000
## 10 0.5775134
```

Data Selection and Cleaning

From examining the data summary and the first ten rows, we see many NA values in the key variable fields, such as violations and corruption. Also notice that, in the field prepost, we have blanks.

It is necessary to clean the data by taking out the records with the blank or NA essential fields before starting analysis on relevant variables:

```
df[df=="" | df=="NA"] = NA  #set all the blanks and "NA" to NA

#exclude the records having NAs in at least one of the essential fields
df_clean = subset(df, !is.na(wbcode) & !is.na(prepost) & !is.na(violations) &
!is.na(corruption))
```

Do another count row, we note that the number of observations dropped in the cleaning process was 66

```
nrow(df)-nrow(df_clean) #number of observations dropped when cleaning data

## [1] 66
```

One last step before starting univariate analysis of key variables is to make sure that, in our cleaned dataset, we have exactly two records per country, one before and one after 2002. Because,

1. With a missing *pre* or *pos* record, it would be difficult to compare country behavior pre and post the policy change.
2. If the dataset includes countries that had more than one pre and/or one post record, further cleaning or manipulation would be required to appropriately weigh different observations.

```
length(unique(df_clean[df_clean$prepost == "pre",]$wbcode))  #the total number of
distinct countries in the data set with prepost == "pre"

## [1] 149

length(unique(df_clean[df_clean$prepost == "pos",]$wbcode))  #the total number of
distinct countries in the data set with prepost == "pos"

## [1] 149

length(unique(df_clean$wbcode))  #the total number of unique countries

## [1] 149

nrow(df_clean)  #the total number observations in the data set

## [1] 298
```

From the above counts, we note that, in the dataset, 149 unique coutries having *prepost* field as *pre* and 149 unique countries having prepost field being *pos*. Because we know that we have exactly 149 different countries in the dataset, so we can conclude, we have 149 unique countries with each of them having one pre and one post record. There are no duplicates because the 2 times 149 is equal to the total observation count of 298.

Finally, we have made sure our dataset is clean enough for our subseqent analysis.

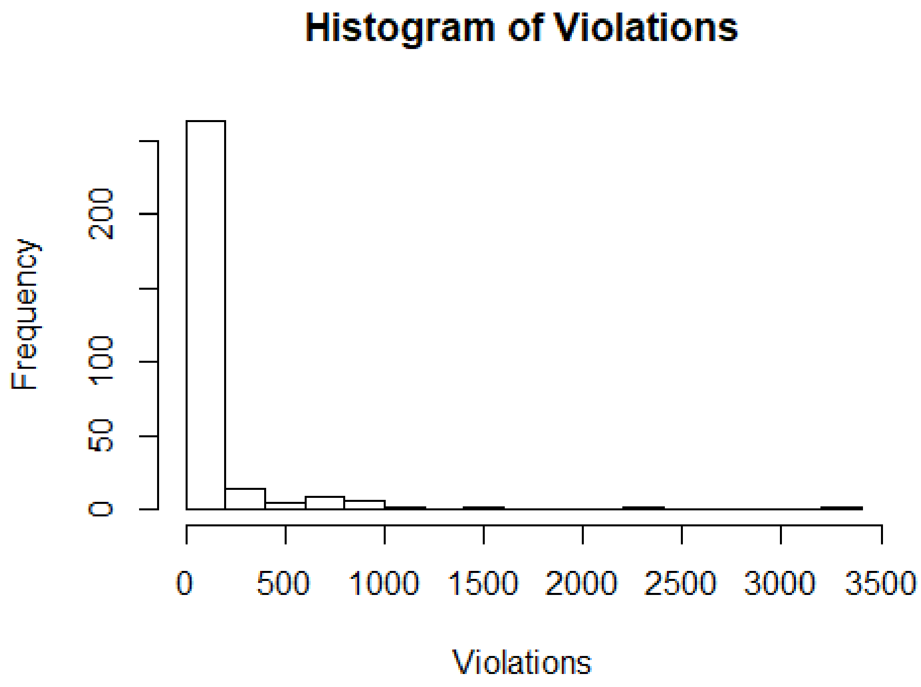
Univariate Analysis of Key Variables

Now we start the univariate analysis.

Target Variable: Violations

Let’s look at the target variable **violations**:

```
summary(df_clean$violations)

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##    0.000    0.654    5.724  100.879   51.915 3392.961

sd(df_clean$violations)

## [1] 302.2331

hist(df_clean$violations,20,xlab = "Violations", main = "Histogram of Violations")
```

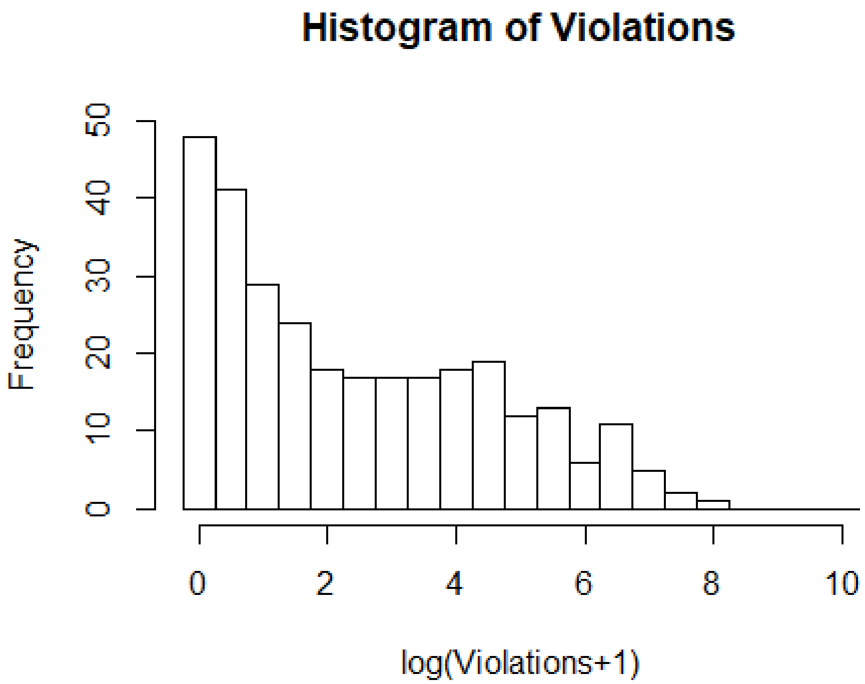


There are several features of the variables worth highlighting:

1. All the values are non-negative.
2. From both the histogram and the numeric summary, we can see that the values are very clustered on the lower end: more than 50% of the values are less than 6.
3. The distribution is right-skewed, with large outliers. This causes the mean to be greater than the median and a high standard deviation, around 302.

Since there are 48 violation data points with the value of zero, as well as outliers–a histogram of $log(volations + 1)$ will help us better visualize the distribution. While drawing the histogram, we adjust the positions of the bins so the first bar is centered around zero.

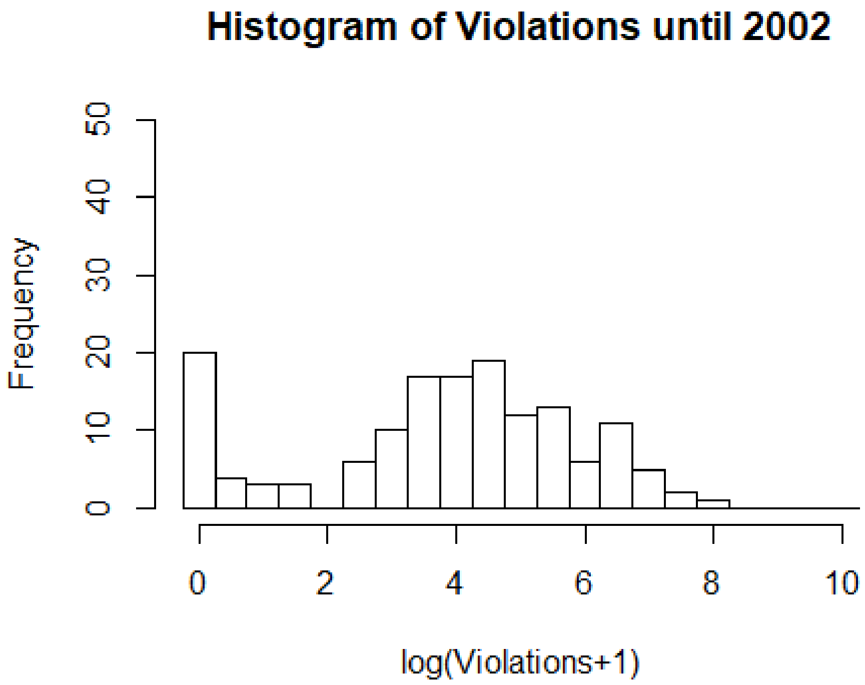
```r
nrow(df_clean[df_clean$violations == 0,])

## [1] 48

hist(log(df_clean$violations+1),breaks = seq(-0.75,10,0.5)+0.5, ylim = c(0,50), xlab =
"log(Violations+1)", main = "Histogram of Violations")
```



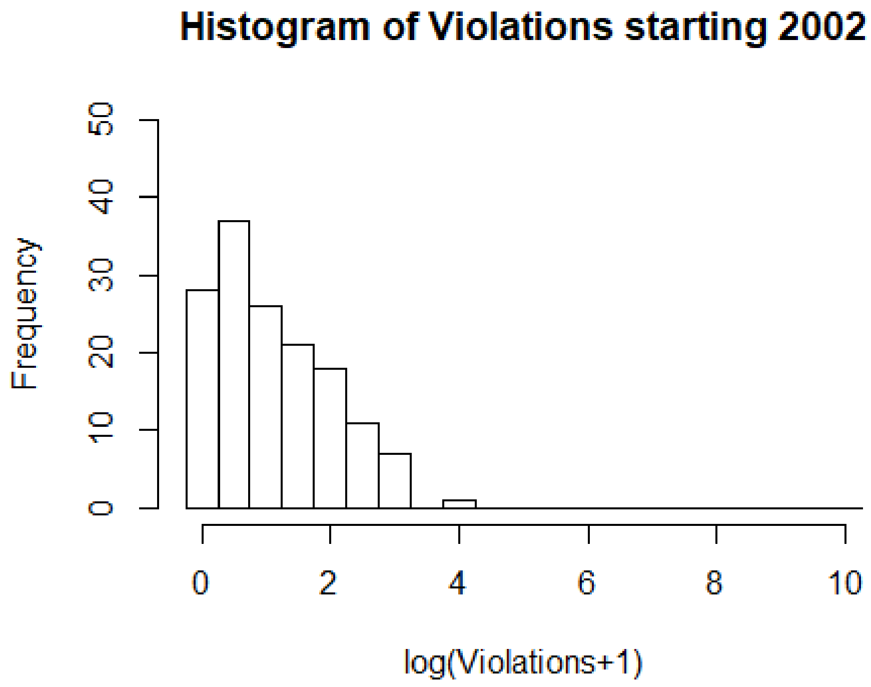
Notice that the frequency distribution appears to have two local peaks, one in (-0.25,0.25) and one in (4.25,4.75). This probably is caused by the change of policy where there were more violations before 2002 and less violation after 2002. We will inspect this further.

Two histograms of violations before and after the change of policy prove the assumption.

```r
hist(log(df_clean[df_clean$prepost == "pre",]$violations+1),breaks = seq(-
0.75,10,0.5)+0.5, ylim = c(0,50), xlab = "log(Violations+1)", main = "Histogram of
Violations until 2002")
```



```
hist(log(df_clean[df_clean$prepost == "pos",]$violations+1),breaks = seq(-
0.75,10,0.5)+0.5, ylim = c(0,50), xlab = "log(Violations+1)", main = "Histogram of
Violations starting 2002")
```



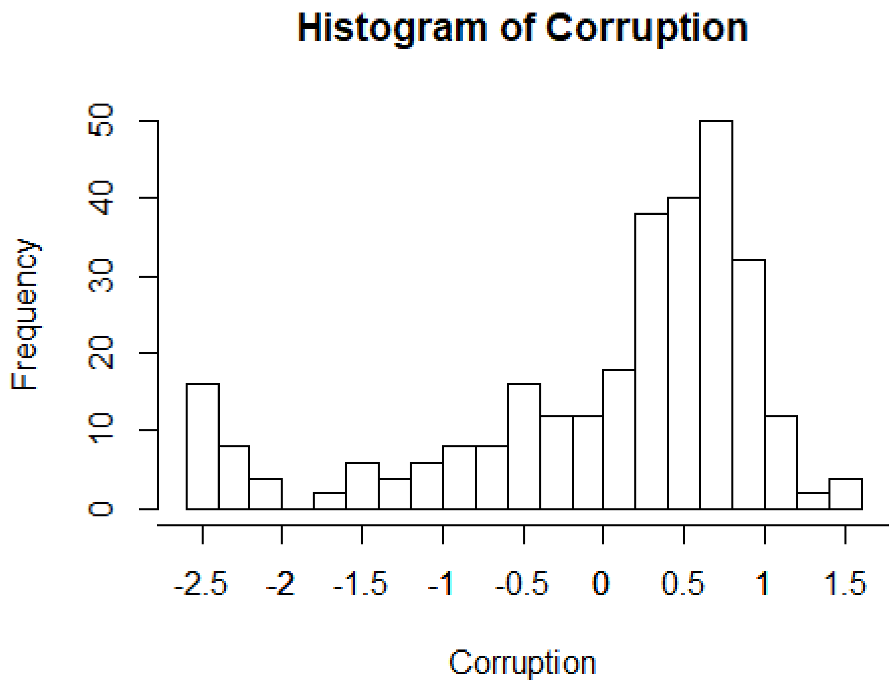
From the above graphic, we demonstrated the shift in the distribution of violation before and after the policy change.

Variable: Corruption

Next, we move on to the other key variable, corruption.

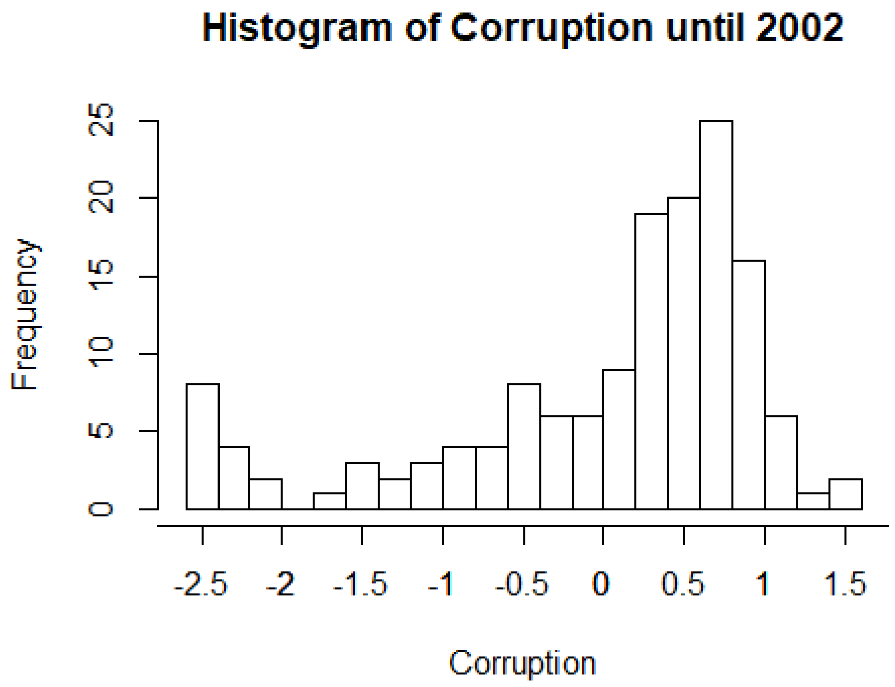
First step is to look at the numeric summary of the variable and its histogram:

```
summary(df_clean$corruption)

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -2.58299 -0.41515  0.32696  0.01364  0.72025  1.58281

sd(df_clean$corruption)

## [1] 1.012474

hist(df_clean$corruption, breaks = 20, xlab = "Corruption", main = "Histogram of
Corruption")
axis(1, at = seq(-3,2,by=0.5), labels = seq(-3,2,by=0.5))
```

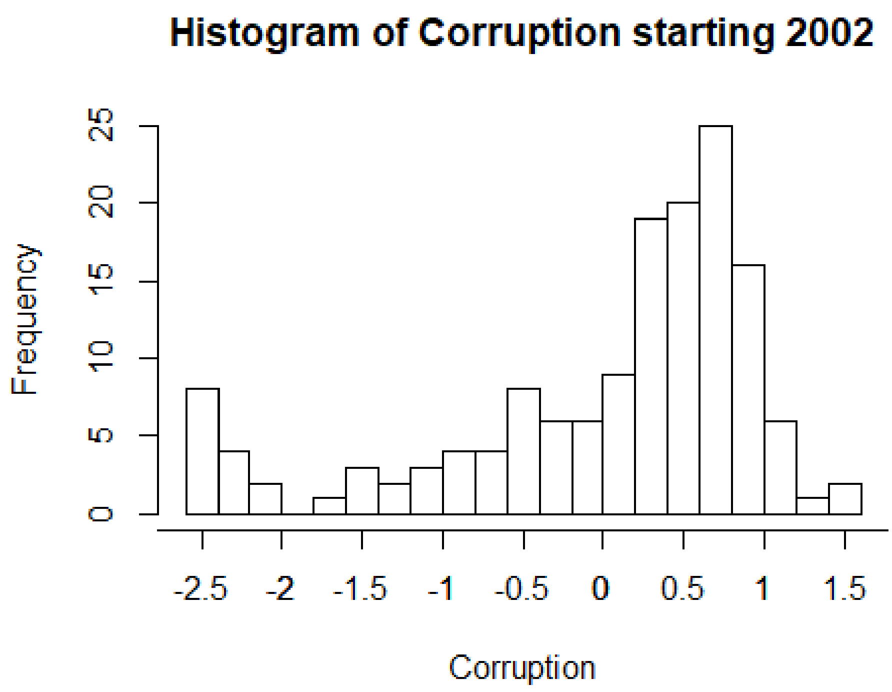


We draw the histograms of corruption of pre and post the policy change separately to compare:

```
hist(df_clean[df_clean$prepost == "pre",]$corruption, breaks = 20, xlab = "Corruption",
main = "Histogram of Corruption until 2002")
axis(1, at = seq(-3,2,by=0.5), labels = seq(-3,2,by=0.5))
```



```
hist(df_clean[df_clean$prepost == "pos",]$corruption, breaks = 20, xlab = "Corruption",
main = "Histogram of Corruption starting 2002")
axis(1, at = seq(-3,2,by=0.5), labels = seq(-3,2,by=0.5))
```



Note that the two histograms appear identical, which likely means that this dataset treats corruption as a constant variable over time. To further check, we test if for each country the corruption is the same pre and post 2002.

```
nrow(unique(df_clean[,c("wbcode", "corruption")]))

## [1] 149
```

Several key features of the variable *corruption*:

1. Through making sure that the number of unique combinations of *wbcode* and *corruption* is the same as the number of unique wbcodes, we are sure that the dataset has corruption as a constant for each country over time.
2. We can see that the histogram appears to have two local peaks: one at around 0.75 and another at around -2.5.
3. This distribution is left-skewed, with most countries having the values between 0 and 1.
4. Some outliers cluster close to the lower local peak at -2.5.

Analysis of Key Relationships

In this section, we will be conducting multivariate analysis on our corruption data. This section will be divided into two segments comprised of **correlations for numerical variables** and a deeper dive into variable relationships while observing the **prepost** variable.

Analyzing correlations among continuous variables

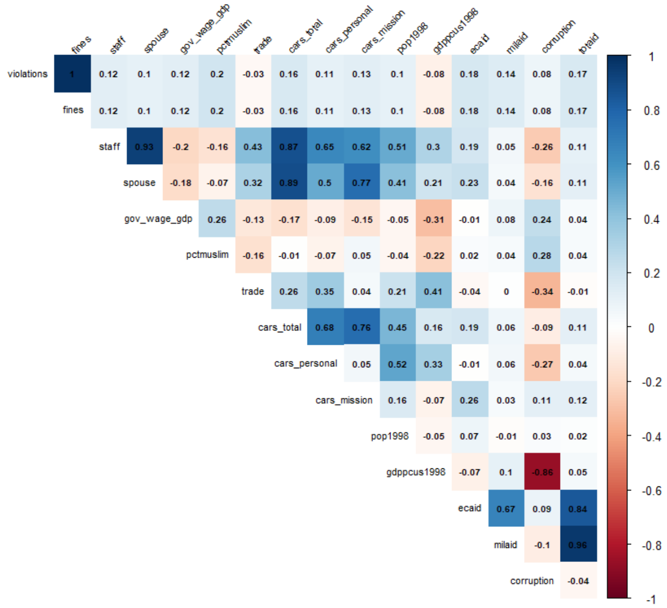
All line items

Here, we do a quick look at correlation of the numerical variables. It appears that most variables in this dataset is static between the two snapshot dates (as mentioned above).

1. Violations and fines are perfectly correlated, so appear to they tell us the same information about the data. We will check to see if they are identical later in this analysis.
2. Since we know that the *prepost* variable captures a time element, we will be viewing the correlations in each of the *pre* and *pos* subset groups.
3. Staff, Spouse, and Car Total have a high positive correlation, which makes sense when we think of the semantic meaning of these variables.
4. GDP and Corruption have a very high negative correlation. This is not a surprising given the semantic definitions.
5. There are many other relationships to look at, but for the sake of brevity, we will end the analysis into the combined data correlations here.

The plot_correlation function was moved to the functions.R file. In it, we use the ***cor function*** *with the following parameter* ***use="pairwise.complete.obs"*** *in order to drop the minimal number of observations when computing the correlations between two variables. Also, I only calculate the correlation for variables with the attribute* ***is.numeric(x)==TRUE*** *AND* ***length(unique(x))>10****, to keep factor variables from being calculated as numeric, AND we exclude the variable* ***distUNplz*** *since we could not figure out its definition with the provided information.*

```
# all lines
plot_correlation(df_clean, 10)
```

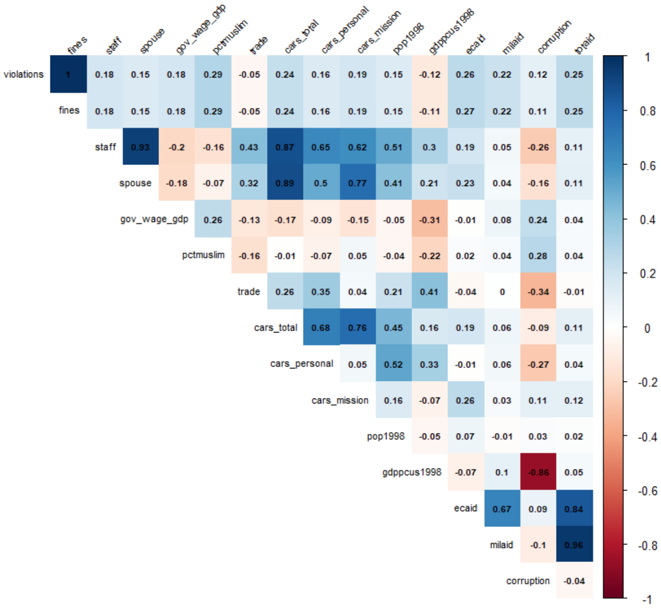


Pre and Post 2002

1. The interesting observation from comparing **pre** and **post** 2002 correlations is that **milaid** and **totaid** show slight positive correlations with **violations** and **fines** pre 2002. Meanwhile, during the **post 2002** period, those correlations have a near zero *R value*.
2. There's a shortage of information about the dataset and how variables are captured and defined, so we will not be able to explain why the correlations are so different between the aforementioned variables during the two snapshots of time.

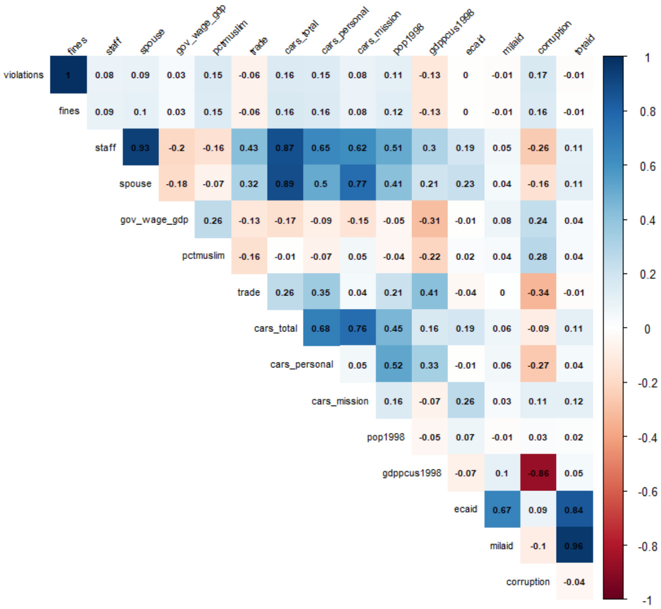
Here, we will subset out data to *prepost == 'pre'*.

```
# pre lines
plot_correlation(df_clean %>% filter(prepost=='pre'), 10)
```



Here, we will subset out data to *prepost == 'pos'*.

```
# post lines
plot_correlation(df_clean %>% filter(prepost=='pos'), 10)
```

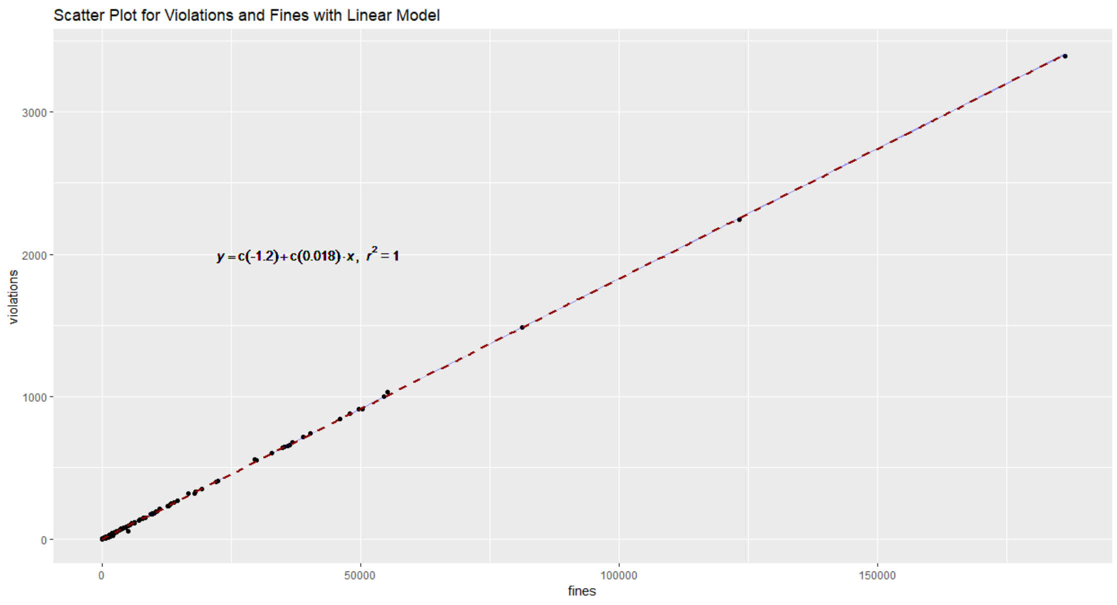


Looking further into the relationship between violations and fines

We observed a very strong correlation between **violations** and **fines**, so now let's plot a scatterplot with these two variables and add the bet fit line. Please note that we saw the correlation metrics prior to even adding a log transformation, so we will keep the variables as is for this graph.

The variables indeed appear to be perfectly correlated. This makes sense since fines are likely violations multiplied by a scalar.

```
# regular
plot_scatter <- ggplot(df_clean, aes(x=fines, y=violations)) +
                    geom_point()+ geom_smooth(method=lm,  linetype="dashed",
color="darkred", fill="blue")+
                    geom_text(x = 40000, y = 2000, label = lm_eqn(df_clean, 'fines',
'violations'), parse = TRUE)+
                    labs(title = "Scatter Plot for Violations and Fines with Linear 
Model")
plot(plot_scatter)
```



Deep dive into variable relationships with violations while considering the prepost 2002 timestamp

In this section, we will analyze variables that we believe are important. We will not cover all variables for brevity.

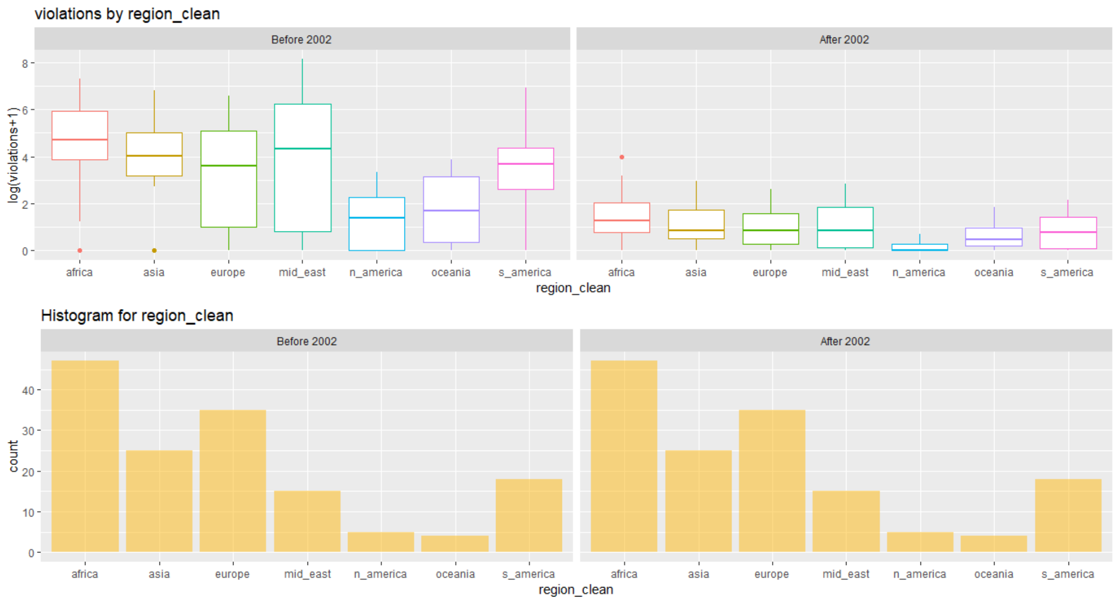
Region

One of the key categorical variables to observe is **region**. Although the given dataset only provides an integer factor for the regions, we were able to use the regional indicators to map the integer factors back to actual region names. Also, please note that the African country of **Zaire** had a missing value for **region**. Since we were able to do a quick internet search and discovered that **Zaire** is in **Africa**, we altered the record when creating the **region_clean** variable.

What we were able to observe:

1. North America region has the lowest mean **log(violations+1)** before and after 2002.
2. Middle East region has the largest interquartile range before and after 2002.
3. The target variable is much lower after 2002 (as we would expect since countries are now paying their fines, which lowers the total unpaid level)

```
df_clean$region_clean <- ifelse(is.na(df_clean$region),6,df_clean$region)
df_clean$region_clean <- as.factor(df_clean$region_clean)
levels(df_clean$region_clean) <- c('n_america', 's_america', 'europe', 'asia', 'oceania', 
'africa', 'mid_east')

plot_vars(df_clean, 'violations', 'region_clean', 'cat', 'prepost')
```



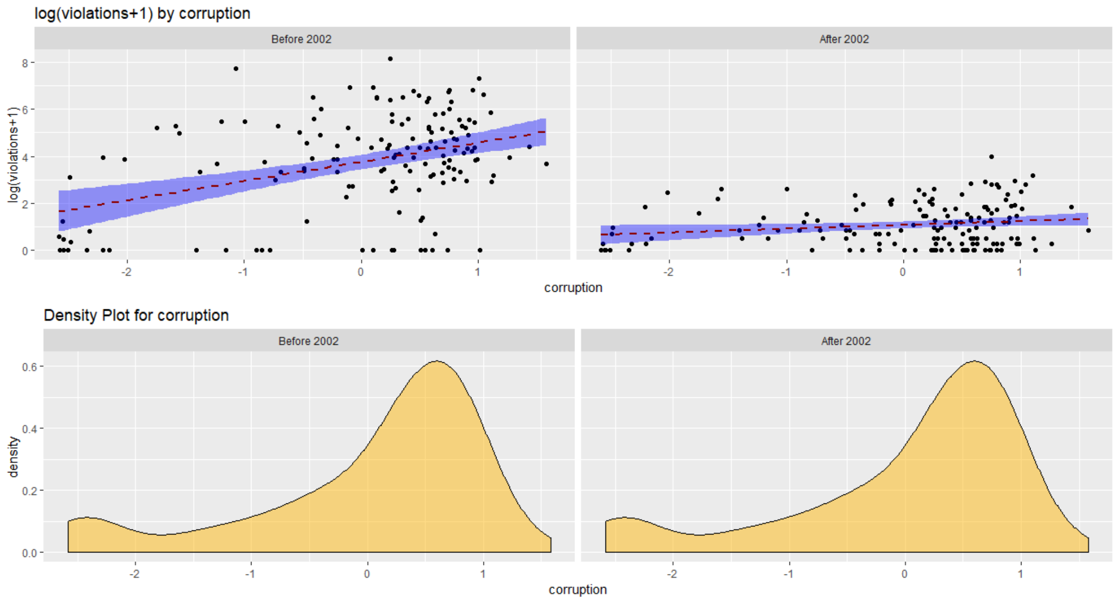
Corruption

One of the key numerical variables to observe is **corruption.**

What we were able to observe:

1. Corruption has a more positive relationship with the target variable before 2002 than after.
2. The value for the corruption variable is static, so they are not different between the two snapshots in time. It is strange to want this variable to be static since we would expect that the corruption index changes over time.

```
plot_vars(df_clean, 'violations', 'corruption', 'numeric', 'prepost')
```



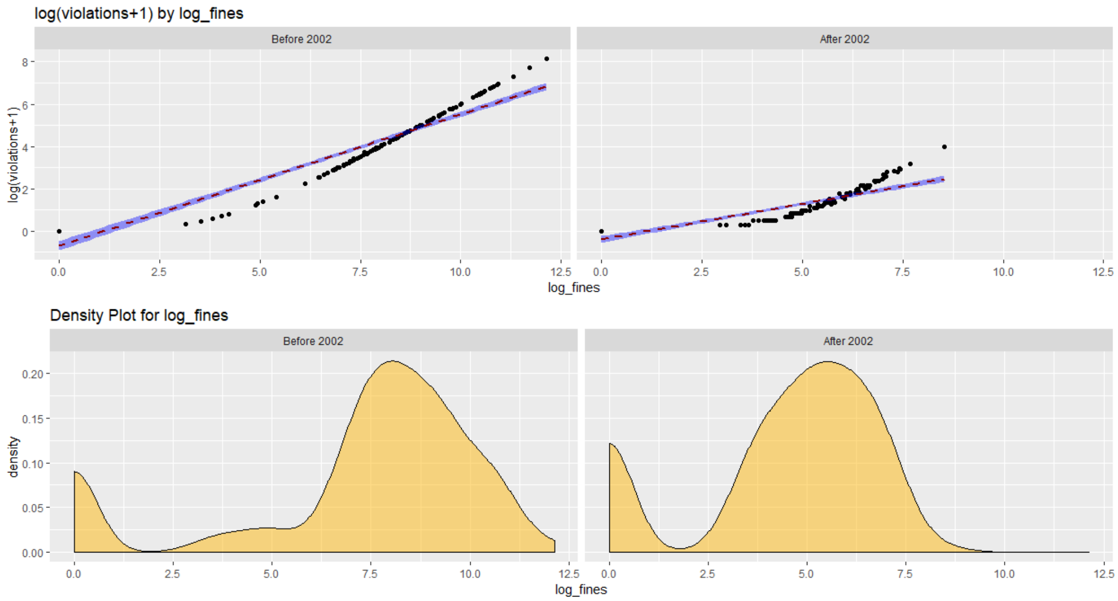
Fines

Another key numerical variables to observe is **fines**. We already plotted the relationship between violations and fines earlier, so here we take a look at the relationship between the log transformations of the two variables.

What we were able to observe:

1. There still remains a clear relationship between the log transformations of the two variables.
2. The relationships appear to be less linear than before, but the linear estimator is still okay in this case.
3. If a predictive model were built to predict violations, this is a classic example of data leakage. If you attempt to use **fines** as a predictor for **violations**, you would not have **fines** information at the time of prediction. This means that this variable is only available in our data because we are looking at historical records.

```
df_clean$log_fines <- log(df_clean$fines +1)
plot_vars(df_clean, 'violations', 'log_fines', 'numeric', 'prepost')
```

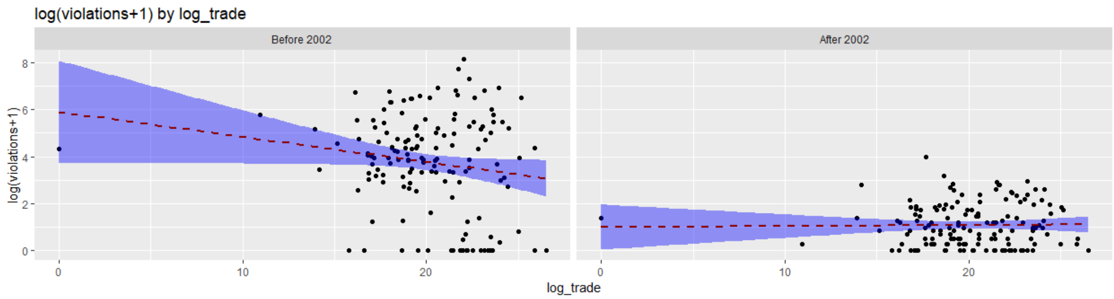


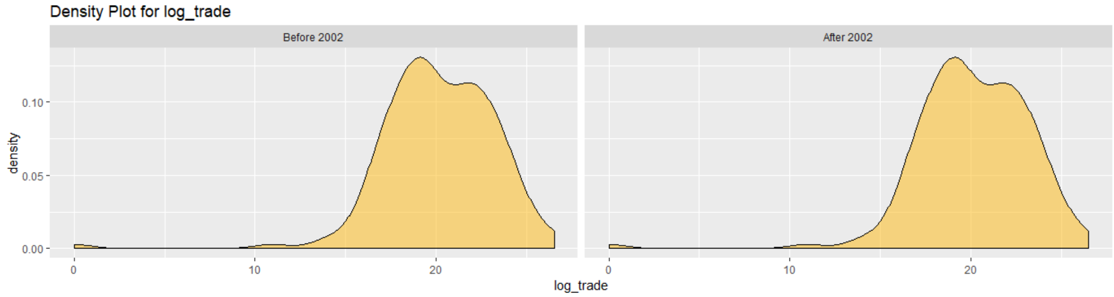
Trade

What we were able to observe:

1. The relationship between trade and the log transformation of violation is all over the place. We would not trust the linear models in this case.

```
df_clean$log_trade <- log(df_clean$trade+1)
plot_vars(df_clean, 'violations', 'log_trade', 'numeric', 'prepost')

## Warning: Removed 4 rows containing non-finite values (stat_smooth).

## Warning: Removed 4 rows containing missing values (geom_point).

## Warning: Removed 4 rows containing non-finite values (stat_density).
```



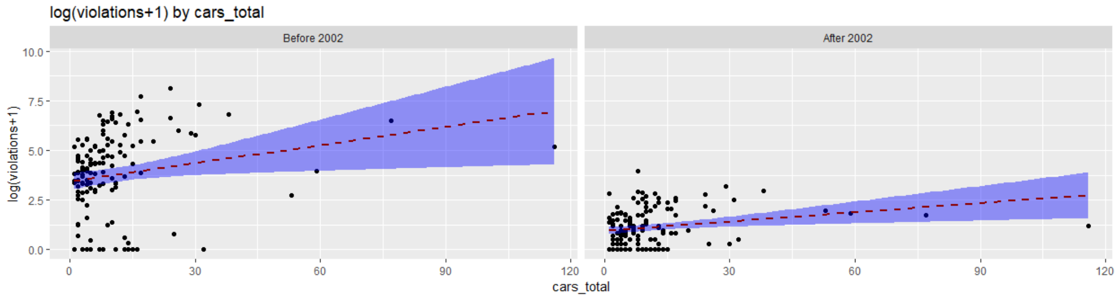


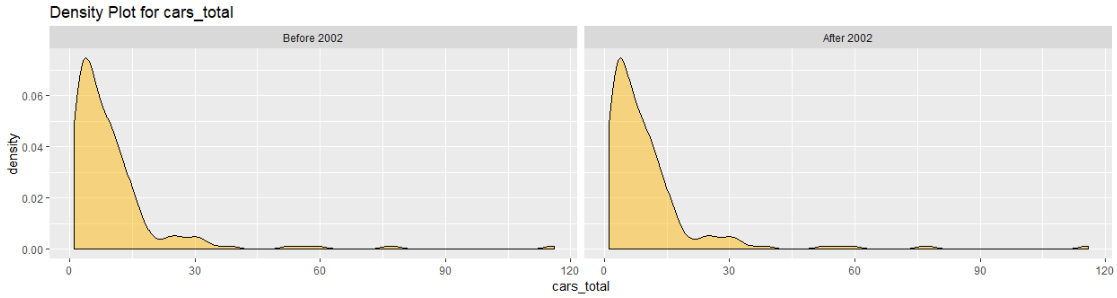
Total Cars

What we were able to observe:

1. There appears to be a positive relationship between number of total cars and the log transformation of violations. This makes intuitive sense.

```
plot_vars(df_clean, 'violations', 'cars_total', 'numeric', 'prepost')

## Warning: Removed 20 rows containing non-finite values (stat_smooth).

## Warning: Removed 20 rows containing missing values (geom_point).

## Warning: Removed 20 rows containing non-finite values (stat_density).
```





Analysis of Secondary Effects

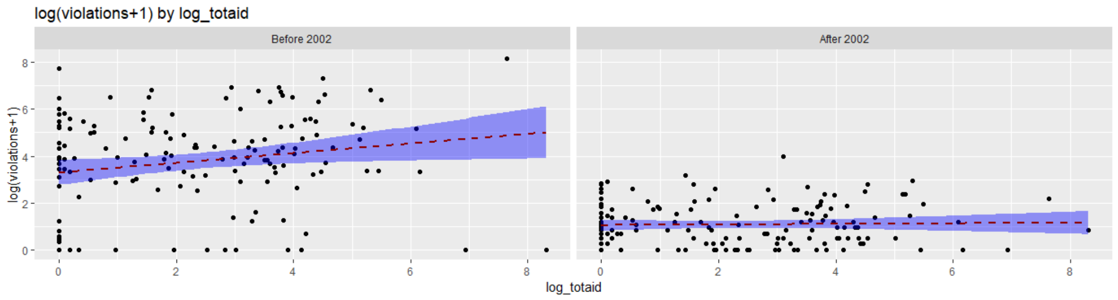
Total Aid

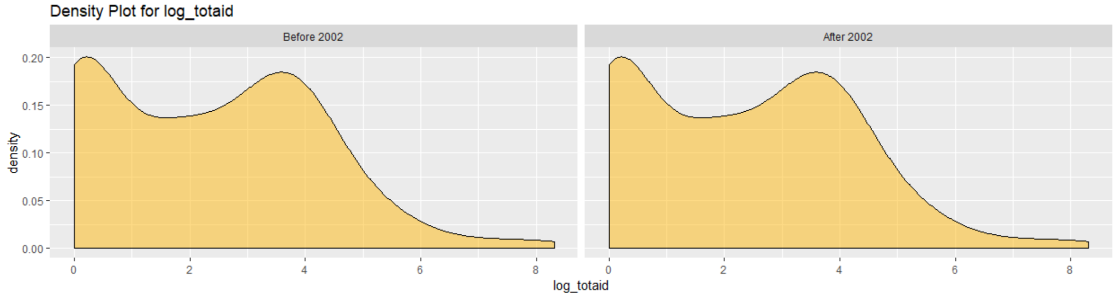
Among the remaining variables provided in the dataset, one of the most fascinating was *totaid*, which was slightly correlated with the number of violations that a given country received before immunity was granted, but became largely neutral after the change in policy.

As we see below, the stark difference in these visualizations (and resultant correlations) demonstrates the strong relationship between the amount of aid that a country received and the number of parking violations it received before the violation policy was altered. This is interesting, as we saw an unclear relationship between the log transformation of violations and that of trade, but we see one related to the amount of aid a country receives.

```r
# plot(log(df_clean[df_clean$prepost == "pre",]$violations + 1), log(df_clean[df_clean$prepost == "pre",]$totaid + 1), xlab = "Log(violations + 1)", ylab = "Log(total aid + 1)", main = "Comparison of total aid vs violations before policy change")
#
# plot(log(df_clean[df_clean$prepost == "pos",]$violations + 1), log(df_clean[df_clean$prepost == "pos",]$totaid + 1), xlab = "Log(violations + 1)", ylab = "Log(total aid + 1)", main = "Comparison of total aid vs violations after policy change")
#

df_clean$log_totaid <- log(df_clean$totaid+1)
plot_vars(df_clean, 'violations', 'log_totaid', 'numeric', 'prepost')
```

```
## Warning: Removed 4 rows containing non-finite values (stat_smooth).

## Warning: Removed 4 rows containing missing values (geom_point).

## Warning: Removed 4 rows containing non-finite values (stat_density).
```





Narratively, this could, perhaps, be an illustrative example of the impact of reliance on the United States with the desire to comply with their instituted laws: prior to a more strictly instituted policy, countries that were dependent on foreign aid were highly likely to violate parking regulations; however, after the United States began enforcing their policy, other countries became much more likely to comply.

That said, this could be a false narrative: there is a slightly negative correlation between “totaid” and country GDP, meaning that these violators could decide to better comply with local jurisdiction because they have less tolerance for paying significant fines.

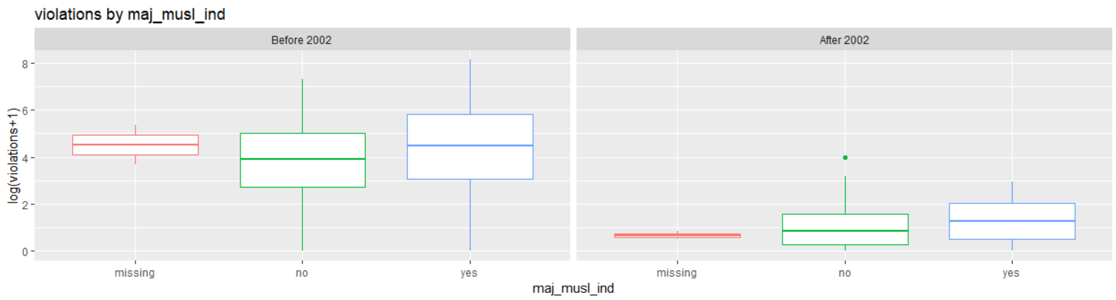
Majority Muslim country indicator

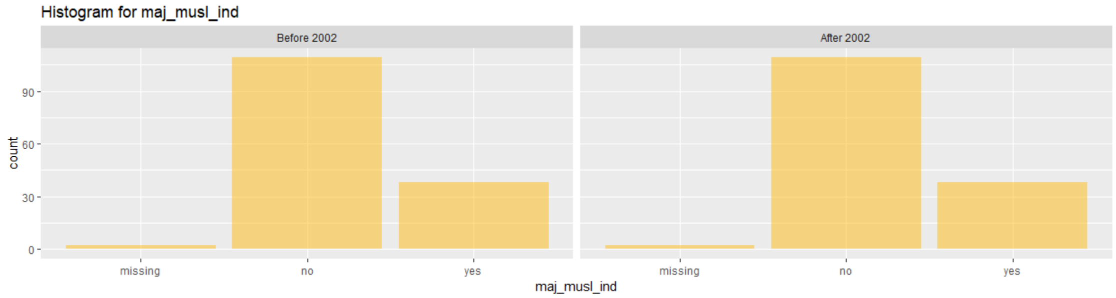
We are provided in the data set variables for **pctmuslim** and **majoritymuslim**, which are respectively percent of population that is muslim and a flag that captures whether or not a country is majority muslim. Unfortunately, the levels for the **majoritymuslim** are not intuitive from the information we were provided, so we created our own flag using the **pctmuslim** variable.

What we were able to observe:

1. There are many countries in these UN events that were not majority muslim than those that are.
2. We are unable to learn much information about this variable. Majority muslim countries appear to have higher log violations, but that difference is so small that it is likely due to noise. Majority muslim countries also represent less points in our data, which makes any aggregate information even more susceptible to noise.

```
df_clean$maj_musl_ind <- ifelse(is.na(df_clean$pctmuslim), 'missing',
ifelse(df_clean$pctmuslim>.5,'yes','no'))
plot_vars(df_clean, 'violations', 'maj_musl_ind', 'cat', 'prepost')
```

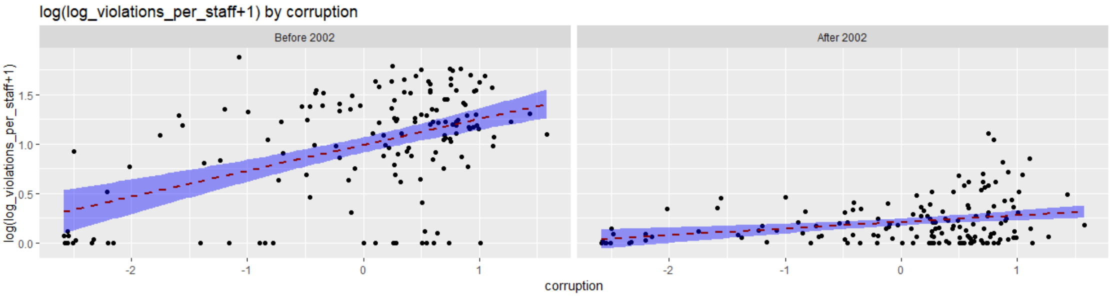


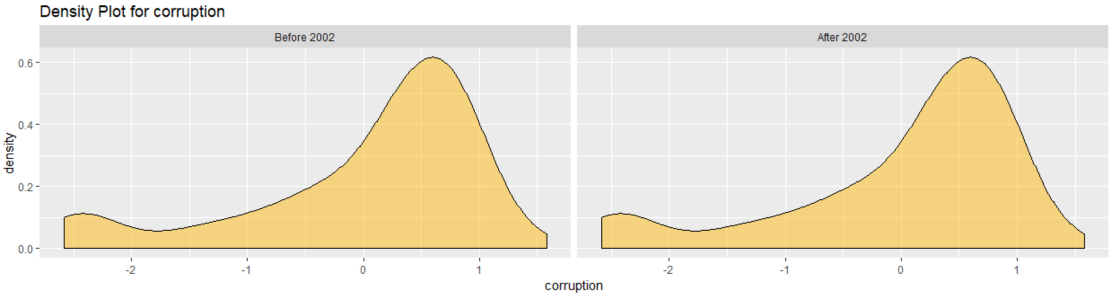


Violations per Staff Member

An additional way of framing the question of the number of violations by the diplomats of a given country, relative to that country’s level of corruption, is by looking at the number of violations per staff member. Logically, a country with more individuals at risk of violating parking policy will be more likely to more likely to, in fact, violate parking policy, but we see a surprisingly strong trend when viewing the log transformation of this variable relative to corruption.

```
df_clean$log_violations_per_staff <- log((df_clean$violations / df_clean$staff) + 1)
plot_vars(df_clean, 'log_violations_per_staff', 'corruption', 'numeric', 'prepost')
```





We can plainly see here that the log transformation of violations per staff member is correlated with corruption both before and after the policy change, but we see that the correlation is quite strong prior to the change.

Conclusion

After performing our exploratory data analysis on the given dataset on diplomatic parking violations in the United States before and after 2002, we are left to form conclusions to our initial question: is there any relationship between the corruption level of a country and the number of parking violations it received?

We learned that we do have robust data detailing the violation history of 149 unique countries before and after the policy change, and that the most common number of violations for a given country was zero. That said, we also saw a large variance in the number of violations that a country may receive, even when looking at countries within the same region. For example, Europe and the Middle East had the largest interquartile ranges of the log transformation of their violation count before 2002, yet showed a dramatic lessening of that range after the policy change.

But what does this mean for our target variables: corruption and violations? We discovered that there is a positive relationship between the number of violations and a country's corruption both before and after 2002. But we also see that the correlation is stronger after 2002. Although countries across the world saw themselves committing fewer parking violations, country's that were more corrupt tended to decrease their violations by less. This supports the hypothesis that there is a positive relationship between the level of corruption of a country and the willingness of its diplomats to violate parking laws, though it does not quantify that relationship.

Other questions that arose, to which we have incomplete answers, include:

1) Is there a relationship between trade with the United States and violations? Not particularly.
2) Do country diplomats bringing higher total cars to the UN tend to have more parking violations? Typically, yes.
3) Are countries who receive more aid more sensitive to changes in fine enforcement policies? Dramatically so.
4) Was there a language barrier between non-English speaking countries and the law enforcement of the United States that may have caused more confusion around diplomatic parking policy?

One final point that we would like to state that, at times, the variables within the provided dataset were presented with limited documentation, which inhibited our ability to perform as robust of an analysis as hoped.