Assignment 1

Bilal Shakir

2018-01-12

This assignment is based on the paper: Murali, Kanta. (2011) "Economic Liberalization, Electoral Coalitions and Private Investment in India." Mimeo. It focuses on the model building approach for nested analysis proposed by Lieberman (2005). In the following analysis, I set out to fulfill the tasks that were assigned under Assignment 1 by Prof. Leo Baccini.

Task 1.1

Focus on the following set of variables:

a. Outcome variable: FDI inflows or FDI stock
b. Explanatory variables: market size, income, infrastructure, and human capital

Solution 1.1

I begin the assignment by importing the data, loading the required packages and transforming the variables. Transformation of both outcome (*fdi*, *fdi_stock*) and explanatory (*grppc*, *trade*, *grp*, *pop*) variables was necessary for better data visualization and description. This point will be elaborated further in the analysis undermentioned.

To emphasize the undermentioned are our variables of interest:

1) Outcome variable: fdi and fdi_stock have already been specified and are available in the dataset. We transform both these variables.

2) Explanatory variables: On the other hand the variables of market size, income, infrastructure and human capital are not available within the dataset. As such we have to select the most appropriate proxies for these variables. Broadly, I opt to use a similar selection strategy as the one suggested by Prof. Baccini in the course materials.

For market size: I opt for population *pop* and gross regional product *grp*.

For Income: I opt for *trade* and GRP per capita *grppc* as proxies.

Infrastructure: *roaddensity* can be thought of as a good proxy. Another option was the number of airports, however, roaddensity has a higher degree of variation and in this case makes more theoretical sense to use as a measure of infrastructure. Especially given the likelihood that airports will often only be present in relatively prosperous or populous regions of Russia, there isnt sufficient variation. Crucialy, the historical nature of the data means that there is a chance that during earlier years there might not be a lot of .

Human capital: Some commonly used indicators for human capital include budgetary expenditure on social services (*bsocial*), healthcare (*bhealthcare*) and education (*beducation*). However, I only opt to include the number of people employed in higher education (*empl_high*) as a proxy for human capital. This was necessary due to several reasons. Crucially, existing literature, such as Baccini et al (2014), have already used *empl_high* as a proxy for human capital with great effect. On the other hand, there can be suspicion for the endogeneous generation of these variables which makes *empl_high* a better measure of human capital from a theoretical poin of view.

Reference

Baccini, Leonardo, Quan Li, and Irina Mirkina. "Corporate tax cuts and foreign direct investment." Journal of policy Analysis and Management 33, no. 4 (2014): 977-1006.

```
options(warn=-1)

## clearing out the global environment is a good idea before starting out

remove(list=ls())

## loading in the required datasets

pacman::p_load(tidyr, tidyverse,
               ggplot2, ggthemes,
               dplyr, RColorBrewer,
               gridExtra,
               texreg, readr,
               foreign, readstata13,
               interplot, stargazer,
               Zelig, broom,
               car, purrr,
               MASS, arm,
               pander, knitr,
               psych, cowplot,
               margins, scales,
               pastecs, plm, survival, ggplot2, ggrepel, pscl)

## renaming datasets for ease of use later

data_russia <- read.dta("DatasetRussia.dta")

# Importing data and transforming the variables

data_russia$ln_pop <- log(data_russia$pop)
data_russia$ln_grppc <- log(data_russia$grppc)
data_russia$ln_trade <- log(data_russia$trade)
data_russia$ln_grp <- log(data_russia$grp)
data_russia$sqrt_fdi <- sqrt(data_russia$fdi)
data_russia$ln_fdi <- log(data_russia$fdi+1)
data_russia$ln_fdi_stock <- log(data_russia$fdi_stock+1)
data_russia$ln_roaddensity <- log(data_russia$roaddensity)
```

Task 1.1

Produce informative descriptive statistics using the appropriate tables, graphs, plots. This descriptive analysis should help you (and the reader) to understand the data that you are dealing with

Solution 1.1

Justification for log transformation

First, in the following code chunk I showcase the rationale behind log-transforming the variables of interest in the earlier code. I only showcase this for the outcome variables. However, the rationale underpinng the transformation of both the outcome variable and the co-variates remains the same. Namely, log-transforming the data for the case of the outcome variables was necessary to ensure that OLS assumptions (recall that OLS is a BLUE estimator) are valid. Moreover, 1 had to be added in the case there were any observations with 0 as their values. This is necessary as the log of 0 is infinity.

```
figure_1 <- ggplot(data = data_russia, mapping = aes(x = fdi)) +
  geom_density(aes(x = fdi, fill = "FDI")) +
  geom_density(aes(x = fdi_stock, fill = "FDI Stock")) +
  theme_linedraw() +
  labs(title = "Figure 1: Density Plot for untransformed Outcome Variables",
       x = "FDI and FDI Stock",
       y = "Density") +
  scale_fill_discrete(name = "Outcome Variable"); figure_1
```

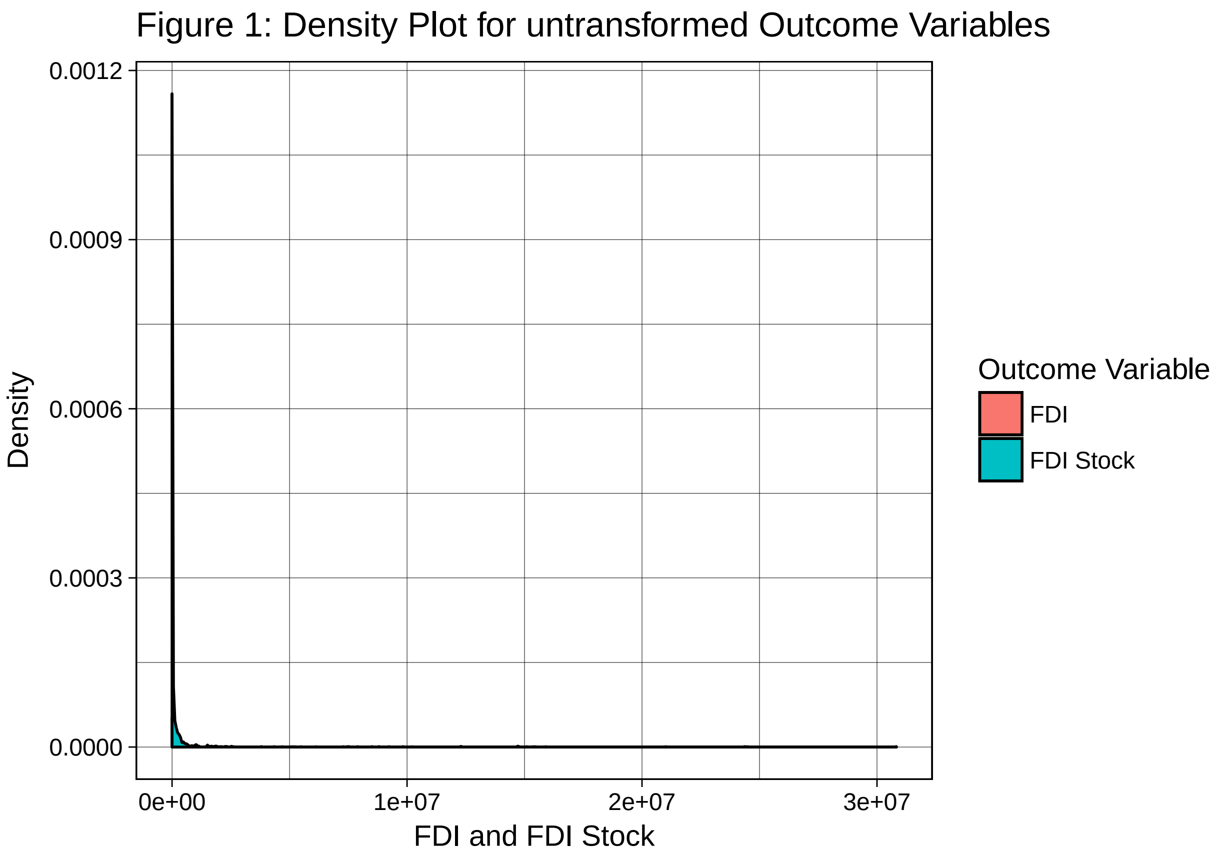


Figure 1 Notes: Figure 1 makes it clear that outcome variables need to be transformed. This attests that our previous strategy of log-transforming the outcome variables and several explanatory variables holds.

More focused descriptive stats

Next, in the following code chunk we see that the dataset has 68 variables. This makes it inefficient to produce descriptive statistics for the entire dataset. As such, we only focus on the variables of interest in our analysis that we have aleady identified earlier. I describe the data through graphs, and box-plots.

```
## Let's start off the assignment by looking at the broad structure of our dataset

glimpse(data_russia)
```

```
## Observations: 1,722
## Variables: 70
## $ id             <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
```

```
## $ region         <chr> "adygea_rep", "adygea_rep", "adygea_rep", "adyg...
## $ year           <dbl> 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997,...
## $ name           <chr> "Republic of Adygea", "Republic of Adygea", "Re...
## $ capital        <chr> "Maykop", "Maykop", "Maykop", "Maykop", "Maykop...
## $ feddistrict    <fctr> Southern, Southern, Southern, Southern, Southe...
## $ economzone     <fctr> North Caucasus, North Caucasus, North Caucasus...
## $ climatezone    <fctr> humid subtropical or semi-arid, humid subtropi...
## $ arctic         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ area           <int> 7600, 7600, 7600, 7600, 7600, 7600, 7600, 7600,...
## $ distance       <int> 1404, 1404, 1404, 1404, 1404, 1404, 1404, 1404,...
## $ airport        <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ intairport     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ republic       <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ border         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ sez            <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ oilprice       <dbl> 31.2, 25.3, 23.7, 20.3, 18.4, 19.2, 22.6, 20.5,...
## $ pop            <dbl> 432, 437, 441, 446, 448, 450, 450, 449, 450, 44...
## $ census         <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ ruspopul       <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ russhare       <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ city           <dbl> 52.1, 52.6, 52.7, 54.0, 53.8, 53.3, 53.8, 53.9,...
## $ mw             <int> 1178, 1170, 1166, 1167, 1159, 1153, 1152, 1152,...
## $ young          <dbl> 24.7, 24.6, 24.5, 24.3, 24.0, 23.6, 23.2, 22.7,...
## $ adult          <dbl> 53.9, 54.0, 54.0, 53.9, 54.1, 54.0, 54.3, 54.5,...
## $ old            <dbl> 21.4, 21.4, 21.5, 21.8, 21.9, 22.4, 22.5, 22.8,...
## $ birth          <dbl> 14.1, 13.4, 11.9, 10.7, 10.9, 10.7, 10.3, 9.8, ...
## $ mort           <dbl> 12.3, 13.4, 13.4, 14.9, 14.5, 14.4, 14.2, 14.0,...
## $ infmort        <dbl> 17.2, NA, NA, NA, NA, 18.7, NA, NA, 12.9, 13.3,...
## $ unemp          <dbl> NA, NA, NA, 8.0, 13.0, 11.3, 11.1, 12.3, 15.6, ...
## $ wagepc         <dbl> 516.581, 1046.706, 17.665, 46.467, 69.492, 73.3...
## $ gini           <dbl> NA, NA, NA, NA, NA, 0.290, 0.361, 0.416, 0.414,...
## $ subsidy        <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, 0.109, 0.17...
## $ brevenue       <dbl> NA, NA, 21.543, 47.993, 58.066, 85.833, 122.482...
## $ btransfers     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 24.430,...
## $ taxes          <dbl> NA, NA, NA, NA, NA, NA, NA, NA, 47.606, 32.687,...
## $ bsocial        <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 30.093,...
## $ beducation     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ bhealthcare    <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ bsecurity      <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ grp            <dbl> NA, NA, NA, NA, NA, 430.947, 508.501, 522.303, ...
## $ grppc          <dbl> NA, NA, NA, NA, NA, 956.724, 1129.535, 1160.714...
## $ realgrpgrowth  <dbl> NA, NA, NA, NA, NA, NA, NA, -9.1, -3.5, 5.4, 5....
## $ trade          <dbl> NA, NA, NA, NA, NA, NA, NA, 19.2, 14.5, 12.7, 1...
## $ fdi            <dbl> 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 52.7, 26.8, 684.5...
## $ roaddensity    <dbl> 184.2, 185.0, 186.0, 186.0, 187.0, 193.0, 195.0...
## $ crimeshare     <int> NA, 1059, 1461, 1392, 1367, 1344, 1510, 1375, 1...
## $ invrisk        <int> NA, NA, NA, NA, NA, NA, 45, 50, 74, 51, 52, 40,...
## $ demrating      <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 22,...
## $ spatial_fdi    <dbl> NA, NA, NA, NA, NA, 0.00, 9.93, 9.93, 13.00, 18...
## $ bcwin          <int> NA, NA, 0, NA, NA, NA, NA, 0, NA, NA, NA, NA, 1...
## $ party_strength <dbl> NA, NA, 0.3333333, NA, NA, NA, NA, 0.3846154, N...
## $ corruption     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 2.3...
## $ deficit        <dbl> NA, NA, NA, NA, NA, 0.00000, 0.00000, 0.00000, ...
## $ empl_high      <dbl> NA, NA, NA, NA, NA, 12.9, 13.3, 13.9, 19.4, 18....
```

```
## $ firms          <int> NA, 6, 1848, 4835, 6016, 6481, 7312, 7419, 7208...
## $ sme_n          <dbl> NA, NA, NA, NA, NA, 2.2, 2.1, 2.2, 2.2, 2.2, 2....
## $ priv_n         <int> NA, NA, NA, 129, 59, 5, NA, NA, NA, NA, NA, NA,...
## $ ffirms_n       <int> NA, NA, NA, NA, NA, NA, NA, NA, 15, 12, 18, 11,...
## $ fdi_stock      <dbl> 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0....
## $ prod_oil       <dbl> 18.0, 12.0, 6.0, 3.0, 7.0, 11.0, 10.0, 0.9, 0.7...
## $ prod_gas       <int> 693, 530, 510, 501, 489, 472, 464, 209, 198, 18...
## $ ln_pop         <dbl> 6.068426, 6.079933, 6.089045, 6.100319, 6.10479...
## $ ln_grppc       <dbl> NA, NA, NA, NA, NA, 6.863515, 7.029561, 7.05679...
## $ ln_trade       <dbl> NA, NA, NA, NA, NA, NA, NA, 2.954910, 2.674149,...
## $ ln_grp         <dbl> NA, NA, NA, NA, NA, 6.065985, 6.231467, 6.25824...
## $ sqrt_fdi       <dbl> 0.000000, 0.000000, 0.000000, 0.000000, 0.00000...
## $ ln_fdi         <dbl> 0.000000, 0.000000, 0.000000, 0.000000, 0.00000...
## $ ln_fdi_stock   <dbl> 0.000000, 0.000000, 0.000000, 0.000000, 0.00000...
## $ ln_roaddensity <dbl> 5.216022, 5.220356, 5.225747, 5.225747, 5.23110...
```

```r
# The glimpse command above shows that there are a total of 68 variables and 1722 observations in our d

## Descriptive statistics

## For whole dataset -  we can see that the dataset are too large

desc_data <- stat.desc(data_russia); glimpse(desc_data)
```

```
## Observations: 14
## Variables: 70
## $ id             <dbl> 1.722000e+03, 0.000000e+00, 0.000000e+00, 1.000...
## $ region         <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ year           <dbl> 1.722000e+03, 0.000000e+00, 0.000000e+00, 1.990...
## $ name           <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ capital        <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ feddistrict    <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ economzone     <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ climatezone    <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ arctic         <dbl> 1.722000e+03, 1.554000e+03, 0.000000e+00, 0.000...
## $ area           <dbl> 1.722000e+03, 0.000000e+00, 0.000000e+00, 1.100...
## $ distance       <dbl> 1.722000e+03, 4.200000e+01, 0.000000e+00, 0.000...
## $ airport        <dbl> 1.722000e+03, 1.470000e+02, 0.000000e+00, 0.000...
## $ intairport     <dbl> 1.722000e+03, 4.830000e+02, 0.000000e+00, 0.000...
## $ republic       <dbl> 1.722000e+03, 1.281000e+03, 0.000000e+00, 0.000...
## $ border         <dbl> 1.722000e+03, 7.980000e+02, 0.000000e+00, 0.000...
## $ sez            <dbl> 1.722000e+03, 1.630000e+03, 0.000000e+00, 0.000...
## $ oilprice       <dbl> 1.722000e+03, 0.000000e+00, 0.000000e+00, 1.350...
## $ pop            <dbl> 1.712000e+03, 0.000000e+00, 1.000000e+01, 4.900...
## $ census         <dbl> 2.370000e+02, 0.000000e+00, 1.485000e+03, 5.052...
## $ ruspopul       <dbl> 2.370000e+02, 0.000000e+00, 1.485000e+03, 3.215...
## $ russhare       <dbl> 2.370000e+02, 0.000000e+00, 1.485000e+03, 8.000...
## $ city           <dbl> 1.719000e+03, 0.000000e+00, 3.000000e+00, 2.390...
## $ mw             <dbl> 1.719000e+03, 0.000000e+00, 3.000000e+00, 8.340...
## $ young          <dbl> 1.636000e+03, 0.000000e+00, 8.600000e+01, 1.230...
## $ adult          <dbl> 1.636000e+03, 0.000000e+00, 8.600000e+01, 5.090...
## $ old            <dbl> 1.636000e+03, 0.000000e+00, 8.600000e+01, 2.400...
## $ birth          <dbl> 1.707000e+03, 0.000000e+00, 1.500000e+01, 6.200...
## $ mort           <dbl> 1.707000e+03, 0.000000e+00, 1.500000e+01, 3.100...
## $ infmort        <dbl> 1.223000e+03, 0.000000e+00, 4.990000e+02, 4.000...
```

```
## $ unemp          <dbl> 1.449000e+03, 0.000000e+00, 2.730000e+02, 8.000...
## $ wagepc         <dbl> 1.710000e+03, 0.000000e+00, 1.200000e+01, 1.120...
## $ gini           <dbl> 1.276000e+03, 0.000000e+00, 4.460000e+02, 2.310...
## $ subsidy        <dbl> 973.000000, 1.000000, 749.000000, 0.000000, 185...
## $ brevenue       <dbl> 1.555000e+03, 0.000000e+00, 1.670000e+02, 1.960...
## $ btransfers     <dbl> 901.000000, 0.000000, 821.000000, -6562.070801,...
## $ taxes          <dbl> 9.820000e+02, 0.000000e+00, 7.400000e+02, 4.085...
## $ bsocial        <dbl> 9.020000e+02, 0.000000e+00, 8.200000e+02, 1.800...
## $ beducation     <dbl> 5.740000e+02, 0.000000e+00, 1.148000e+03, 1.861...
## $ bhealthcare    <dbl> 5.740000e+02, 0.000000e+00, 1.148000e+03, 1.230...
## $ bsecurity      <dbl> 410.000000, 0.000000, 1312.000000, 4.267000, 10...
## $ grp            <dbl> 1.292000e+03, 0.000000e+00, 4.300000e+02, 8.515...
## $ grppc          <dbl> 1.292000e+03, 0.000000e+00, 4.300000e+02, 2.113...
## $ realgrpgrowth  <dbl> 1131.0000000, 8.0000000, 591.0000000, -22.90000...
## $ trade          <dbl> 1.116000e+03, 0.000000e+00, 6.060000e+02, 1.000...
## $ fdi            <dbl> 1.722000e+03, 5.630000e+02, 0.000000e+00, 0.000...
## $ roaddensity    <dbl> 1.667000e+03, 0.000000e+00, 5.500000e+01, 1.000...
## $ crimeshare     <dbl> 1.694000e+03, 0.000000e+00, 2.800000e+01, 2.300...
## $ invrisk        <dbl> 1.230000e+03, 0.000000e+00, 4.920000e+02, 1.000...
## $ demrating      <dbl> 243.0000000, 0.0000000, 1479.0000000, 14.000000...
## $ spatial_fdi    <dbl> 1.312000e+03, 9.300000e+01, 4.100000e+02, 0.000...
## $ bcwin          <dbl> 2.300000e+02, 2.170000e+02, 1.492000e+03, 0.000...
## $ party_strength <dbl> 2.270000e+02, 1.500000e+01, 1.495000e+03, 0.000...
## $ corruption     <dbl> 4.050000e+02, 0.000000e+00, 1.317000e+03, 2.000...
## $ deficit        <dbl> 1.148000e+03, 4.110000e+02, 5.740000e+02, -2.19...
## $ empl_high      <dbl> 1.185000e+03, 0.000000e+00, 5.370000e+02, 7.300...
## $ firms          <dbl> 1.690000e+03, 0.000000e+00, 3.200000e+01, 2.000...
## $ sme_n          <dbl> 1.214000e+03, 0.000000e+00, 5.080000e+02, 1.000...
## $ priv_n         <dbl> 1164.000000, 0.000000, 558.000000, 1.000000, 17...
## $ ffirms_n       <dbl> 1.015000e+03, 0.000000e+00, 7.070000e+02, 1.000...
## $ fdi_stock      <dbl> 1.722000e+03, 1.275000e+03, 0.000000e+00, 0.000...
## $ prod_oil       <dbl> 5.590000e+02, 0.000000e+00, 1.163000e+03, 3.000...
## $ prod_gas       <dbl> 4.770000e+02, 0.000000e+00, 1.245000e+03, 1.000...
## $ ln_pop         <dbl> 1.712000e+03, 0.000000e+00, 1.000000e+01, 3.891...
## $ ln_grppc       <dbl> 1.292000e+03, 0.000000e+00, 4.300000e+02, 5.353...
## $ ln_trade       <dbl> 1116.00000000, 0.00000000, 606.00000000, -2.302...
## $ ln_grp         <dbl> 1.292000e+03, 0.000000e+00, 4.300000e+02, 4.444...
## $ sqrt_fdi       <dbl> 1.722000e+03, 5.630000e+02, 0.000000e+00, 0.000...
## $ ln_fdi         <dbl> 1.722000e+03, 5.630000e+02, 0.000000e+00, 0.000...
## $ ln_fdi_stock   <dbl> 1722.0000000, 1275.0000000, 0.0000000, 0.000000...
## $ ln_roaddensity <dbl> 1667.00000000, 0.00000000, 55.00000000, -2.3025...
```

```r
## For OUTCOME variables

# First FDI inflows or FDI

desc_ln_fdi <- stat.desc(data_russia$ln_fdi)

# Second is for FDI stock

desc_ln_fdi_stock <- stat.desc(data_russia$ln_fdi_stock)

## For explanatory variables
```

```r
## First market size variables: pop and grp

desc_ln_pop <- stat.desc(data_russia$ln_pop)

desc_ln_grp <- stat.desc(data_russia$ln_grp)

## Income variables: trade and gross regional product per capita

desc_ln_trade <- stat.desc(data_russia$ln_trade)

desc_ln_grppc <- stat.desc(data_russia$ln_grppc)

## Infrastructire variable: road density

desc_ln_roaddensity <- stat.desc(data_russia$ln_roaddensity)

## Human capital- number of people of employed workforce with higher education

desc_empl_high <- stat.desc(data_russia$empl_high)

## Let's show all these descriptive stats in one table

## First, we try to bind all of the variables of interest in our model together

descriptives_var <- cbind(data_russia$ln_fdi,
                          data_russia$ln_fdi_stock,
                          data_russia$ln_pop,
                          data_russia$ln_grp,
                          data_russia$ln_trade,
                          data_russia$ln_grppc,
                          data_russia$roaddensity,
                          data_russia$empl_high)

## Combine all the descriptives stats that we calculated into a single dataframe

descriptives_total <- data.frame(desc_ln_fdi, desc_ln_fdi_stock, desc_ln_pop, desc_ln_grp, desc_ln_trade

## Setnames
descriptives_names <- setNames(descriptives_total,
                             c("FDI", "FDI Stock", "Population", "Gross Regional Product", "Trade", "Gr
```

```
##                        FDI    FDI Stock   Population Gross Regional Product
## nbr.val      1.722000e+03 1722.0000000 1.712000e+03           1.292000e+03
## nbr.null     5.630000e+02 1275.0000000 0.000000e+00           0.000000e+00
## nbr.na       0.000000e+00    0.0000000 1.000000e+01           4.300000e+02
## min          0.000000e+00    0.0000000 3.891820e+00           4.444450e+00
## max          1.638064e+01   17.2437504 9.351319e+00           1.248834e+01
## range        1.638064e+01   17.2437504 5.459499e+00           8.043894e+00
## sum          1.063628e+04 5256.1995005 1.226251e+04           1.025612e+04
## median       7.776782e+00    0.0000000 7.184629e+00           7.934495e+00
## mean         6.176702e+00    3.0523807 7.162681e+00           7.938175e+00
## SE.mean      1.149027e-01    0.1274272 2.050275e-02           3.413431e-02
## CI.mean.0.95 2.253636e-01    0.2499285 4.021310e-02           6.696480e-02
## var          2.273493e+01   27.9613036 7.196611e-01           1.505375e+00
```

```
## std.dev      4.768116e+00    5.2878449 8.483284e-01           1.226937e+00
## coef.var     7.719517e-01    1.7323674 1.184373e-01           1.545617e-01
##                     Trade Gross regional product per capita  Road density
## nbr.val      1116.00000000                     1.292000e+03 1667.00000000
## nbr.null        0.00000000                     0.000000e+00    0.00000000
## nbr.na        606.00000000                     4.300000e+02   55.00000000
## min            -2.30258508                     5.353473e+00   -2.30258508
## max            12.36824929                     1.065774e+01    6.74299852
## range          14.67083437                     5.304269e+00    9.04558360
## sum          7233.31918428                     9.941159e+03 7042.22588983
## median          6.54893481                     7.643654e+00    4.72384170
## mean            6.48146880                     7.694395e+00    4.22449064
## SE.mean         0.05446722                     2.246752e-02    0.03371468
## CI.mean.0.95    0.10686980                     4.407685e-02    0.06612760
## var             3.31081299                     6.521879e-01    1.89484484
## std.dev         1.81956396                     8.075815e-01    1.37653363
## coef.var        0.28073327                     1.049571e-01    0.32584606
##              People Employed with higher education
## nbr.val                               1.185000e+03
## nbr.null                              0.000000e+00
## nbr.na                                5.370000e+02
## min                                   7.300000e+00
## max                                   5.120000e+01
## range                                 4.390000e+01
## sum                                   2.494030e+04
## median                                2.020000e+01
## mean                                  2.104667e+01
## SE.mean                               1.649467e-01
## CI.mean.0.95                          3.236203e-01
## var                                   3.224077e+01
## std.dev                               5.678095e+00
## coef.var                              2.697860e-01
```

```
## Here is the tabular representation of our variables of interest

#options(digits = 4)
#knitr::kable(descriptives_names, digits = 2, caption = "Descriptive Stats for variables of interest")

## Just the descriptives

descriptives_descriptives <- stat.desc(descriptives_names, basic = F); descriptives_descriptives
```

```
##                       FDI    FDI Stock   Population Gross Regional Product
## median       6.976742e+00 4.170113e+00 4.675659e+00           6.189472e+00
## mean         9.283294e+02 5.947213e+02 1.001379e+03           8.587114e+02
## SE.mean      7.571954e+02 3.870731e+02 8.747426e+02           7.289884e+02
## CI.mean.0.95 1.635821e+03 8.362205e+02 1.889767e+03           1.574884e+03
## var          8.026828e+06 2.097558e+06 1.071244e+07           7.439936e+06
## std.dev      2.833166e+03 1.448295e+03 3.272987e+03           2.727625e+03
## coef.var     3.051897e+00 2.435250e+00 3.268480e+00           3.176416e+00
##                     Trade Gross regional product per capita Road density
## median       4.896141e+00                      5.328871e+00 3.059668e+00
## mean         6.427613e+02                      8.358174e+02 6.278827e+02
## SE.mean      5.143548e+02                      7.067214e+02 5.073944e+02
## CI.mean.0.95 1.111196e+03                      1.526779e+03 1.096159e+03
```

```
## var          3.703852e+06                      6.992371e+06 3.604287e+06
## std.dev      1.924540e+03                      2.644309e+03 1.898496e+03
## coef.var     2.994174e+00                      3.163740e+00 3.023648e+00
##              People Employed with higher education
## median                              2.062333e+01
## mean                                1.917473e+03
## SE.mean                             1.773188e+03
## CI.mean.0.95                        3.830741e+03
## var                                 4.401876e+07
## std.dev                             6.634663e+03
## coef.var                            3.460108e+00
```

```
## Just the basic statistics

descriptives_basic <- stat.desc(descriptives_names,
                                desc = F); descriptives_basic
```

```
##                FDI FDI Stock Population Gross Regional Product       Trade
## nbr.val      14.00    14.000      14.00                  14.00   14.000000
## nbr.null      2.00     3.000       1.00                   1.00    1.000000
## nbr.na        0.00     0.000       0.00                   0.00    0.000000
## min           0.00     0.000       0.00                   0.00   -2.302585
## max       10636.28  5256.200   12262.51               10256.12 7233.319184
## range     10636.28  5256.200   12262.51               10256.12 7235.621769
## sum       12996.61  8326.098   14019.31               12021.96 8998.658534
##           Gross regional product per capita Road density
## nbr.val                              14.000    14.000000
## nbr.null                              1.000     1.000000
## nbr.na                                0.000     0.000000
## min                                   0.000    -2.302585
## max                                9941.159  7042.225890
## range                              9941.159  7044.528475
## sum                               11701.444  8790.357286
##           People Employed with higher education
## nbr.val                                   14.00
## nbr.null                                   1.00
## nbr.na                                     0.00
## min                                        0.00
## max                                    24940.30
## range                                  24940.30
## sum                                    26844.62
```

From the descriptive statistics above (under the output *descriptives_names*), we can observe several trends. First, regarding the outcome variables we observe that there are a very large number of zero observations (563 for FDI and 1275 for FDI Stock). This can be gauged by the very fact that the median for FDI stock is zero, meaning at least half the region-years have FDI stock equal to zero. Whereas the median for FDI is 7.76. The Maximum values of the outcome variables FDI and FDI stock are 16.38 and 17.24. The descriptive stats also show that there are no missing observations for the outcome variables however this is because we log transformed the outcome variables so that number is not substantively important.

Whereas for the case of the explanatory variables we immediately see that there are a lot of missing observations. For instance, 10 and 430 for population and gross regional product, 606 out of a total (out of a total of 1116) for trade, 430 for grppc and 55 for roaddensity and 537 for *empl_high*. This indicates that there are a lot of explanatory variables that are just missing from the dataset. Note that it is odd to have a minimum value of -2.3 for trade as it is not possible for trade to be in the negative. It is possible that log transforming the data lead to a negative value in this case which is inaccurate.

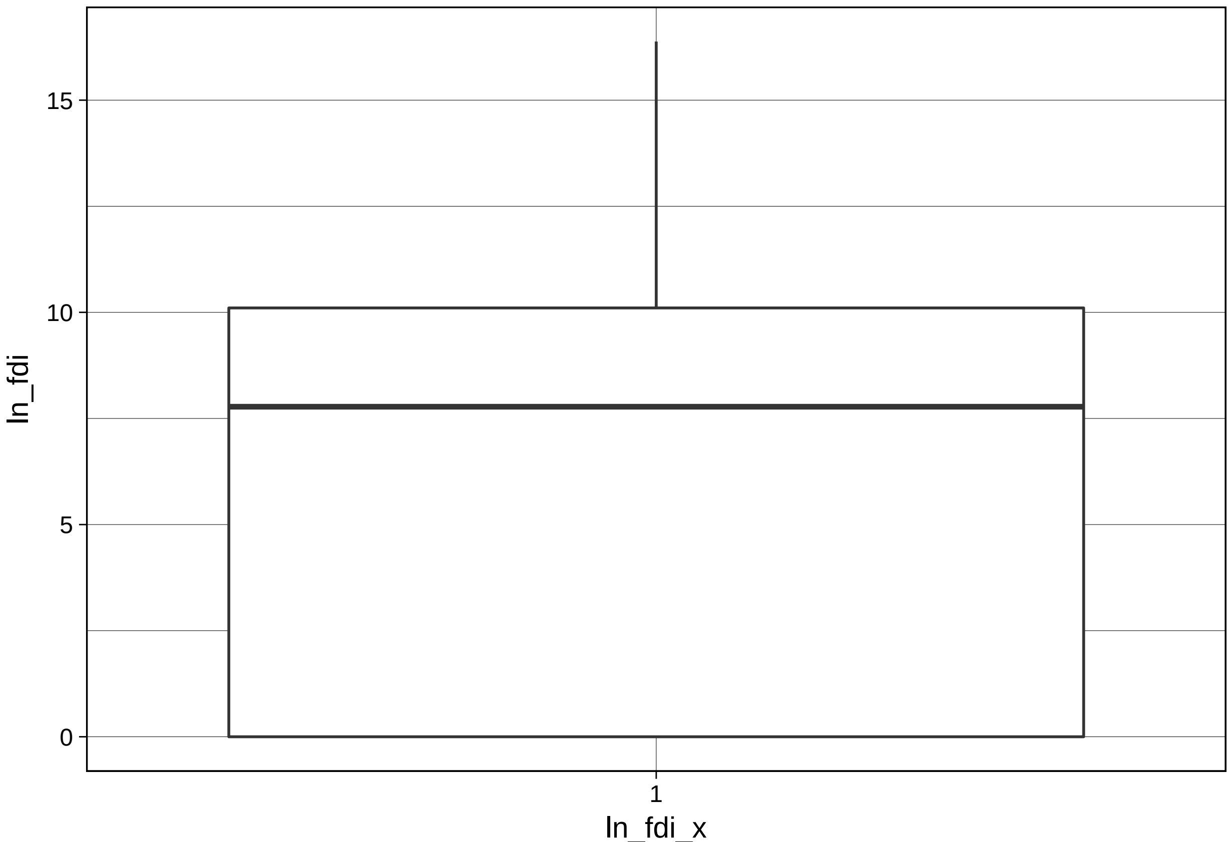
Boxplots for describing data

As King et al have argued, it is often efficient and desirable to visualize your data graphically rather than simply

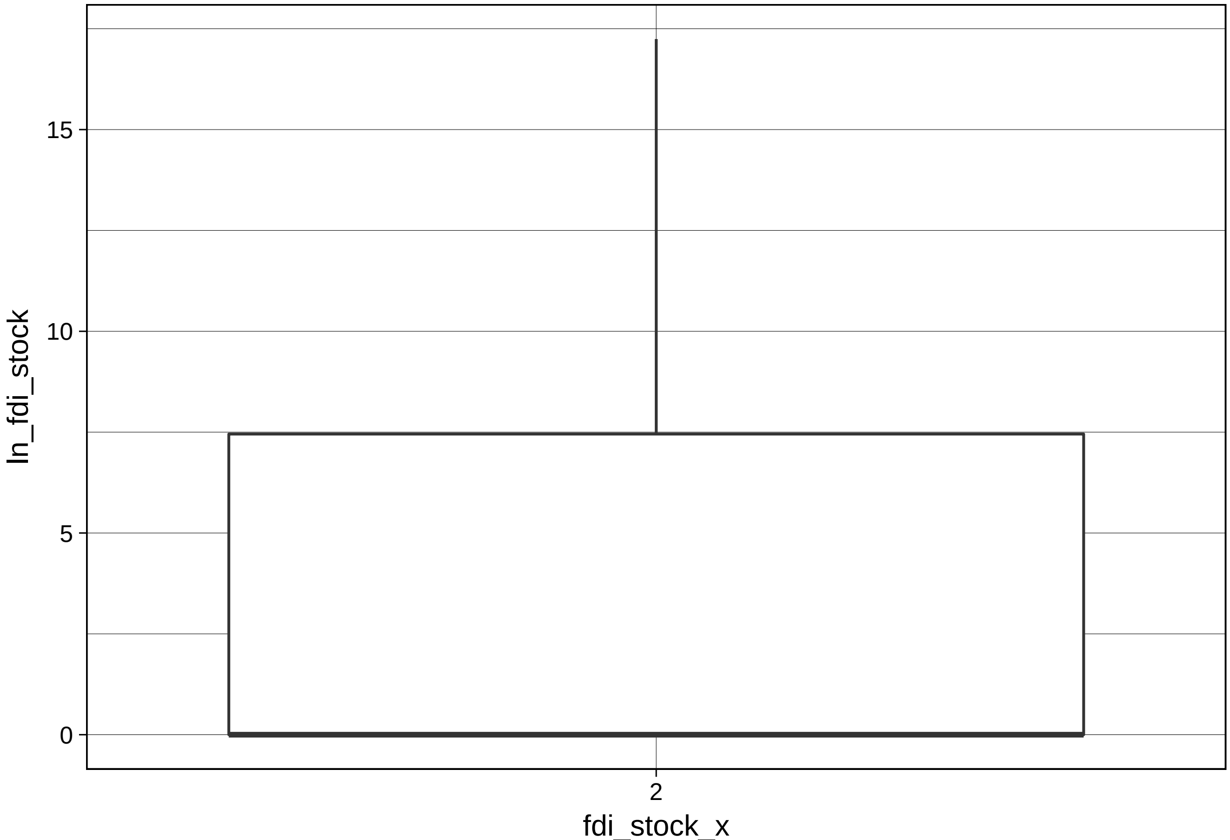
Now that we have created graphs for the descriptive stats of our data. I now choose to create boxplots and frequency graphs for my variables of interest. This is because these are efficient ways to describe and get a sense of our data.

In order to create a boxplot using ggplot. I need to first create new variables for the x-values. This is necessary because ggplot2 works with 2 variables in order to produce boxplots. It requires both an x and a y variable value. I achieve this by creating a new variable and filling it with just 1 value number and converting this variable to a categorical so that all the variables of interest in our data set can be represented on a single plot. I do this in the following code chunk

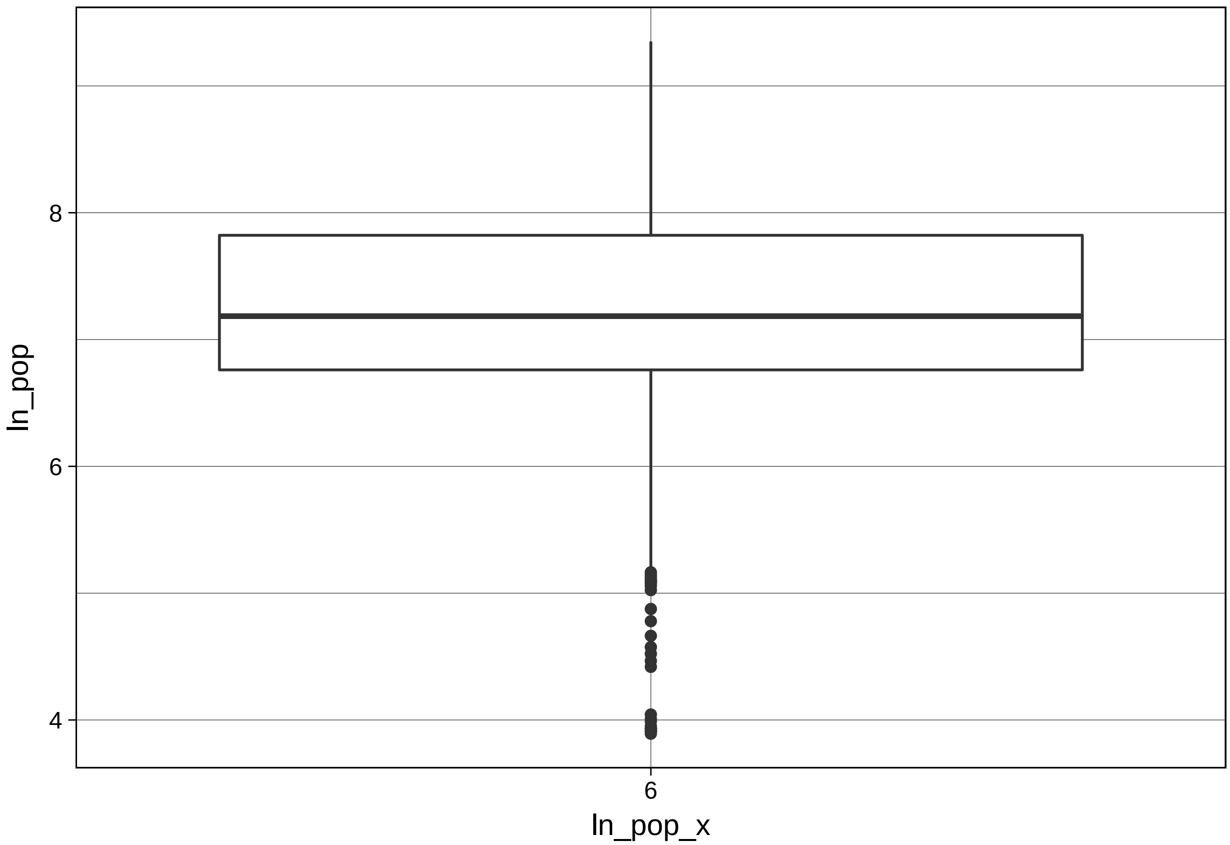
```
## Creating new categorical variables for the x axis

data_russia$ln_fdi_x <-  rep(1, each = 1722) %>% as.factor()
data_russia$fdi_stock_x <- rep(2, each = 1722) %>% as.factor()
data_russia$ln_grppc_x <- rep(3, each = 1722)  %>% as.factor()
data_russia$ln_trade_x <- rep(4, each = 1722) %>% as.factor()
data_russia$ln_roaddensity_x <- rep(5, each = 1722)  %>% as.factor()
data_russia$ln_pop_x <- rep(6, each = 1722) %>% as.factor()
data_russia$empl_high_x <- rep(6, each = 1722) %>% as.factor()

## Boxplot of FDI

ln_fdi_boxplot <- ggplot(subset(data_russia, !is.na(fdi)),
                    aes(x = ln_fdi_x, y = ln_fdi)) +
  geom_boxplot() +
  theme_linedraw(); ln_fdi_boxplot
```



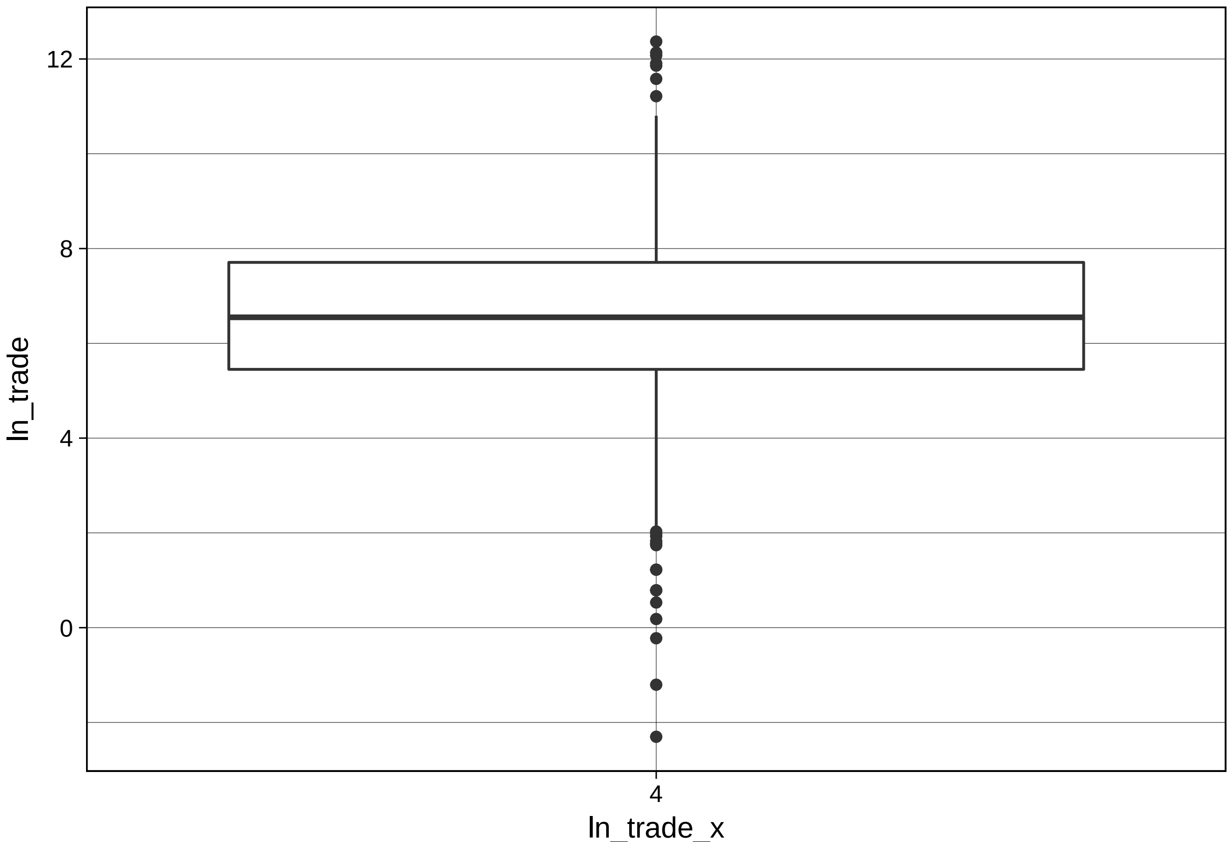

```
## Boxplot of FDI stock x

fdi_stock_boxplot <- ggplot(data_russia,
                            aes(x = fdi_stock_x,
                                y = ln_fdi_stock)) +
  geom_boxplot() +
  theme_linedraw(); fdi_stock_boxplot
```



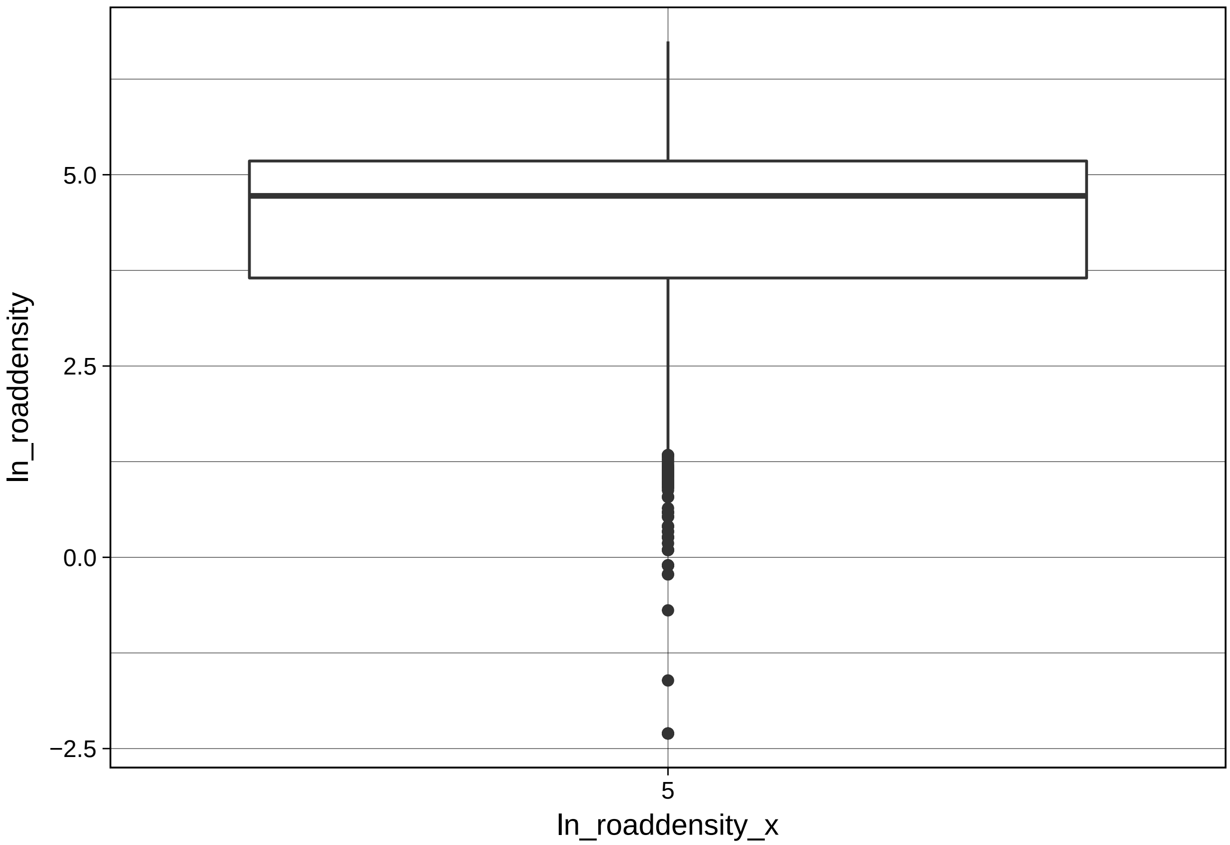

```
## Boxplot of ln_pop

ln_pop_boxplot <- ggplot(subset(data_russia, !is.na(ln_pop)),
                             aes(x = ln_pop_x,
                                 y = ln_pop)) +
  geom_boxplot() +
  theme_linedraw();ln_pop_boxplot
```



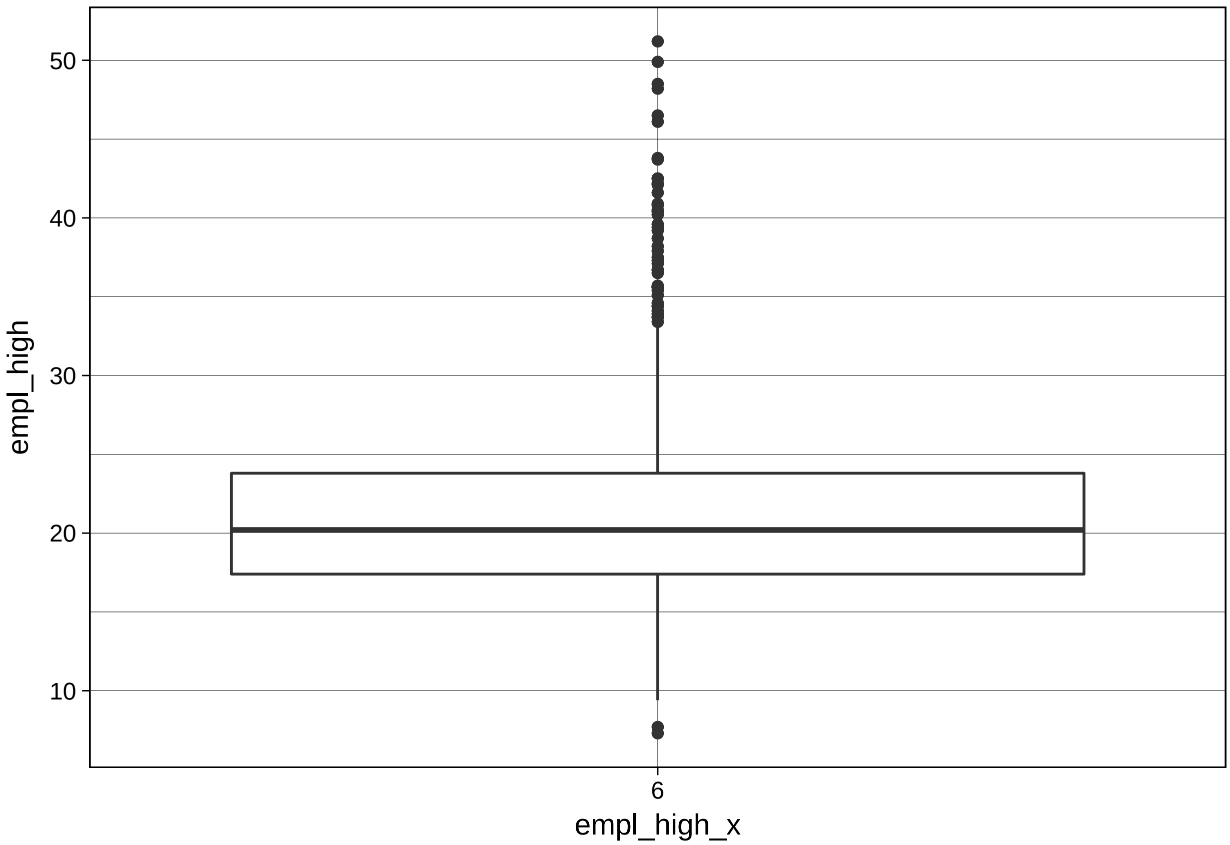

```
## Box plot of Trade

ln_trade_boxplot <- ggplot(subset(data_russia, !is.na(ln_trade)),
                           aes(x = ln_trade_x,
                               y = ln_trade)) +
  geom_boxplot() +
  theme_linedraw(); ln_trade_boxplot
```




```
## Box plot of roaddensity

ln_roaddensity_boxplot <- ggplot(subset(data_russia, !is.na(ln_roaddensity)),
                              aes(x = ln_roaddensity_x,
                                  y = ln_roaddensity)) +
  geom_boxplot() +
  theme_linedraw();ln_roaddensity_boxplot
```




```
## Box plot of empl_high

ln_empl_boxplot <- ggplot(subset(data_russia, !is.na(empl_high)),
                          aes(x = empl_high_x,
                              y = empl_high)) +
  geom_boxplot() +
  theme_linedraw();ln_empl_boxplot
```



Task 1.3

Move from a cross-regions time-series dataset to a cross-regions dataset. This makes it easier to visualize the fitted and actual values.

Solution 1.3

While there are several ways to break down the time-series dataset by region to remove the temporal dimension. I opt for the approach that was provided in the .Rmarkdown file that was provided by Prof. Baccini as part of the course materials. Through the following code chunk I break down the time-series dataset into a cross-regions dataset that breaks down the time dimension of the data.

Note that the code in the following code chunk does not deal with NAs in the dataset properly and returns a warning in the console output. A more elegant solution to this problem has been underlined in the subsequent code chunk. However, I opt and demonstrate this code as it was the recommended way to proceed with this task.

Collapsing time dimension (Class Approach)

```
# Collapse time dimensions

russia_region <- data_russia %>%
  group_by(name) %>%
  summarise_all(mean, na.rm = TRUE)
```

Collapsing time dimension (Alternative Approach)

```
data_region <- data_russia %>%
  group_by(region, name) %>%
  summarise_if(is.numeric, funs(mean(., na.rm = TRUE),
                                median(.,na.rm = TRUE))) %>%
  set_names(~sub('_mean', '', .x))
```

Now that we have the cross-regions dataset that we desired. Let's visualize the fitted values vs. the actual value plot. In order to do that, however, we would first have to re-estimate a linear model on the smaller cross-regions dataset that only includes the observations with the collapsed time-dimension.

Running and OLS regression

I first run an OLS regression which regresses the log of population, trade, gross regional product, road density, and people employed in the workforce with higher education onto the log of fdi.

```
## Let's take a look at the dataset that we have created

glimpse(russia_region)
```

```
## Observations: 82
## Variables: 77
## $ name            <chr> "Altai Krai", "Altai Republic", "Amur Oblast"...
## $ id              <dbl> 3, 4, 5, 6, 7, 9, 10, 12, 13, 14, 15, 18, 19,...
## $ region          <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ year            <dbl> 2000, 2000, 2000, 2000, 2000, 2000, 2000, 200...
## $ capital         <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ feddistrict     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ economzone      <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ climatezone     <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ arctic          <dbl> 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, ...
## $ area            <dbl> 169100, 92600, 363700, 587400, 44100, 27100, ...
## $ distance        <dbl> 3430, 3938, 8289, 1261, 1411, 681, 383, 1782,...
## $ airport         <dbl> 2, 1, 3, 12, 1, 2, 1, 1, 2, 5, 1, 3, 1, 0, 1,...
## $ intairport      <dbl> 1, 0, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2, 1, 0, 1, ...
## $ republic        <dbl> 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, ...
## $ border          <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, ...
## $ sez             <dbl> 0.1904762, 0.1904762, 0.0000000, 0.0000000, 0...
## $ oilprice        <dbl> 33.71905, 33.71905, 33.71905, 33.71905, 33.71...
## $ pop             <dbl> 2606.00000, 202.04762, 957.00000, 1410.52381,...
## $ census          <dbl> 2616424.3, 204557.5, 927730.7, 1377948.0, 100...
## $ ruspopul        <dbl> 2367370.0, 115656.0, 839521.0, 1284656.3, 677...
## $ russhare        <dbl> 91.13333, 57.05000, 91.10000, 94.03333, 69.83...
## $ city            <dbl> 53.60952, 25.93333, 66.04762, 73.93809, 67.01...
## $ mw              <dbl> 1136.7619, 1091.8095, 1049.6190, 1097.3810, 1...
## $ young           <dbl> 20.2300, 27.6150, 22.6050, 20.8500, 21.7150, ...
## $ adult           <dbl> 59.690, 57.680, 62.120, 61.135, 59.640, 57.78...
## $ old             <dbl> 20.0800, 14.7050, 15.2750, 18.0150, 18.6450, ...
## $ birth           <dbl> 10.295238, 16.500000, 11.666667, 10.338095, 1...
## $ mort            <dbl> 14.295238, 13.033333, 13.490476, 14.609524, 1...
## $ infmort         <dbl> 12.913333, 19.533333, 19.420000, 12.093333, 1...
## $ unemp           <dbl> 10.238889, 12.716667, 10.561111, 9.188889, 11...
## $ wagepc          <dbl> 204.7450, 208.3045, 317.1353, 320.5058, 224.8...
## $ gini            <dbl> 0.3677500, 0.3410000, 0.3508750, 0.3415625, 0...
## $ subsidy         <dbl> 18.586667, 0.937250, 14.750000, 7.519833, 3.7...
## $ brevenue        <dbl> 844.2748, 135.7202, 502.1945, 688.4958, 366.5...
```

```
## $ btransfers       <dbl> 531.21882, 143.60036, 266.63245, 262.46446, 1...
## $ taxes            <dbl> 513.7508, 83.7630, 328.3257, 494.6646, 440.73...
## $ bsocial          <dbl> 609.10708, 112.84273, 363.34008, 510.70827, 2...
## $ beducation       <dbl> 328.38943, 69.64486, 194.65143, 286.62728, 13...
## $ bhealthcare      <dbl> 185.52386, 32.76029, 121.67086, 155.53400, 10...
## $ bsecurity        <dbl> 75.4646, 10.2846, 47.0266, 55.4726, 29.6772, ...
## $ grp              <dbl> 4236.6100, 294.6219, 2439.7456, 4838.3748, 20...
## $ grppc            <dbl> 1661.484, 1440.443, 2701.570, 3674.223, 2076....
## $ realgrpgrowth    <dbl> 3.250000, 3.878571, 2.500000, 6.328571, 5.028...
## $ trade            <dbl> 664.55714, 119.28572, 211.76429, 1474.67857, ...
## $ fdi              <dbl> 10914.4095, 92.1000, 36940.4187, 75240.0481, ...
## $ roaddensity      <dbl> 95.614286, 33.400000, 19.800000, 16.800000, 6...
## $ crimeshare       <dbl> 2105.0000, 2369.0000, 2132.9524, 2170.3333, 2...
## $ invrisk          <dbl> 55.933333, 28.866667, 56.200000, 45.066667, 3...
## $ demrating        <dbl> 25.66667, 27.33333, 26.00000, 37.00000, 27.33...
## $ spatial_fdi      <dbl> 26.81250, 23.70000, 37.10000, 59.65000, 17.61...
## $ bcwin            <dbl> 0.0000000, 0.0000000, 0.3333333, 0.3333333, 0...
## $ party_strength   <dbl> 0.27422901, 0.32323232, 0.21111111, 0.1969056...
## $ corruption       <dbl> 2.466667, 2.800000, 2.600000, 3.000000, 2.800...
## $ deficit          <dbl> -285.56592, -76.70950, -149.32113, -131.35122...
## $ empl_high        <dbl> 18.91333, 21.12667, 18.88667, 19.78000, 18.26...
## $ firms            <dbl> 17917.000, 7628.000, 42771.905, 12017.714, 13...
## $ sme_n            <dbl> 14.1400002, 1.3000000, 4.3200000, 5.5133334, ...
## $ priv_n           <dbl> 134.00000, 19.00000, 61.09091, 68.11765, 53.3...
## $ ffirms_n         <dbl> 57.307692, 5.666667, 40.615385, 63.923077, 10...
## $ fdi_stock        <dbl> 1.730514e+04, 3.137000e+02, 9.122461e+04, 4.9...
## $ prod_oil         <dbl> NaN, NaN, NaN, 5.009000e+03, 2.611722e+03, Na...
## $ prod_gas         <dbl> NaN, NaN, NaN, 325.375000, 7400.333333, NaN, ...
## $ ln_pop           <dbl> 7.865101, 5.308253, 6.860101, 7.247774, 6.914...
## $ ln_grppc         <dbl> 7.262585, 7.131989, 7.765570, 8.028517, 7.503...
## $ ln_trade         <dbl> 6.3688660, 4.6766030, 5.1481712, 7.0827360, 6...
## $ ln_grp           <dbl> 8.212552, 5.541186, 7.684211, 8.333686, 7.514...
## $ sqrt_fdi         <dbl> 71.983094, 3.284596, 126.766272, 186.419054, ...
## $ ln_fdi           <dbl> 6.4715047, 0.9319396, 6.9455204, 7.6719885, 6...
## $ ln_fdi_stock     <dbl> 3.0416203, 1.0804405, 3.6011294, 3.4396213, 2...
## $ ln_roaddensity   <dbl> 4.5529100, 3.4863607, 2.9766632, 2.7561000, 4...
## $ ln_fdi_x         <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ fdi_stock_x      <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ ln_grppc_x       <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ ln_trade_x       <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ ln_roaddensity_x <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ ln_pop_x         <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ empl_high_x      <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
```

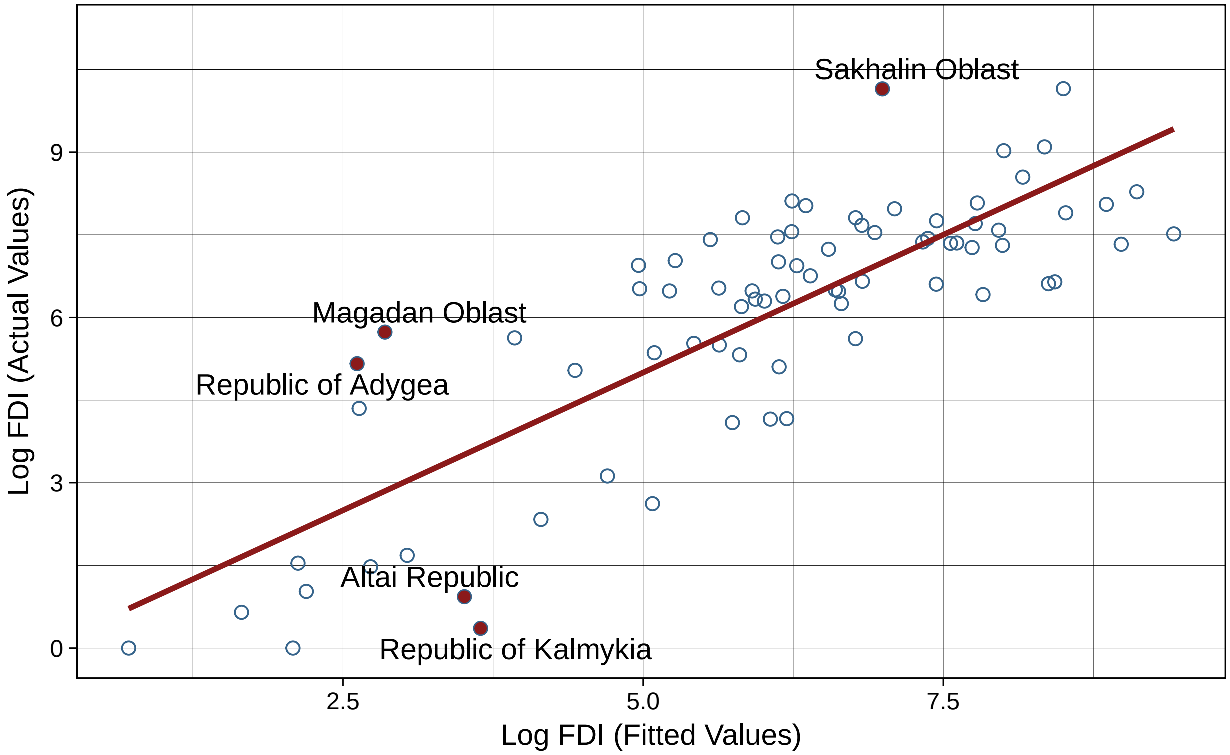
```
## Now let's define our model for the cross-regions data

model_1_region <- lm(ln_fdi ~ ln_pop + ln_trade + ln_roaddensity + empl_high,
                     data = russia_region, na.action = na.exclude)

table1_region <- screenreg(list(model_1_region), custom.model.names = "Model 1 on regional dataset",
          custom.note = "Standard errors in parentheses. *p < 0.05",
          stars = 0.05); table1_region
```

```
##
## ===========================================
```

```
##                  Model 1 on regional dataset
## -------------------------------------------
## (Intercept)      -0.49
##                  (1.84)
## ln_pop            0.46
##                  (0.32)
## ln_trade          0.94 *
##                  (0.15)
## ln_roaddensity    0.04
##                  (0.15)
## empl_high        -0.14 *
##                  (0.05)
## -------------------------------------------
## R^2               0.67
## Adj. R^2          0.65
## Num. obs.        78
## RMSE              1.39
## ===========================================
## Standard errors in parentheses. *p < 0.05
```

```r
##### Getting the fitted vs. residuals plot using the Baccini regional dataset

## Creating and storing the fitted and residual values into the Baccini regional dataset

russia_region$model1_fitted <- predict(model_1_region, newdata = russia_region)
russia_region$model1_residual <- resid(model_1_region)

## Now let's plot the fitted vs actual values!

figure_2 <- ggplot(russia_region, aes(model1_fitted, ln_fdi)) +
  geom_point(size = 2, shape = 1, colour = "steelblue4") +
  geom_smooth(method = "lm", se = FALSE, colour = "firebrick4") +
  xlab("Log FDI (Fitted Values)") +
  ylab("Log FDI (Actual Values)") +
  ggtitle("Figure 2: Fitted vs. Actual Values \n (Linear Model 1)") +
  geom_text_repel(data = subset(russia_region,
                                model1_residual > 2.5 | model1_residual < -2.5),
                  mapping = aes(label = name)) +
  geom_point(data = subset(russia_region,
                           model1_residual > 2.5 | model1_residual < -2.5),
             aes(color = "Outliers"), color = "firebrick4") +
  theme_linedraw(); figure_2
```

Figure 2: Fitted vs. Actual Values
(Linear Model 1)



Task

Focus on cases off-the-line and select variables that might help build a better model. Explain the rationale behind the selection of these variables.

Solution

I follow Lieberman (2005)'s suggestions in proceeding with 'nested analysis' using mixed methods. Lieberman (2005) suggests that to perform nested analysis one first runs a preliminary Large N Analysis (LNA) as was done in the previous chunk of code. The results from the model indicate that that our model is poor and there are a large number of values 'off-the-line'. Following Murali (2011) and Lieberman (2005), I opt to use a model building approach. Using Small N Analysis (SNA), for model building means that I should focus on 'off-the-line' cases (Lieberman 2005). To be sure, off-the-line cases can be selected on the basis of the magnitude of the residuals. In other words, the distance between the actual and predicted values of the outcome variable (Note that off the line cases cannot have a residual value of zero, as that implies that the case is on-the-line).

More specifically, our model predicts the log of FDI using measures of market size, income, infrastructure, and human capital as explanatory measures. I select cases in Figure 2 above on the basis of regions that have the greatest residual values. Following the advice of Lieberman (2005), I will use these 'off-the-line' cases to examine if there are other explanatory variables whose inclusion can improve the goodness-of-fit of our baseline model.

Based on the aforementione criteria, I choose to focus on five cases of russian regions namely: 1) "Sakhalin Oblast", 2) Magadan Oblast, 3) Republic of Adygea, 4) Altar Republic and 5) Republic of Kalmykia.

Note that uptil now in our baseline model earlier we only predicted FDI. However, the assignment asks us to use both FDI and FDI stock as outcome variables. Therefore, in the following code chunk I re-run the baseline model with a different outcome variable (i.e. FDI stock). Figure 3 below reveals that two out of the five outliers in the FDI baseline (Model 1) are also outliers in the baseline with FDI stock as outcome variable (Model 2). This gives us further evidence that the 'off-the-line' cases choosen under Model 1 are suitable for theory building. However, a very preliminary glance at table 2 in the code chunk below shows that Model 1 explains more of the variance in FDI (adjusted r^2 value of 0.67) than Model 2 explains the variance in FDI stock (adjusted r^2 value of 0.52). As such, i will focus on FDI inflows as an outcome variable.

```
## Model 2 - Baseline explanatory variables on FDI stock

## Now let's define our model for the cross-regions data

model_2_region <- lm(ln_fdi_stock ~ ln_pop + ln_trade + ln_roaddensity + empl_high,
                     data = russia_region, na.action = na.exclude)

table2_region <- screenreg(list(model_1_region, model_2_region), custom.model.names = c("Model 1 (DV: FI
          custom.note = "Standard errors in parentheses. *p < 0.05",
          stars = 0.05); table2_region
```

```
##
## ============================================================
##                 Model 1 (DV: FDI)  Model 2 (DV: FDI stock
## ------------------------------------------------------------
## (Intercept)     -0.49               1.92 *
##                 (1.84)             (0.83)
## ln_pop           0.46              -0.07
##                 (0.32)             (0.14)
## ln_trade         0.94 *             0.40 *
##                 (0.15)             (0.07)
## ln_roaddensity   0.04               0.06
##                 (0.15)             (0.07)
## empl_high       -0.14 *            -0.06 *
##                 (0.05)             (0.02)
## ------------------------------------------------------------
## R^2              0.67               0.55
## Adj. R^2         0.65               0.52
## Num. obs.       78                 78
## RMSE             1.39               0.63
## ============================================================
## Standard errors in parentheses. *p < 0.05
```

```
##### Getting the fitted vs. residuals plot

## Creating and storing the fitted and residual values into the Baccini regional dataset

russia_region$model2_fitted <- predict(model_2_region, newdata = russia_region)
russia_region$model2_residual <- resid(model_2_region)

## taking a look at the residual values so as to specify outlier criteria

stat.desc(russia_region$model2_residual)
```

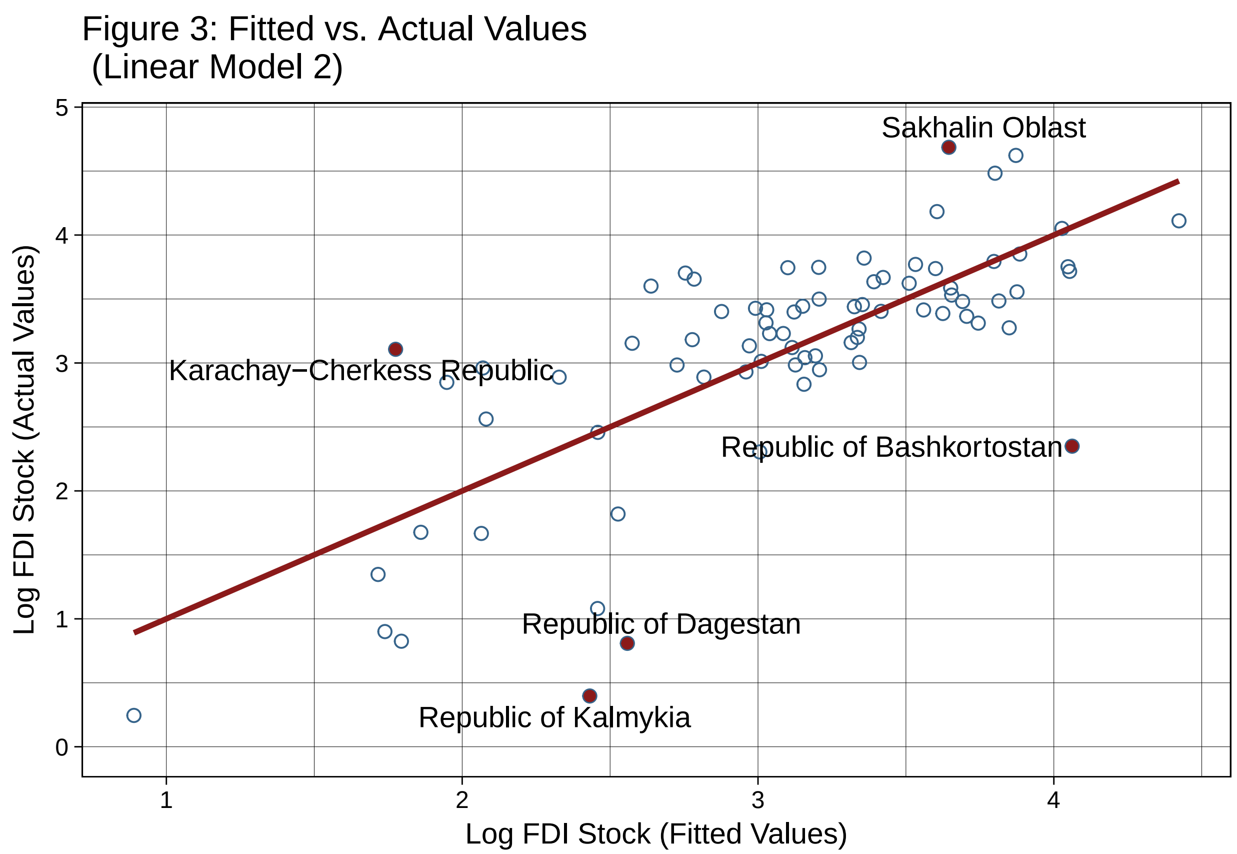
```
##       nbr.val      nbr.null        nbr.na           min           max
##  7.800000e+01  0.000000e+00  4.000000e+00 -2.034423e+00  1.331896e+00
##         range           sum        median          mean       SE.mean
```

```
##  3.366319e+00 -3.417405e-16  4.118750e-04 -4.402139e-18  6.923867e-02
##  CI.mean.0.95           var       std.dev      coef.var
##  1.378718e-01  3.739315e-01  6.114994e-01 -1.389096e+17
```

```r
glimpse(russia_region$model2_residual)
```

```
##  num [1:82] -0.117 -1.377 0.963 0.114 -0.142 ...
```

```r
## Now let's plot the fitted vs actual values!

figure_3 <- ggplot(russia_region, aes(model2_fitted, ln_fdi_stock)) +
  geom_point(size = 2, shape = 1, colour = "steelblue4") +
  geom_smooth(method = "lm", se = FALSE, colour = "firebrick4") +
  xlab("Log FDI Stock (Fitted Values)") +
  ylab("Log FDI Stock (Actual Values)") +
  ggtitle("Figure 3: Fitted vs. Actual Values \n (Linear Model 2)") +
  geom_text_repel(data = subset(russia_region,
                                model2_residual > 1 | model2_residual < -1.7),
                  mapping = aes(label = name)) +
  geom_point(data = subset(russia_region,
                           model2_residual > 1 | model2_residual < -1.7),
             aes(color = "Outliers"), color = "firebrick4") +
  theme_linedraw(); figure_3
```



Including additional variables

Corruption

I suspect that one of the variables that might be missing in Model 1 (baseline) is *corruption*. Corruption may influence the FDI inflows as firms might be potentially dissuaded from investing in a particular region that has a higher degree of corruption that others. Conversely, it is also possible that corrupt regions might attract more investment as public officials might offer these firms incentives, such as tax breaks or tax-exemptions in return for the bribes they receive from these firms. Therefore, there are clear theoretical reasons why we expect that corruption might influence FDI inflows. Note that the missing value is for the region Checheneya so this does not affect our selected cases.

Share of Russian Population

In addition to corruption, I also include *russhare* as a potential omitted variable in our baseline model. The politics of Russia lead us to suspect that regions with a lower share of russian population might not comprise a large share of the winning coalition of the politicians. Even apart from the point of view of winning coalitions, there is reason to believe that regions with a lower share of russian population might not be the foci of the same developmental goals of the government as regions with a high share of the russian population. Ultimately, this is something that will influence FDI inflow into these regions.

Share of elderly population

I also include the variable *old* as a region with a high number of elderly people might translate into a decreased availability of labour force and consumers. This, in turn, is something that detract potential FDI inflows into the region.

Running the new model

In the following code chunk I show the descriptive stats for each of the new variables.

```
## log transforming corruption

russia_region$ln_corruption <- log(russia_region$corruption)

## Descriptive for corruption
stat.desc(russia_region$corruption)
```

```
##      nbr.val     nbr.null       nbr.na          min          max
##  81.00000000   0.00000000   1.00000000   2.00000000   3.80000000
##        range          sum       median         mean      SE.mean
##   1.80000000 222.45190476   2.73333333   2.74631981   0.04734770
## CI.mean.0.95          var      std.dev     coef.var
##   0.09422492   0.18158615   0.42612927   0.15516374
```

```
## Descriptive for russhare
stat.desc(russia_region$russhare)
```

```
##      nbr.val     nbr.null       nbr.na          min          max
##   82.0000000    0.0000000    0.0000000    2.8000000   97.0333328
##        range          sum       median         mean      SE.mean
##   94.2333328 6266.6500018   87.8833351   76.4225610    2.7137381
## CI.mean.0.95          var      std.dev     coef.var
##    5.3994876  603.8787029   24.5739436    0.3215535
```

```
## Descriptive for old
stat.desc(russia_region$old)
```

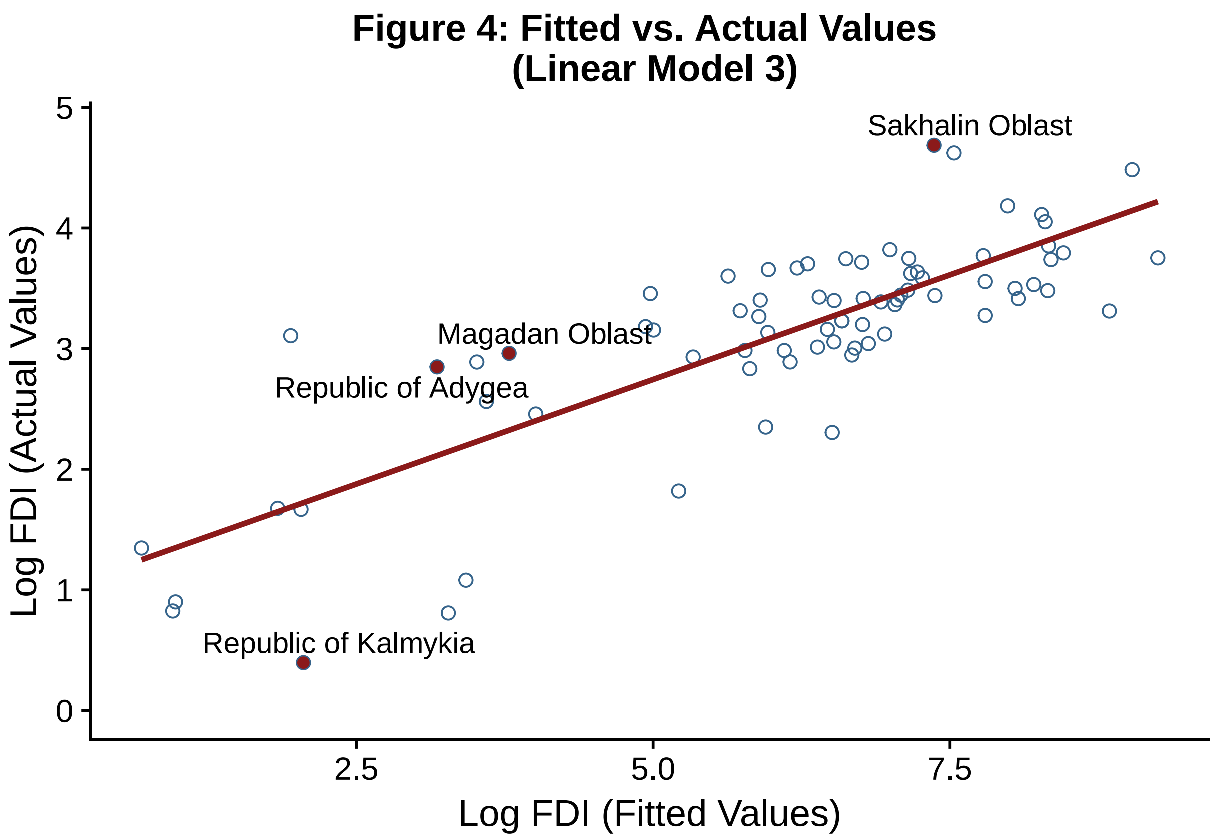
```
##      nbr.val     nbr.null       nbr.na          min          max
##   82.0000000    0.0000000    0.0000000    4.9750000   26.7250002
##        range          sum       median         mean      SE.mean
##   21.7500001 1551.2224994   19.9175000   18.9173476    0.5634320
## CI.mean.0.95          var      std.dev     coef.var
##    1.1210530   26.0313616    5.1020938    0.2697045
```

```r
## Now let's redefine our previous model by including additional variables to see if it improves the ori
model_3_region <- lm(ln_fdi ~ ln_pop + ln_trade + ln_roaddensity + empl_high + ln_corruption + russhare
                     data = russia_region, na.action = na.exclude)

table3_region <- screenreg(list(model_1_region, model_3_region),
                           custom.model.names = c("Model 1 Baseline",
                                                  "Model 3 with additional controls"),
          custom.note = "Standard errors in parentheses. *p < 0.05",
          stars = 0.05);table3_region
```

```
##
## =================================================================
##                 Model 1 Baseline  Model 3 with additional controls
## -----------------------------------------------------------------
## (Intercept)     -0.49             -6.56 *
##                 (1.84)            (1.96)
## ln_pop           0.46              0.69 *
##                 (0.32)            (0.30)
## ln_trade         0.94 *            0.51 *
##                 (0.15)            (0.16)
## ln_roaddensity   0.04              0.00
##                 (0.15)            (0.25)
## empl_high       -0.14 *           -0.04
##                 (0.05)            (0.04)
## ln_corruption                      2.95 *
##                                   (1.06)
## russhare                           0.03 *
##                                   (0.01)
## old                                0.00
##                                   (0.08)
## -----------------------------------------------------------------
## R^2              0.67              0.77
## Adj. R^2         0.65              0.74
## Num. obs.       78                77
## RMSE             1.39              1.16
## =================================================================
## Standard errors in parentheses. *p < 0.05
```

```r
## Creating and storing the fitted and residual values into the Baccini regional dataset

russia_region$model3_fitted <- predict(model_3_region, newdata = russia_region)
russia_region$model3_residual <- resid(model_3_region)

## Now let's plot the fitted vs actual values!

## Now let's plot the fitted vs actual values!

figure_4 <- ggplot(russia_region, aes(model3_fitted, ln_fdi_stock)) +
  geom_point(size = 2, shape = 1, colour = "steelblue4") +
  geom_smooth(method = "lm", se = FALSE, colour = "firebrick4") +
  xlab("Log FDI (Fitted Values)") +
  ylab("Log FDI (Actual Values)") +
  ggtitle("Figure 4: Fitted vs. Actual Values \n (Linear Model 3)") +
```

```r
  geom_text_repel(data = subset(russia_region,
                              name == "Magadan Oblast" | name == "Republic of Adygea" |
                                name == "Sakhalin Oblast" | name == "Altar Republic" |
                                name == "Republic of Kalmykia"),
                  mapping = aes(label = name)) +
  geom_point(data = subset(russia_region,
                          name == "Magadan Oblast" | name == "Republic of Adygea" |
                                name ==  "Sakhalin Oblast" | name == "Altar Republic" |
                                name == "Republic of Kalmykia"), color = "firebrick4"); figure_4
```




```r
figure_2; figure_4
```

Figure 2: Fitted vs. Actual Values
(Linear Model 1)

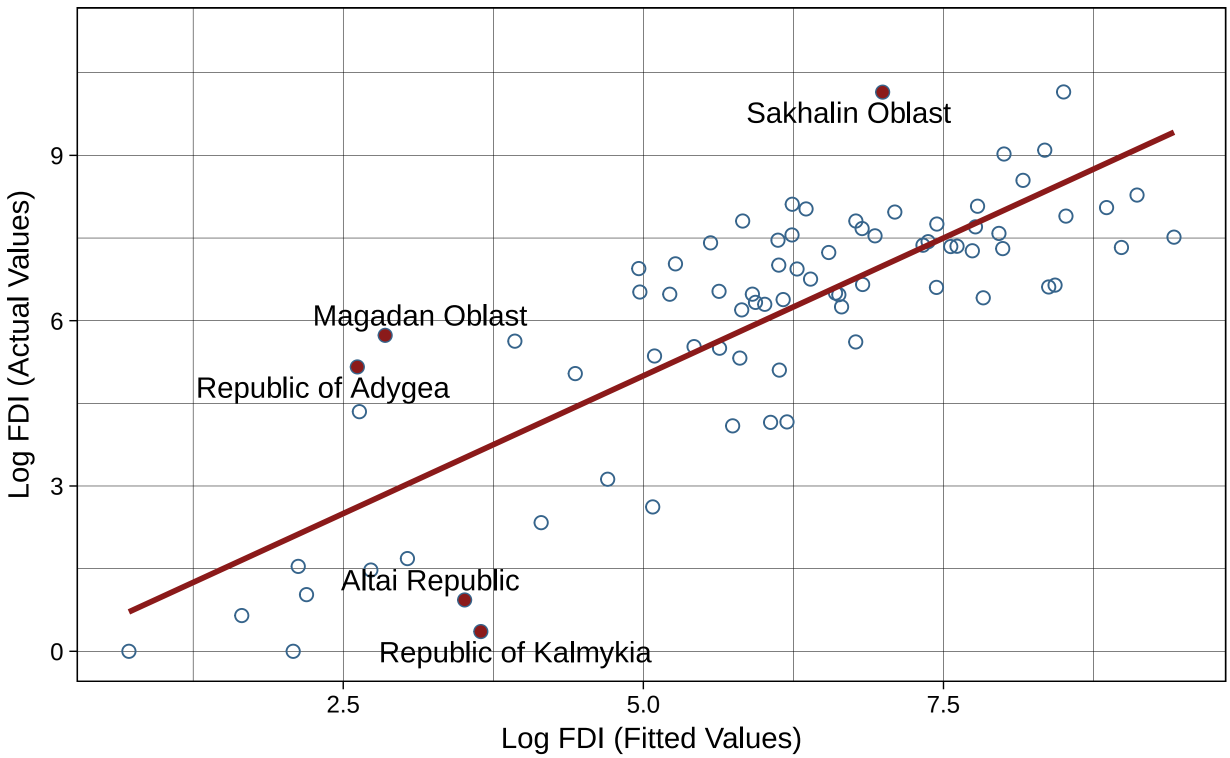


**Figure 4: Fitted vs. Actual Values
(Linear Model 3)**

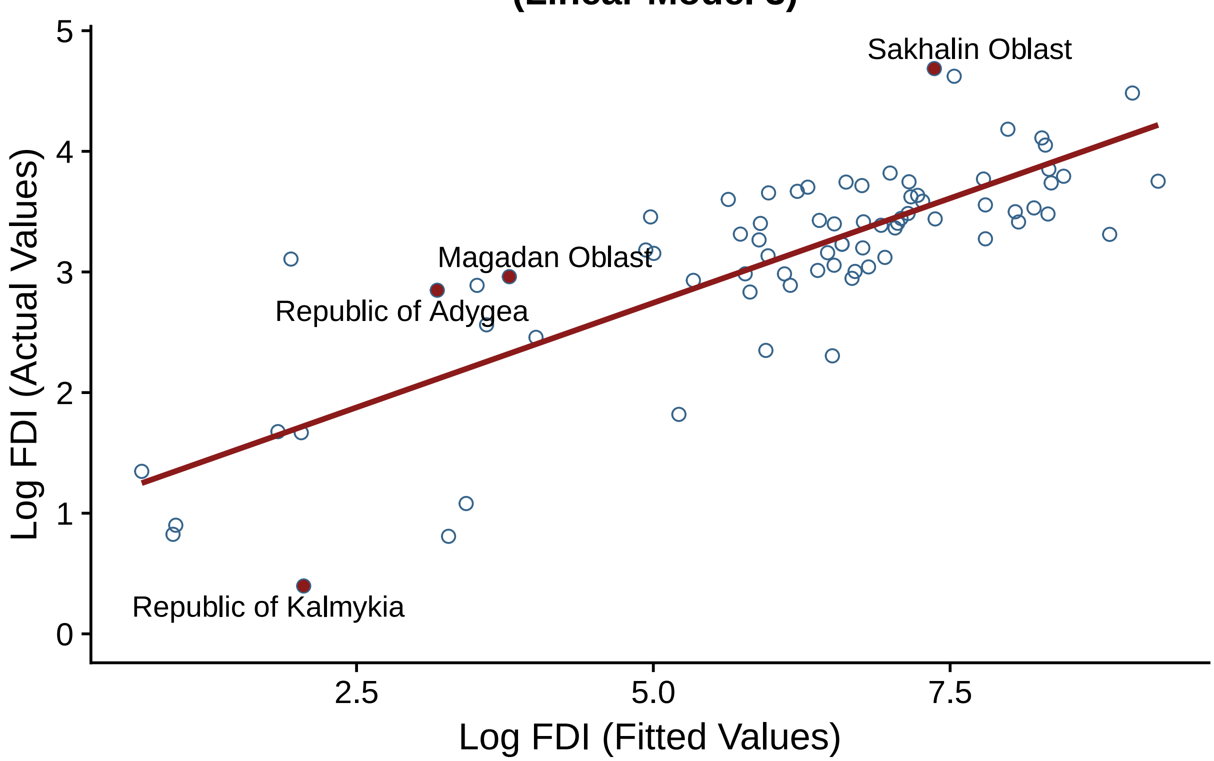


Figure 4 above plots the fitted vs actual values of FDI. While this is hard to visualize this at a cursory glance, a closer inspection of Figure 2 and Figure 4 indicates that the inclusion of the control variables somewhat decreased the residual values for these cases. For instance, note that y-axis values for Figures 2 and 4 have different limits. However, further goodness of fit tests are necessary as the evidence is weak to make a definitive judgement on whether the inclusion of additional control variables improved the model fitness for the 'off-the-line' cases.

Task 1.4

Build a dichotomous measure of FDI. Note that you can chose whatever threshold to move from a continuous to a dummy variable as long as you are able to motivate it. Replicate the previous steps using the dummy dependent variable. How would you select cases on-the-line and off-the-line with a dummy dependent variable?

Solution 1.4

Let's specify the logit regression

I create a dichotomous measure for FDI by following a simple critera. Namely, if the log of FDI has a value greater than the median value of FDI for our data, than the dichotomous FDI will be specified as 1. Conversely, if the regional FDI is lower than the median FDI of all the regions, than the dichotomous FDI variable is recorded as zero.

```
## First let's build a dichotomous measure of the FDI variable

russia_region$fdi_dummy <- ifelse(russia_region$ln_fdi >= median(russia_region$ln_fdi), 1, 0) %>% as.fa

## Adding a dummy for false positives and negatives

#russia_region$fdi_dummy_false <- ifelse(russia_region$model1_fitted_logit >= 0.5, 1, 0) %>% as.factor(

# Logistic regression

formula_logit_1 <- fdi_dummy ~ ln_pop + ln_trade + ln_roaddensity + empl_high

model_1_logit <- glm(formula_logit_1, family = binomial,
               data = russia_region)

screenreg(list(model_1_logit))
```

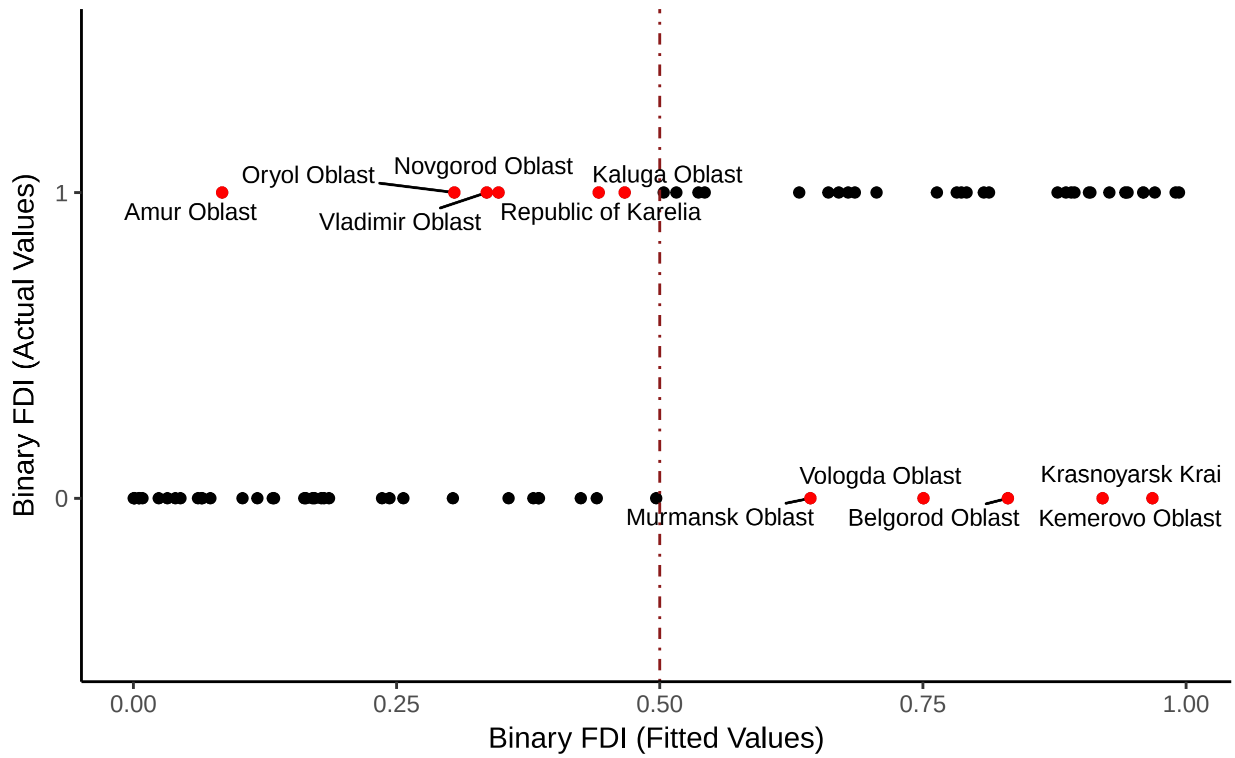
```
##
## ==========================
##                 Model 1
## --------------------------
## (Intercept)     -14.66 **
##                  (4.94)
## ln_pop            0.37
##                  (0.71)
## ln_trade          1.47 ***
##                  (0.44)
## ln_roaddensity   -0.02
##                  (0.31)
## empl_high         0.12
##                  (0.11)
## --------------------------
```

```
## AIC               74.41
## BIC               86.19
## Log Likelihood   -32.20
## Deviance          64.41
## Num. obs.         78
## ==========================
## *** p < 0.001, ** p < 0.01, * p < 0.05
```

```
## Let's calculated the predicted values for the logit

russia_region$model1_fitted_logit <- predict(model_1_logit, russia_region, type = "response")

#russia_region$model1_residuals_logit <- residuals(model_1_logit)

russia_region$model1_probs_logit <- model_1_logit$family$linkinv(russia_region$model1_fitted_logit)

## let's plot the 2 x 2

figure_5 <- ggplot(russia_region, aes(model1_fitted_logit, fdi_dummy)) +
  geom_point() +
  geom_vline(xintercept = 0.5, linetype = "dotdash", color = "firebrick4") +
  labs(y = "Binary FDI (Actual Values)",
       x = "Binary FDI (Fitted Values)",
       title = "Figure 5: Plot of Actual versus Predicted Values for Model 4",
       subtitle = "The points in red represent cases that are false negatives and false positives") +
  theme_classic() +
  geom_text_repel(aes(label = ifelse(model1_fitted_logit >= 0.5 & fdi_dummy == 0,
                              as.character(name),'')),
            size = 3) +
  geom_text_repel(aes(label = ifelse(model1_fitted_logit <= 0.5 & fdi_dummy == 1,
                              as.character(name),'')),
            size = 3) +
   geom_point(data = russia_region[russia_region$name %in% c("Amur Oblast", "Oryol Oblast","Novgorod Obl
```

Figure 5: Plot of Actual versus Predicted Values for Model 4

The points in red represent cases that are false negatives and false positives




```r
glimpse(russia_region)
```

```
## Observations: 82
## Variables: 87
## $ name              <chr> "Altai Krai", "Altai Republic", "Amur Obla...
## $ id                <dbl> 3, 4, 5, 6, 7, 9, 10, 12, 13, 14, 15, 18, ...
## $ region            <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ year              <dbl> 2000, 2000, 2000, 2000, 2000, 2000, 2000, ...
## $ capital           <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ feddistrict       <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ economzone        <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ climatezone       <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ arctic            <dbl> 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ...
## $ area              <dbl> 169100, 92600, 363700, 587400, 44100, 2710...
## $ distance          <dbl> 3430, 3938, 8289, 1261, 1411, 681, 383, 17...
## $ airport           <dbl> 2, 1, 3, 12, 1, 2, 1, 1, 2, 5, 1, 3, 1, 0,...
## $ intairport        <dbl> 1, 0, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2, 1, 0, ...
## $ republic          <dbl> 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, ...
## $ border            <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, ...
## $ sez               <dbl> 0.1904762, 0.1904762, 0.0000000, 0.0000000...
## $ oilprice          <dbl> 33.71905, 33.71905, 33.71905, 33.71905, 33...
## $ pop               <dbl> 2606.00000, 202.04762, 957.00000, 1410.523...
## $ census            <dbl> 2616424.3, 204557.5, 927730.7, 1377948.0, ...
## $ ruspopul          <dbl> 2367370.0, 115656.0, 839521.0, 1284656.3, ...
## $ russhare          <dbl> 91.13333, 57.05000, 91.10000, 94.03333, 69...
## $ city              <dbl> 53.60952, 25.93333, 66.04762, 73.93809, 67...
## $ mw                <dbl> 1136.7619, 1091.8095, 1049.6190, 1097.3810...
```

```
## $ young              <dbl> 20.2300, 27.6150, 22.6050, 20.8500, 21.715...
## $ adult              <dbl> 59.690, 57.680, 62.120, 61.135, 59.640, 57...
## $ old                <dbl> 20.0800, 14.7050, 15.2750, 18.0150, 18.645...
## $ birth              <dbl> 10.295238, 16.500000, 11.666667, 10.338095...
## $ mort               <dbl> 14.295238, 13.033333, 13.490476, 14.609524...
## $ infmort            <dbl> 12.913333, 19.533333, 19.420000, 12.093333...
## $ unemp              <dbl> 10.238889, 12.716667, 10.561111, 9.188889,...
## $ wagepc             <dbl> 204.7450, 208.3045, 317.1353, 320.5058, 22...
## $ gini               <dbl> 0.3677500, 0.3410000, 0.3508750, 0.3415625...
## $ subsidy            <dbl> 18.586667, 0.937250, 14.750000, 7.519833, ...
## $ brevenue           <dbl> 844.2748, 135.7202, 502.1945, 688.4958, 36...
## $ btransfers         <dbl> 531.21882, 143.60036, 266.63245, 262.46446...
## $ taxes              <dbl> 513.7508, 83.7630, 328.3257, 494.6646, 440...
## $ bsocial            <dbl> 609.10708, 112.84273, 363.34008, 510.70827...
## $ beducation         <dbl> 328.38943, 69.64486, 194.65143, 286.62728,...
## $ bhealthcare        <dbl> 185.52386, 32.76029, 121.67086, 155.53400,...
## $ bsecurity          <dbl> 75.4646, 10.2846, 47.0266, 55.4726, 29.677...
## $ grp                <dbl> 4236.6100, 294.6219, 2439.7456, 4838.3748,...
## $ grppc              <dbl> 1661.484, 1440.443, 2701.570, 3674.223, 20...
## $ realgrpgrowth      <dbl> 3.250000, 3.878571, 2.500000, 6.328571, 5....
## $ trade              <dbl> 664.55714, 119.28572, 211.76429, 1474.6785...
## $ fdi                <dbl> 10914.4095, 92.1000, 36940.4187, 75240.048...
## $ roaddensity        <dbl> 95.614286, 33.400000, 19.800000, 16.800000...
## $ crimeshare         <dbl> 2105.0000, 2369.0000, 2132.9524, 2170.3333...
## $ invrisk            <dbl> 55.933333, 28.866667, 56.200000, 45.066667...
## $ demrating          <dbl> 25.66667, 27.33333, 26.00000, 37.00000, 27...
## $ spatial_fdi        <dbl> 26.81250, 23.70000, 37.10000, 59.65000, 17...
## $ bcwin              <dbl> 0.0000000, 0.0000000, 0.3333333, 0.3333333...
## $ party_strength     <dbl> 0.27422901, 0.32323232, 0.21111111, 0.1969...
## $ corruption         <dbl> 2.466667, 2.800000, 2.600000, 3.000000, 2....
## $ deficit            <dbl> -285.56592, -76.70950, -149.32113, -131.35...
## $ empl_high          <dbl> 18.91333, 21.12667, 18.88667, 19.78000, 18...
## $ firms              <dbl> 17917.000, 7628.000, 42771.905, 12017.714,...
## $ sme_n              <dbl> 14.1400002, 1.3000000, 4.3200000, 5.513333...
## $ priv_n             <dbl> 134.00000, 19.00000, 61.09091, 68.11765, 5...
## $ ffirms_n           <dbl> 57.307692, 5.666667, 40.615385, 63.923077,...
## $ fdi_stock          <dbl> 1.730514e+04, 3.137000e+02, 9.122461e+04, ...
## $ prod_oil           <dbl> NaN, NaN, NaN, 5.009000e+03, 2.611722e+03,...
## $ prod_gas           <dbl> NaN, NaN, NaN, 325.375000, 7400.333333, Na...
## $ ln_pop             <dbl> 7.865101, 5.308253, 6.860101, 7.247774, 6....
## $ ln_grppc           <dbl> 7.262585, 7.131989, 7.765570, 8.028517, 7....
## $ ln_trade           <dbl> 6.3688660, 4.6766030, 5.1481712, 7.0827360...
## $ ln_grp             <dbl> 8.212552, 5.541186, 7.684211, 8.333686, 7....
## $ sqrt_fdi           <dbl> 71.983094, 3.284596, 126.766272, 186.41905...
## $ ln_fdi             <dbl> 6.4715047, 0.9319396, 6.9455204, 7.6719885...
## $ ln_fdi_stock       <dbl> 3.0416203, 1.0804405, 3.6011294, 3.4396213...
## $ ln_roaddensity     <dbl> 4.5529100, 3.4863607, 2.9766632, 2.7561000...
## $ ln_fdi_x           <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ fdi_stock_x        <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ ln_grppc_x         <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ ln_trade_x         <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ ln_roaddensity_x   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ ln_pop_x           <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
## $ empl_high_x        <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...
```

```
## $ model1_fitted       <dbl> 6.6284552, 3.5112219, 4.9617720, 6.8221503...
## $ model1_residual     <dbl> -0.1569505, -2.5792824, 1.9837484, 0.84983...
## $ model2_fitted       <dbl> 3.1584778, 2.4576677, 2.6382538, 3.3257665...
## $ model2_residual     <dbl> -0.116857463, -1.377227237, 0.962875647, 0...
## $ ln_corruption       <dbl> 0.9028677, 1.0296194, 0.9555114, 1.0986123...
## $ model3_fitted       <dbl> 6.8126765, 3.4231844, 5.6313084, 7.3739384...
## $ model3_residual     <dbl> -0.34117189, -2.49124482, 1.31421191, 0.29...
## $ fdi_dummy           <fctr> 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0,...
## $ model1_fitted_logit <dbl> 0.4401839459, 0.0323009716, 0.0842924849, ...
## $ model1_probs_logit  <dbl> 0.6083029, 0.5080745, 0.5210607, 0.6615124...
```

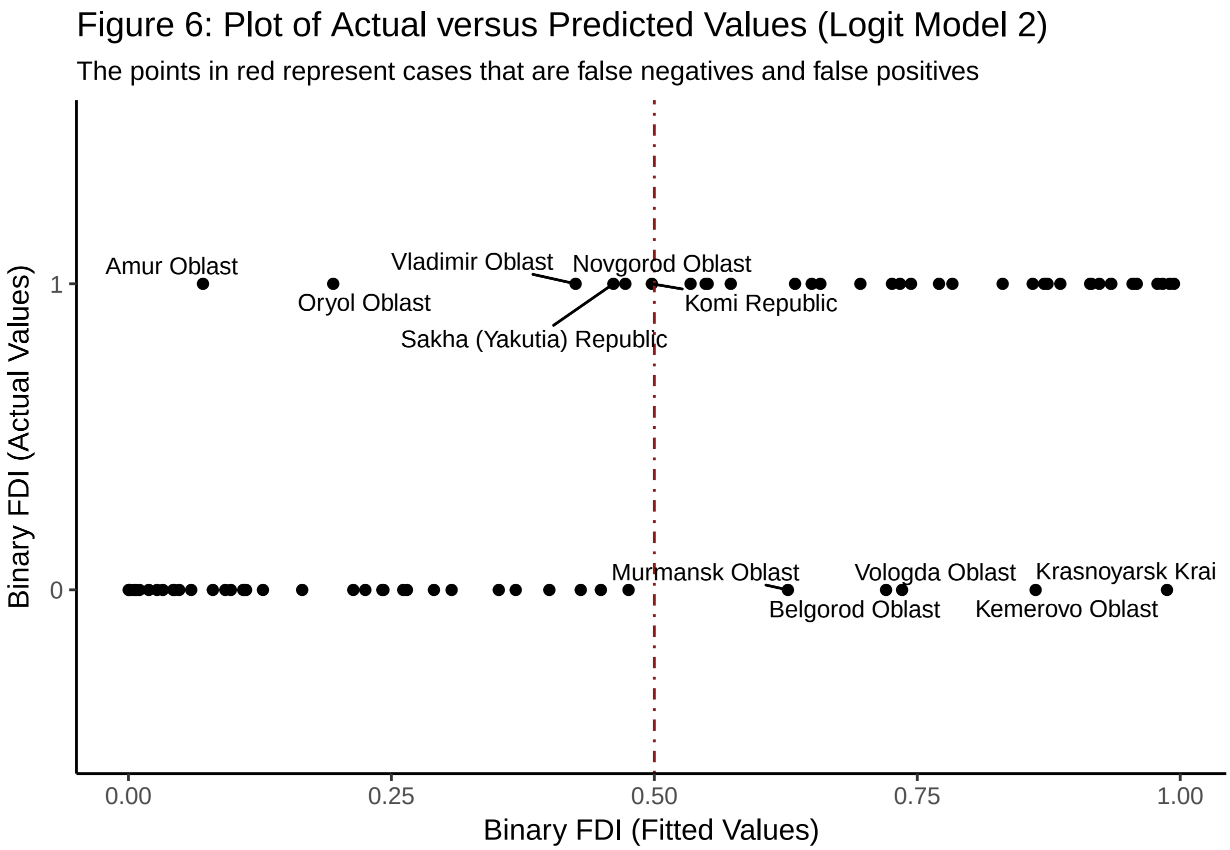
Based on the Figure 5 above, we can see that the off-the-line cases are Amur Oblast, Oryol Oblast, Novgorod Oblast, Vladimir Oblast, Kaluga Oblast, Republic of Karelia, Murmansk Oblast, Vologda Oblast, Kemerovo Oblast, Belgorod Oblast, and Krasnoyarsk Krai. Off-the-line cases for a logistic regression have a somewhat less direct meaning than that for the earlier model. For instance, in this case I selected off the line cases based on whether they were "false positives" or "false negatives". False positives were seleced on the basis if their predicted probability was higher than 0.5 when their actual value was in fact 0. False negatives implies cases that had a predicted probably of less than 0.5 when in fact their actual value was 1 based on the dummy that we had specified earlier.

```r
## Now let's redefine our previous model by including additional variables to see if it improves the ori

# Logistic regression

formula_logit_2 <- fdi_dummy ~ ln_pop + ln_trade + ln_roaddensity + empl_high + ln_corruption + russhare

model_2_logit <- glm(formula_logit_2, family = binomial,
               data = russia_region)

screenreg(list(model_1_logit, model_2_logit))
```

```
##
## =====================================
##                  Model 1     Model 2
## -------------------------------------
## (Intercept)     -14.66 **   -20.60 **
##                  (4.94)      (6.86)
## ln_pop            0.37        0.62
##                  (0.71)      (0.79)
## ln_trade          1.47 ***    1.32 **
##                  (0.44)      (0.47)
## ln_roaddensity   -0.02       -0.16
##                  (0.31)      (0.66)
## empl_high         0.12        0.18
##                  (0.11)      (0.12)
## ln_corruption                 3.61
##                              (2.57)
## russhare                     -0.00
##                              (0.03)
## old                           0.05
##                              (0.21)
## -------------------------------------
## AIC              74.41       78.14
## BIC              86.19       96.89
## Log Likelihood  -32.20      -31.07
```

```
## Deviance          64.41       62.14
## Num. obs.         78          77
## =====================================
## *** p < 0.001, ** p < 0.01, * p < 0.05
```

```r
## Let's calculated the predicted values for the logit

russia_region$model2_fitted_logit <- predict(model_2_logit, russia_region, type = "response")

#russia_region$model1_residuals_logit <- residuals(model_1_logit)

russia_region$model2_probs_logit <- model_2_logit$family$linkinv(russia_region$model2_fitted_logit)

## let's plot the 2 x 2

figure_6 <- ggplot(russia_region, aes(model2_fitted_logit, fdi_dummy)) +
  geom_point() +
  geom_vline(xintercept = 0.5, linetype = "dotdash", color = "firebrick4") +
  labs(y = "Binary FDI (Actual Values)",
       x = "Binary FDI (Fitted Values)",
       title = "Figure 6: Plot of Actual versus Predicted Values (Logit Model 2)",
       subtitle = "The points in red represent cases that are false negatives and false positives") +
  theme_classic() +
  geom_text_repel(aes(label = ifelse(model2_fitted_logit >= 0.5 & fdi_dummy == 0,
                                     as.character(name),'')),
            size = 3) +
  geom_text_repel(aes(label = ifelse(model2_fitted_logit <= 0.5 & fdi_dummy == 1,
                                     as.character(name),'')),
            size = 3); figure_6
```




```
  #geom_point(data = russia_region[russia_region$name %in% c("Amur Oblast", "Oryol Oblast","Novgorod Ob
```

Did the model improve?

Figure 6 in the code chunk above makes it difficult to comment on whether the model improved with the inclusion of new variables. Indeed, what we observe between Figures 5 and 6 is that while some false (positives and negatives both) cases disappeared, they were replaced by newer falsely identified regions. As such the visual evidence is inconclusive. In this case, the hitmiss command used in the code chunk below is useful in commenting on whether the model improved or not. Based on the command the inlclusion of additional variables did not significant alter the percentage of correctly predicting a logistic regression between logit Models 1 and 2 that have a correctly predicted percentage of 85.9 % and 85.7 %. Indeed, if anything there is some intuition to suspect that the model slightly worsened with the inclusion of additional explanatory variables.

```
hitmiss(model_1_logit, digits = 4, k = 0.5)
```

```
## Classification Threshold = 0.5
##        y=0 y=1
## yhat=0  36   6
## yhat=1   5  31
## Percent Correctly Predicted = 85.9%
## Percent Correctly Predicted = 87.8%, for y = 0
## Percent Correctly Predicted = 83.78%  for y = 1
## Null Model Correctly Predicts 52.56%

## [1] 85.89744 87.80488 83.78378
```

```r
hitmiss(model_2_logit, digits = 4, k = 0.5)
```

```
## Classification Threshold = 0.5
##        y=0 y=1
## yhat=0  35   6
## yhat=1   5  31
## Percent Correctly Predicted = 85.71%
## Percent Correctly Predicted = 87.5%, for y = 0
## Percent Correctly Predicted = 83.78%  for y = 1
## Null Model Correctly Predicts 51.95%

## [1] 85.71429 87.50000 83.78378
```