Tutorial 1: Multiple Linear Regression

July 3, 2024

In this tutorial, we will study the dataset we will be studying is part of the **World Economics and Politics Dataverse** (WEP Dataverse).

The WEP Dataverse is a collaboration between of the Security and Political Economy (SPEC) Lab at the University of Southern California and the Niehaus Center for Globalization and Governance at Princeton University to create a queryable data resource for comparative and international political economy.

The WEP Dataverse contains more than 800 variables from around 100 country-year and data-year datasets widely used for research. You can visit https://ncgg.princeton.edu/wep/dataverse.html for more information, including the **codebook**.

We will study the statistical association between corruption and economic development, one of the most classical questions in the political economy of development. Another way to rephrase the question is: **Does corruption hinder economic development?**

William Easterly, a Professor of Economics at New York University, has a brief discussion on this topic in his 2001 book ***The Elusive Quest for Growth*** published by the MIT Press (see Chapter 12). We will revisit this question again later this term when we cover observational or non-experimental causal inference.

Before you start, please download the dataset from GitHub and import it into `RStudio`.

1 Variables

The WEP Dataverse retrieved variables containing `_VDEM` in their names from from the **Varieties of Democracy** (V-Dem) Dataset, the largest cross-national datasets of democratic politics and institutions built by the University of Gothenburg in Sweden.[1] Another well-known dataset from the same university is the **Quality of Government** (QoG) project.[2]

We will need the following variables.

- `country` (categorical)
- `year` (continuous/categorical)
- `v2x_corr_VDEM` – the index of political corruption, ranging between 0 and 10 (continuous)
- `v2x_polyarchy_VDEM` – the index of electoral democracy, ranging between 0 and 10 (continuous)
- `democracy_DD` – the indicator of democracy (binary/dummy)
- `gdppc_WDI_PW` – gross domestic product, per capita (continuous)
- `lngdppc_WDI_PW` – logged gross domestic product, per capita (continuous)

The binary indicator of democracy (i.e., it takes the value of 1 if the country is a democracy) is originally constructed by José Antonio Cheibub and his coauthors.[3]

[1] See https://v-dem.net/.
[2] See https://www.gu.se/en/quality-government.
[3] See Cheibub, José Antonio, Jennifer Gandhi, and James Raymond Vreeland. 2010. "Democracy and Dictatorship Revisited." *Public Choice 143*(1): 67-101.

2 Preparation

Let us load the following packages first.

```
library(car)
library(tidyverse)
library(stargazer)
library(modelsummary)
```

Now make sure you have imported the dataset into `RStudio`. Since the dataset is a `.csv` file, we can use `read_csv()` from the `readr` package – this package has been included when you load `tidyverse`.

```
dta <- read_csv("05_01_23_0912pm_wep.csv")
```

We can use the `tidyverse` approach to process the original dataset – namely, to subset the dataset and to select/recode the variables for our purpose.

```
dta_sel <- dta |>
  dplyr::select(country, year,
                gdppc_WDI_PW, lngdppc_WDI_PW,
                democracy_DD,
                v2x_polyarchy_VDEM, v2x_corr_VDEM) |>
  mutate(v2x_polyarchy_VDEM_10 = v2x_polyarchy_VDEM*10,
         v2x_corr_VDEM_10 = v2x_corr_VDEM*10) |>
  filter(year == 2000) |>
  unique() |>
  drop_na()
```

What does this chunk of code do? Can you unpack it line by line?

We can use `head()` to take a quick a look at our dataset.

```
head(dta_sel)
```

Whenever you run into any new functions, it is always a good idea to use the question `?` to see what the function does. You can also consult the package manual, if necessary.

```
?head
```

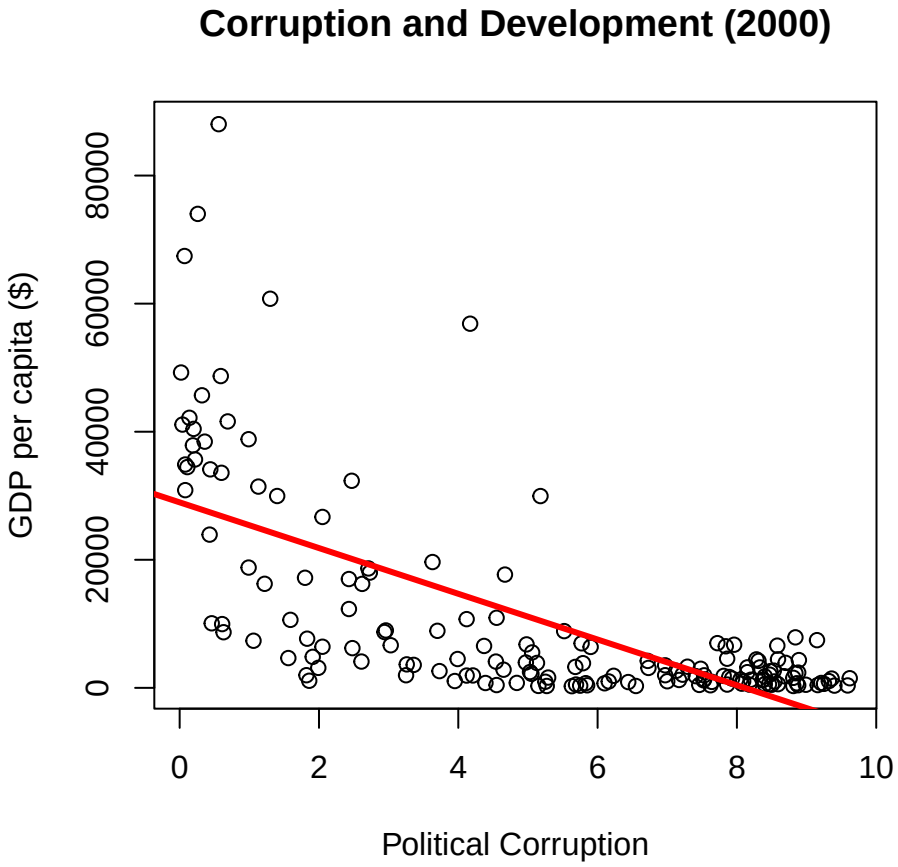
3 Visualize the Data

Before we conduct any regression analysis, we can create a quick scatter plot to "see" the data. Since we are interested in the relationship between corruption and development, the variables of our particular interest now are `v2x_corr_VDEM_10` and `gdppc_WDI_PW`.

```
plot(gdppc_WDI_PW ~ v2x_corr_VDEM_10, data=dta_sel,
     xlab = "Political Corruption",
     ylab = "GDP per capita ($)",
     main = "Corruption and Development (2000)")
```

Let's draw the regression line, using `abline`. The `abline` function has two arguments `col` and `lwd` we can use to choose the color and width of the line.

```
plot(gdppc_WDI_PW ~ v2x_corr_VDEM_10, data=dta_sel,
     xlab = "Political Corruption",
     ylab = "GDP per capita ($)",
     main = "Corruption and Development (2000)")
abline(lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10, data=dta_sel),
       col = "red",
       lwd = 3)
```



What can we learn from this (two-dimensional) scatter plot? Anything we can do to make it (look) better?

4 Correlation Without Regression

We can also check the pairwise correlation between `v2x_corr_VDEM_10` and `gdppc_WDI_PW`. One of the most common measures of pairwise correlation is the **Pearson correlation coefficient**. You can consult any statistical introductory textbook to see the formal definition or the formula.

To get the Pearson correlation coefficient between two variables, we can use `cor()`.

```
cor(dta_sel$v2x_corr_VDEM_10, dta_sel$gdppc_WDI_PW)
```

```
[1] -0.6780111
```

In the line above, we use the dollar sign `$` to call out a variable in a dataset.

> **What if our data contains missing values?**

5 Bivariate Linear Regression

5.1 Model Specification

Now let us start the real business – linear regression. Again, we are interested in the correlation between political corruption and economic development. To put it more specifically, here political corruption is the **explanatory** variable with economic development as the **outcome** variable.

It is always a good idea to **specify the model** before we start any analysis.

$$Y = \alpha + \beta X + \epsilon,$$

where

- Y is GDP per capita (`gdppc_WDI_PW`)
- X is political corruption (`v2x_corr_VDEM_10`)

Our goal is to use linear regression to estimate α and β, the intercept and slope respectively. In other words, linear regression should generate the following fitted line:

$$Y = \hat{\alpha} + \hat{\beta} X.$$

Note the fitted line should not include the error term.

5.2 The Analysis

The most canonical function to carry out linear regression in `R` is `lm()`. Below we will carry out the analysis and use `summary` to see the results.

```
b_corruption <- lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10, data=dta_sel)
summary(b_corruption)
```

```
Call:
lm(formula = gdppc_WDI_PW ~ v2x_corr_VDEM_10, data = dta_sel)

Residuals:
   Min     1Q Median     3Q    Max
-21218  -7758   -397   4963  61059
```

```
Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)         28966       1828   15.85   <2e-16 ***
v2x_corr_VDEM_10    -3570        304  -11.74   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11850 on 162 degrees of freedom
Multiple R-squared:  0.4597,    Adjusted R-squared:  0.4564
F-statistic: 137.8 on 1 and 162 DF,  p-value: < 2.2e-16
```

`R` also provides several useful tools for us to look into some more specific information of any linear regression analysis.

Function	Purpose
`coef()`	View the coefficients
`confint()`	View the confidence intervals
`resid()`	View the residuals
`fitted()`	View the predicted outcome

That being said, we can also use `ls()` to open (or "unearth") the output generated by the `lm()` function to obtain all the information above. We can similar use the dollar sign `$` to call them out.

```
ls(b_corruption)
```

```
 [1] "assign"        "call"          "coefficients"  "df.residual"
 [5] "effects"       "fitted.values" "model"         "qr"
 [9] "rank"          "residuals"     "terms"         "xlevels"
```

5.3 Statistical Inference

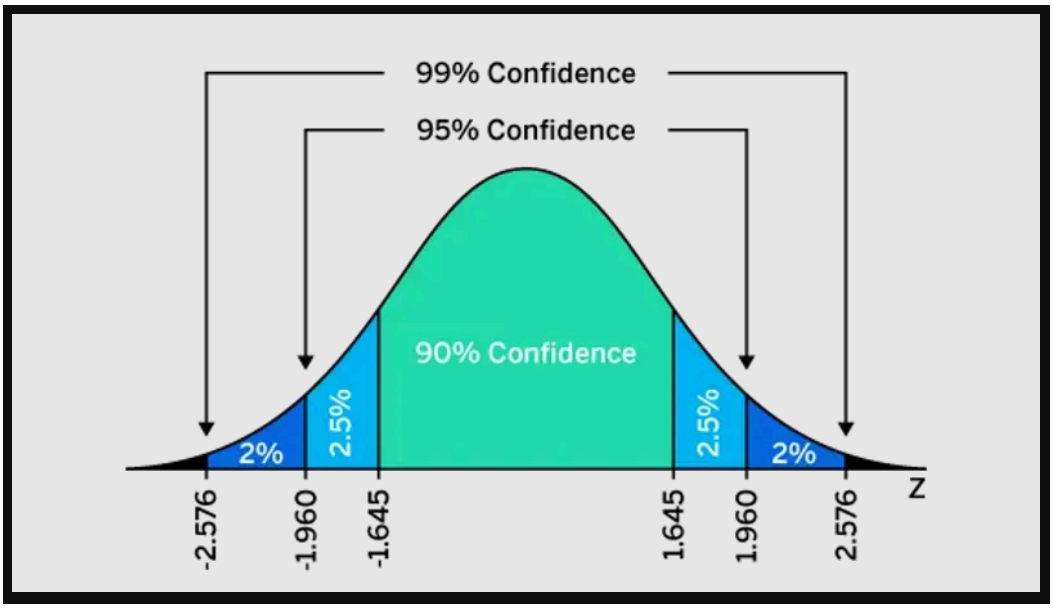
5.3.1 Statistical and Substantive Significance

The first order of business: **How should we interpret the estimated intercept and slope?** Are they statistically significant? Even if they are, does that mean corruption plays an important role in economic development?

```
confint(b_corruption)
```

```
                    2.5 %    97.5 %
(Intercept)      25357.313 32574.741
v2x_corr_VDEM_10 -4169.988 -2969.175
```

What do confidence intervals tell us?[4]



There are several ways to characterize the substantive significance of the estimated slope.

For instance, you can try plugging the standard deviation/error of the predictor to see the corresponding change in the outcome variable when we move the predictor by one standard deviation/error.

```
coef(b_corruption)[2]*sd(dta_sel$v2x_corr_VDEM_10, na.rm=T)
```

```
v2x_corr_VDEM_10
       -10898.71
```

We can also check the median of the outcome variable compare it with the estimated slope.

```
median(dta_sel$gdppc_WDI_PW, na.rm=T)
```

```
[1] 3287.696
```

[4]Source: Qualtrics.

```r
coef(b_corruption)[2]
```

```
v2x_corr_VDEM_10
       -3569.582
```

There is no a golden rule for this and you should read more quant research papers to see what researchers do.

5.3.2 Model Fit

Next, how about the **goodness-of-fit**? Let us take a look at the reported results again.

```r
b_corruption <- lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10, data=dta_sel)
summary(b_corruption)
```

```
Call:
lm(formula = gdppc_WDI_PW ~ v2x_corr_VDEM_10, data = dta_sel)

Residuals:
   Min     1Q Median     3Q    Max
-21218  -7758   -397   4963  61059

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)         28966       1828   15.85   <2e-16 ***
v2x_corr_VDEM_10    -3570        304  -11.74   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11850 on 162 degrees of freedom
Multiple R-squared:  0.4597,    Adjusted R-squared:  0.4564
F-statistic: 137.8 on 1 and 162 DF,  p-value: < 2.2e-16
```

R^2 is 0.46 while the adjusted R^2 is about the same. Again, R^2 is the fraction of the variance of the actual outcome explained by the variance of the predicted outcome. So we can verify the R^2 as follows.

```r
var_y_hat <- var(fitted(b_corruption))
var_y <- var(dta_sel$gdppc_WDI_PW)
var_y_hat/var_y
```

```
[1] 0.459699
```

The adjusted R^2 is a more advanced version to penalize the inclusion to too many variables (or the so-called "overfitting"). In other words, the formula of the adjusted R^2 will take the number of explanatory variables into account and the inclusion of more explanatory variables will suppress R^2 (i.e., adjust the original R^2 downwards).[5]

We can also try to calculate the sum of squared residuals as follows.

```r
sum(resid(b_corruption)^2)
```

```
[1] 22756218447
```

5.3.3 Diagnostics

Finally, you can try to run some **diagnostic tests to see how our model performs against the assumptions**.[6]

[5] See Section 4.2.6 in Imai (2018) for a more detailed discussion.
[6] See Section 1.6.1 in Roback and Leglar (2020) for a more detailed discussion.

6 Multiple Linear Regression

Now, let us consider another explanatory variable – democracy (`v2x_polyarchy_VDEM_10`). Political economists have also paid close attention to the relationship between democracy and development, with some arguing that democracy is more likely to induce long-term economic growth/development. For more discussion, check this article: **Bueno de Mesquita, Bruce, and George W. Downs. "Development and Democracy."** ***Foreign Affairs 84*****(5): 77-86.**

6.1 Model Specification

And with one more predictor our model now becomes

$$\text{Development} = \alpha + \beta_1(\text{Corruption}) + \beta_2(\text{Democracy}) + \epsilon.$$

Instead of using X and Y, you can also use the abbreviation of the key variable names and explain the operationalization of each variable more carefully in the text. We will see good examples of quantitative research articles/notes in Week 5.

It is fairly straightforward to include more explanatory variables using `lm()`. You can use the functions have introduced previously to obtain the predicted outcome, residuals, confidence intervals, and estimated coefficients.

```
m_mod <- lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10 + v2x_polyarchy_VDEM_10,
data=dta_sel)
summary(m_mod)
```

```
Call:
lm(formula = gdppc_WDI_PW ~ v2x_corr_VDEM_10 + v2x_polyarchy_VDEM_10,
    data = dta_sel)

Residuals:
   Min     1Q Median     3Q    Max
-22085  -7706   -266   4829  61175

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)           30112.6     4204.1   7.163 2.68e-11 ***
v2x_corr_VDEM_10      -3652.8      410.4  -8.900 1.11e-15 ***
```

```
v2x_polyarchy_VDEM_10   -139.0      458.8  -0.303    0.762
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11890 on 161 degrees of freedom
Multiple R-squared:   0.46, Adjusted R-squared:  0.4533
F-statistic: 68.58 on 2 and 161 DF,  p-value: < 2.2e-16
```

> **How else can you specify the data generation process of the dependent variable?**

6.2 Model Comparison

We can also use `anova()` to compare the goodness-of-fit across different models.

```
anova(b_corruption, m_mod)
```

```
Analysis of Variance Table

Model 1: gdppc_WDI_PW ~ v2x_corr_VDEM_10
Model 2: gdppc_WDI_PW ~ v2x_corr_VDEM_10 + v2x_polyarchy_VDEM_10
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    162 2.2756e+10
2    161 2.2743e+10  1  12972292 0.0918 0.7623
```

Numbers containing the small `e` mean they are tiny or huge (or: containing too many digits), and you can use the argument `scipen` in the `options()` function to turn it off.

```
options(scipen=9999)
anova(b_corruption, m_mod)
```

```
Analysis of Variance Table

Model 1: gdppc_WDI_PW ~ v2x_corr_VDEM_10
Model 2: gdppc_WDI_PW ~ v2x_corr_VDEM_10 + v2x_polyarchy_VDEM_10
  Res.Df         RSS Df Sum of Sq      F Pr(>F)
1    162 22756218447
2    161 22743246155  1  12972292 0.0918 0.7623
```

6.3 Regression Tables

We can also use the `stargazer()` function to present several regression models. Here is a quick example.

```
stargazer(list(b_corruption, m_mod),
          omit.stat = c("f", "rsq", "ser"),
          dep.var.caption = "GDP per capita",
          column.labels = c("Model 1", "Model 2"),
          covariate.labels = c("Corruption", "Democracy"),
          type = "text",
          digits = 3,
          no.space = T,
          intercept.bottom = TRUE,
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
==========================================
                    GDP per capita
              ----------------------------
                       gdppc_WDI_PW
                  Model 1        Model 2
                    (1)            (2)
------------------------------------------
Corruption     -3,569.582***  -3,652.836***
                 (304.047)      (410.420)
Democracy                       -139.042
                               (458.829)
Constant       28,966.030***  30,112.600***
                (1,827.460)    (4,204.064)
------------------------------------------
Observations       164            164
Adjusted R2       0.456          0.453
==========================================
Note:          *p<0.05; **p<0.01; ***p<0.001
```

> **Jack Russ has one of the most comprehensive guide on this function: https://www.jakeruss.com/cheatsheets/stargazer/. Check it out to the arguments you can add to adjust the table.**

	(1)	(2)
(Intercept)	28 966.027***	30 112.600***
	(1827.460)	(4204.064)
v2x_corr_VDEM_10	−3569.582***	−3652.836***
	(304.047)	(410.420)
v2x_polyarchy_VDEM_10		−139.042
		(458.829)
Num.Obs.	164	164
Log.Lik.	−1770.061	−1770.015
RMSE	11 779.53	11 776.18

* p < 0.1, ** p < 0.05, *** p < 0.01

A more recent alternative to `stargazer` is `modelsummary`:

```
modelsummary(list(b_corruption, m_mod),
             stars = c('*'=.1,'**'=.05,'***'=.01),
             gof_omit = "DF|Deviance|R2|AIC|BIC|F")
```

Visit https://modelsummary.com/vignettes/modelsummary.html to see the arguments you can add to adjust the table.

Problem Set

Question 1

Visit V-Dem's website, https://v-dem.net/data/the-v-dem-dataset/, and read through the definitions of political corruption (`v2x_corr`) and electoral democracy (`v2x_polyarchy`) in their codebook. Do you agree with their definitions? How else can you measure the same concepts? Discuss.

Question 2

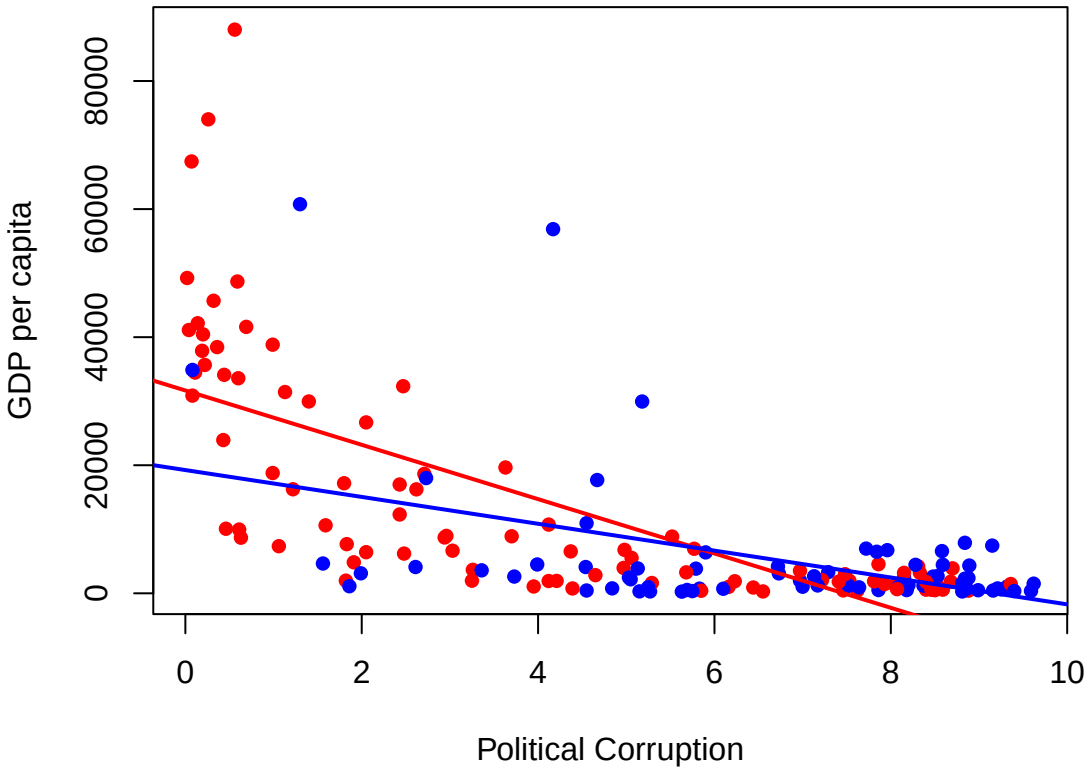
Use `v2x_polyarchy_VDEM_10` (the electoral democracy index) as the only independent variable to predict or explain a country's GDP per capita (`gdppc_WDI_PW`).

- Specify the model to be estimated (i.e., model specification with the error term)
- Show the scatter plot with the regression line
- Run the model (name the model as `b_democracy` in the script); interpret the estimated intercept and slope
- Calculate the 95% confidence interval of the estimated slope ($\widehat{\beta}$) and explain whether $\widehat{\beta}$ is statistically significant; if yes, discuss its substantive significance
- Use `stargazer` or `modelsummary` to show `b_democracy` and `m_mod` together with `b_corruption`; look across all model specifications and discuss whether democracy has an impact on development when we account for political corruption
- Use `anova()` to compare across different models to see which model performs the best

7 Extra: Interaction (Using the Binary Predictor of Democracy)

7.1 Scatterplot with Regression Line by Group

```
plot(gdppc_WDI_PW ~ v2x_corr_VDEM_10, data=dta_sel,
     col = ifelse(dta_sel$democracy_DD == 1, "red", "blue"),
     pch = 16,
     xlab = "Political Corruption",
     ylab = "GDP per capita")
abline(lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10,
       data=dta_sel[dta_sel$democracy_DD == 1,]),
       col="red", lwd=2)
abline(lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10,
       data=dta_sel[dta_sel$democracy_DD == 0,]),
       col="blue", lwd=2)
```



7.2 Multiple Regression Analysis with Interaction Term

```
m_int_1 <- lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10 + democracy_DD,
              data=dta_sel)
m_int_2 <- lm(gdppc_WDI_PW ~ v2x_corr_VDEM_10 + democracy_DD +
              v2x_corr_VDEM_10:democracy_DD, data=dta_sel)

stargazer(list(m_int_1, m_int_2),
          omit.stat = c("f", "rsq", "ser"),
          dep.var.caption = "GDP per capita",
          column.labels = c("Model 1", "Model 2"),
          covariate.labels = c("Corruption", "Democracy (=1)",
          "Corruption x Democracy (=1)"),
          type = "text",
          digits = 3,
          no.space = T,
          intercept.bottom = TRUE,
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
=============================================================
                                  GDP per capita
                              ---------------------------
                                     gdppc_WDI_PW
                                 Model 1        Model 2
                                   (1)            (2)
-------------------------------------------------------------
Corruption                    -3,573.812***  -2,097.128***
                                (335.680)      (586.749)
Democracy (=1)                   -62.117      12,423.810**
                               (2,058.916)    (4,580.555)
Corruption x Democracy (=1)                   -2,144.598**
                                               (707.101)
Constant                      29,022.800***  19,247.150***
                               (2,627.025)    (4,117.713)
-------------------------------------------------------------
Observations                       164            164
Adjusted R2                       0.453          0.479
=============================================================
Note:                         *p<0.05; **p<0.01; ***p<0.001
```

8 Extra: Log Transformation

```
m_log1 <- lm(lngdppc_WDI_PW ~ v2x_corr_VDEM_10, data=dta_sel)
m_log2 <- lm(lngdppc_WDI_PW ~ v2x_polyarchy_VDEM_10, data=dta_sel)
m_log3 <- lm(lngdppc_WDI_PW ~ v2x_corr_VDEM_10 + v2x_polyarchy_VDEM_10, data=dta_sel)

stargazer(list(m_log1, m_log2, m_log3),
          omit.stat = c("f", "rsq", "ser"),
          dep.var.caption = "GDP per capita (log)",
          column.labels = c("Model 1", "Model 2", "Model 3"),
          covariate.labels = c("Corruption", "Democracy (=1)"),
          type = "text",
          digits = 3,
          no.space = T,
          intercept.bottom = TRUE,
          star.cutoffs = c(0.05, 0.01, 0.001))
```

9 Extra: Panel Analysis (Use Country-Year as the Unit of Observation)

```
dta_sel_2 <- dta |>
  dplyr::select(country, year,
                gdppc_WDI_PW, lngdppc_WDI_PW, democracy_DD,
                v2x_polyarchy_VDEM, v2x_corr_VDEM) |>
  mutate(v2x_polyarchy_VDEM_10 = v2x_polyarchy_VDEM*10,
         v2x_corr_VDEM_10 = v2x_corr_VDEM*10) |>
  filter(year >= 2000 & year <= 2005) |>
  unique() |>
  drop_na()

m_long_1 <- lm(gdppc_WDI_PW ~ v2x_corr_VDEM + v2x_polyarchy_VDEM +
                as.factor(country) + as.factor(year),
                data=dta_sel_2)
m_long_2 <- lm(lngdppc_WDI_PW ~ v2x_corr_VDEM + v2x_polyarchy_VDEM +
                as.factor(country) + as.factor(year),
                data=dta_sel_2)

stargazer(list(m_long_1, m_long_2),
```

```
omit = c("as.factor"),
omit.stat = c("f", "rsq", "ser"),
dep.var.caption = "GDP per capita",
column.labels = c("Model 1", "Model 2"),
covariate.labels = c("Corruption", "Democracy"),
type = "text",
digits = 3,
no.space = T,
intercept.bottom = TRUE,
star.cutoffs = c(0.05, 0.01, 0.001))
```