Applied Econometrics Homework
M2 FE

Khalil Janbek, Romain Jouhameau

01/01/2022

```r
library(data.table)
library(dtplyr)
library(dplyr, warn.conflicts = FALSE)
library(readxl)
library(janitor)
library(plm)
library(modelsummary)
library(kableExtra)
library(ggrepel) # for spacing text inside plots
library(tidyverse)
library(sf)
library(usmap)
library(tsibble)
library(DBI)
library(DataExplorer)
library(patchwork)
```

```r
con <- dbConnect(RSQLite::SQLite(), "Data/DB.sqlite")
DB = tbl(con, "DB")

DB %>%
  head() %>%
  collect()
```

```
#> # A tibble: 6 x 43
#>   Loan_Seq_Number Reporting_Period Current_UPB Delinquency_Status Loan_Age
#>   <chr>           <chr>                  <dbl> <chr>                 <int>
#> 1 F20Q10000703    2020-02-01            342000 0                         0
#> 2 F20Q10000703    2020-03-01            342000 0                         1
#> 3 F20Q10000703    2020-04-01            341000 0                         2
#> 4 F20Q10000703    2020-05-01            341000 0                         3
#> 5 F20Q10000703    2020-06-01            340000 0                         4
#> 6 F20Q10000703    2020-07-01            340000 0                         5
#> # ... with 38 more variables: Time_to_Maturity <int>, Zero_Balance_Code <int>,
#> #   Current_Interest_Rate <dbl>, E_LoanToValue <int>,
#> #   Delinquency_Due_To_Disaster <chr>, Borrower_Assistance_Status_Code <chr>,
#> #   Interest_Bearing_UPB <dbl>, Credit_score <int>, First_Payment_Date <int>,
#> #   FstTime_HB_Flag <chr>, Maturity_Date <int>, MSA <int>,
#> #   Mortgage_Insurance_pct <int>, Number_of_Units <int>,
#> #   Occupancy_Status <chr>, O_CombinedLoanToValue <int>, ...
```

PART A:

Option 1 - Master thesis dataset Khalil's Master thesis explores the topic of the performance of securitized mortgage loans in the US in the context of the coronavirus crisis. Based on issuance and performance data for more than 346,724 mortgage loans purchased by Freddie Mac in 2020Q1, combined with neighborhood-level coronavirus data, the thesis will examine, at the US local level, whether the coronavirus outbreak affected these loans' (i) delinquency status and (ii) prepayment. In parallel, the Master thesis also enquires whether the status of the lender that originated the loan - commercial bank or shadow bank, Fintech or not - has an influence on those two loan performance metrics.

1. In your data set, which are the variables which are varying with respect to two indices (or more if you consider inflows and outflows from one individual or country to another individual or countries? Which are the variables which are varying only with respect to time? Which are the variables which are varying only with respect to individuals? For each loan, the database records:

- (i) time-invariant issuance metrics. Those variables vary only with respect to individuals (in other words by loan) and do not vary over. Those time-invariant variables regroup for instance borrower credit score at issuance, the original loan to value, XXX and the lender type.
- (ii) time-varying performance metrics. Those variables vary both with respect to time and with respect to individuals. For instance, prepayment (Zero Balance Code), Delinquency status, and unpaid principal.

2. What is the largest number of period T for individuals? What is the number of individuals? The dataset counts 17 time periods, spanning from a month in 2020Q1 to June 2021. The sample consists of 346,724 loans, originated at various dates but purchased by Freddie Mac in 2020Q1.

```
DB %>%
  count(Loan_Seq_Number) %>%
  summarise(Number_Loans = n(),
            Max_T = max(n, na.rm = TRUE),
            Min_T = min(n, na.rm = TRUE),
            Mean_T = mean(n, na.rm = TRUE),
            Median_T = median(n, na.rm = TRUE),
            ) %>%
  collect()
```

```
#> # A tibble: 1 x 5
#>   Number_Loans Max_T Min_T Mean_T Median_T
#>          <int> <int> <int>  <dbl>    <int>
#> 1       346724    17     1   14.0       15
```

Lenders originating and selling these mortgage loans are classified as either commercial banks or shadow banks. The latter category also includes Fintech lenders.

```
DB %>%
  mutate(Type = case_when(
          Lender_Type == 1 ~ 'Bank',
          Lender_Type == 0 & Fintech == 0 ~ 'Shadow Bank',
          Lender_Type == 0 & Fintech == 1 ~ 'Fintech'
        )) %>%
  group_by(Type) %>%
  count(Loan_Seq_Number) %>%
  summarise(Number_Loans = n(),
                 Max_T = max(n, na.rm = TRUE),
                 Min_T = min(n, na.rm = TRUE),
                 Mean_T = mean(n, na.rm = TRUE),
                 Median_T = median(n, na.rm = TRUE)) %>%
  collect()
```

```
#> # A tibble: 3 x 6
#>   Type        Number_Loans Max_T Min_T Mean_T Median_T
#>   <chr>              <int> <int> <int>  <dbl>    <int>
#> 1 Bank              147506    17     1   14.3       15
#> 2 Fintech            52273    17     1   13.7       15
#> 3 Shadow Bank       146945    17     1   13.9       15
```

3. Comment on the structure of the unbalanced panel (how many (and which) countries have a single observation, discontinuities between observations, how many have at least 2 consecutive observations (which is useful to compute lags, autocorrelations, first difference and within estimators)? As shown in the previous question, our dataset has an unbalanced panel data structure, with time periods per individual loan varying from 1 to 17.

```
DB %>%
  group_by(Loan_Seq_Number) %>%
  tally() %>%
  count(n) %>%
  rename(Number_Periods_available = n,
         Number_of_loans = nn) %>%
  mutate(cum = cumsum(Number_of_loans),
         percent = 100 * (cum / 346724))
```

```
#> # Source:   lazy query [?? x 4]
#> # Database: sqlite 3.37.0 [/Users/romainjouhameau/Documents/M2
#> #   FE/Applied_Econometrics/Data/DB.sqlite]
#>    Number_Periods_available Number_of_loans   cum percent
#>                       <int>           <int> <int>   <dbl>
#>  1                        1             123   123  0.0355
#>  2                        2             467   590  0.170
#>  3                        3             963  1553  0.448
#>  4                        4            1655  3208  0.925
#>  5                        5            3058  6266  1.81
#>  6                        6            6189 12455  3.59
#>  7                        7           10496 22951  6.62
#>  8                        8           11092 34043  9.82
#>  9                        9           11599 45642 13.2
#> 10                       10           11310 56952 16.4
#> # ... with more rows
```

```
DB_temp <- DB %>%
  # Reformatting the "Zero Balance Code" variable: 1 if prepaid, 0 otherwise
  mutate(Zero_Balance_Code = ifelse(Zero_Balance_Code == 1, 1, 0),
         Zero_Balance_Code = ifelse(is.na(Zero_Balance_Code) == T, 0, Zero_Balance_Code)) %>%
  # Reformatting the "Delinquency status" variable: counting "REO acquisition as "NA"
  mutate(Delinquency_Status = ifelse(Delinquency_Status == 'RA', NA, Delinquency_Status)) %>%
  # Reformatting the "Estimated Loan-to-Value" variable: counting "999" as "NA"
  mutate(E_LoanToValue = ifelse(E_LoanToValue == 999, NA, E_LoanToValue)) %>%
  # Reformatting the "Original Debt-to-Income" variable: counting "999" as "NA"
  mutate(O_DebtToIncome = ifelse(O_DebtToIncome == 999, NA, O_DebtToIncome))
```

```
DB_KJ <- DB_temp %>%
  select(Loan_Seq_Number, Reporting_Period, Delinquency_Status, Zero_Balance_Code,
         confirmed, Credit_score, Current_Interest_Rate,Current_UPB, E_LoanToValue,
         O_DebtToIncome, O_LoanToValue, Time_to_Maturity, Lender_Type, Fintech, MSA, Seller_Name) %>%
  mutate(Delinquency_Status = as.integer(Delinquency_Status)) %>%
  collect() %>%
  mutate_if(is.integer, as.double)
```

```
Loans_by_MSA <-
  DB_temp %>%
  group_by(MSA) %>%
  summarise(LoansPerMSA = n_distinct(Loan_Seq_Number)) %>%
  collect() %>%
  arrange(desc(LoansPerMSA))
```

```r
Loans_by_MSA

# Identifying which loans are not related to an MSA (22,351 / 346,724)
#sum(Loans_by_MSA$LoansPerMSA)
```

```
#> # A tibble: 405 x 2
#>      MSA LoansPerMSA
#>    <int>       <int>
#>  1    NA       22351
#>  2 38060       11947
#>  3 31084       11555
#>  4 16984       10082
#>  5 12060        8319
#>  6 40140        7655
#>  7 19740        7428
#>  8 47894        7176
#>  9 19124        6679
#> 10 26420        6170
#> # ... with 395 more rows
```

```
full_dates <- DB %>%
  group_by(Loan_Seq_Number) %>%
  tally() %>%
  filter(n == 17) %>%
  pull(Loan_Seq_Number)

DB_test <- DB_KJ %>%
  filter(Loan_Seq_Number %in% full_dates)
```

Starting by exploring missing observations, the graph shows that 6.75% of monthly loan performance observations relate to mortgage loans originated in locations that are not in an MSA - amounting to 22,351 loans, or 6.4% of the total of mortgages in our database. Those loans are likely to have been originated in sparsely populated areas.

Monthly developments in COVID-19 confirmed cases by MSA are also missing in 8.45% of our monthly loan performance observations. This percentage includes the 6.75% of loan performance observations that are not located in an MSA. As our loan performance data start in February 2020, the remaining missing COVID-19 observations are related to monthly loan performance data that date prior to March 2020 - date at which COVID-19 cases started to be tracked.

As briefly shown in question 2, we have between 1 and 17 consecutive observation for each loan contract. Nonetheless, the number of loans for which we have only 1 period of observation is very small compared to the total number of loans of the dataset - 123 out 346,724, or 0.3%. All the other loans therefore have at least 2 consecutive observations.

None of our loan performance data display discontinuities between observations.

4. Compute between transformed and within transformed variables for all variables. Present a table with the within, between and pooled variance for each variable. Compute the share of between and within variance in the total variance for each variable. Comment these results.

- Pooled deviation = $x_{i,t} - \overline{x}$
- Within deviation = $x_{i,t} - \overline{x}_i$
- Between deviation = $\overline{x}_i - \overline{x}$

```
pooled <- DB_test %>%
  mutate_at(vars(Delinquency_Status:Fintech), ~ . -  mean(., na.rm = T)) %>%
  summarise(across(c(Delinquency_Status:Fintech), ~var(., na.rm = T))) %>%
  pivot_longer(c(Delinquency_Status:Fintech), names_to = 'variable', values_to = 'Pooled')
```

```
# Within transformation
within_transformation <- DB_test %>%
  group_by(Loan_Seq_Number) %>%
  mutate_at(vars(Delinquency_Status:Fintech), ~ . -  mean(., na.rm = T)) %>%
  ungroup() %>%
  summarise(across(c(Delinquency_Status:Fintech), ~var(., na.rm = T))) %>%
  pivot_longer(c(Delinquency_Status:Fintech), names_to = 'variable', values_to = 'Within')
```

```
between_transformation <- DB_test %>%
  group_by(Loan_Seq_Number) %>%
  mutate_at(vars(Delinquency_Status:Fintech), ~  mean(., na.rm = T)) %>%
  ungroup() %>%
  summarise(across(c(Delinquency_Status:Fintech), ~var(., na.rm = T))) %>%
  pivot_longer(c(Delinquency_Status:Fintech), names_to = 'variable', values_to = 'Between')
```

```
pooled %>%
  left_join(within_transformation, by = 'variable') %>%
  left_join(between_transformation, by = 'variable') %>%
  kable()
```

variable	Pooled	Within	Between
Delinquency_Status	1.291464e+00	5.487819e-01	7.427458e-01
Zero_Balance_Code	1.997200e-03	1.883500e-03	1.137000e-04
confirmed	1.193854e+14	5.874923e+13	6.063599e+13
Credit_score	1.979622e+03	0.000000e+00	1.979622e+03
Current_Interest_Rate	2.162531e-01	7.420000e-05	2.161789e-01
Current_UPB	1.443434e+10	2.284779e+08	1.420587e+10
E_LoanToValue	2.999428e+02	4.042771e+01	2.633113e+02
O_DebtToIncome	9.013702e+01	0.000000e+00	9.013702e+01
O_LoanToValue	2.929620e+02	0.000000e+00	2.929620e+02
Time_to_Maturity	5.226496e+03	2.656281e+01	5.199934e+03
Lender_Type	2.500002e-01	0.000000e+00	2.500002e-01
Fintech	1.153918e-01	0.000000e+00	1.153918e-01

5. Plot the distribution of the within and between transformed dependent variable and of you key (preferred) explanatory variable (not all the explanatory variable) [in Burnside and Dollar: GDP growth and foreign aid EDA/GDP], using on the same graph an histogram, a normal law with same empirical mean and standard error and a kernel continuous approximation. Comment the between and within difference for each variable, and compare within/within for dependent and explanatory variable, and between/between for dependent and explanatory variable: kurtosis, skewness, non-normality, high leverage observation (far from the mean), several modes (mixture of distribution)? Definitions of between and within transformation: XXX

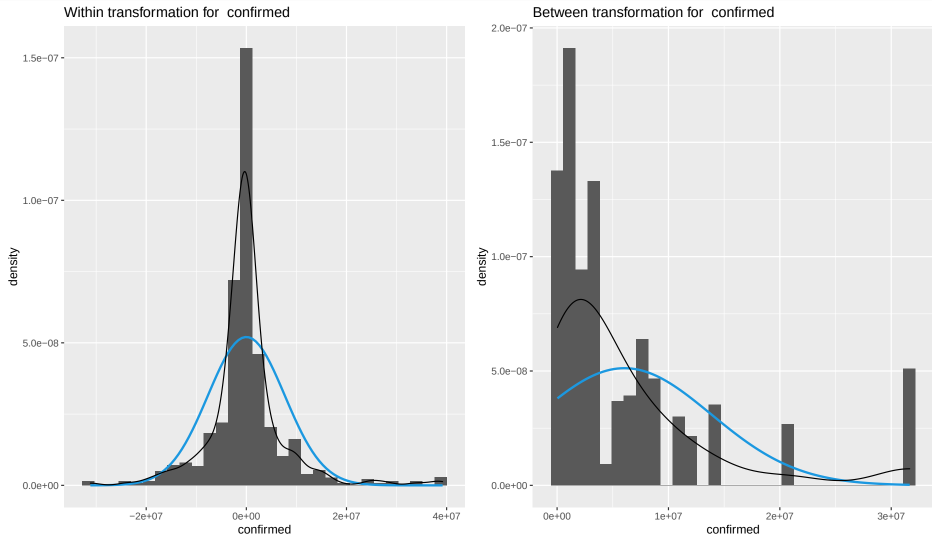
We carry out the calculation of within and between transformations for all variables except the binary variables - namely the lender type - because XXX.

```
within_transformation_graph <- DB_test %>%
  group_by(Loan_Seq_Number) %>%
  mutate_at(vars(Delinquency_Status:Fintech), ~ . -  mean(., na.rm = T)) %>%
  ungroup()

between_transformation_graph <- DB_test %>%
  group_by(Loan_Seq_Number) %>%
  mutate_at(vars(Delinquency_Status:Fintech), ~  mean(., na.rm = T)) %>%
  ungroup()
```

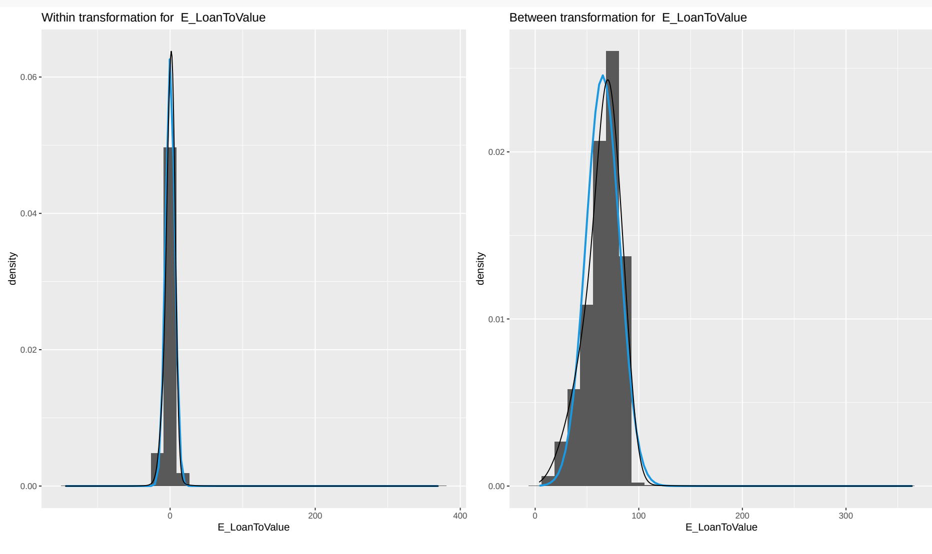
```
within_between_plot <- function(col, adjust = 1) {
  # col is the name of the variable we're interested in
  # adjust is used to smooth the kernel curve (the black one)

  within_data <- within_transformation_graph %>%
  select( {{ col }} ) %>%
  drop_na()

  between_data <- between_transformation_graph %>%
  select( {{ col }} ) %>%
  drop_na()

  # Get the name of the variable for the title plot
  col_name <- within_data %>% names()


within_plot <- ggplot(within_data, aes( {{ col }} )) +
  # To compute the histogram
  geom_histogram(aes(y = ..density..)) +
  # To compute the normal curve
  stat_function(fun = dnorm,
                args = list(mean = mean(within_data %>% pull()), # there's only 1 column
                            sd = sd(within_data %>% pull())), # pull is like within_data$col
                col = "#1b98e0",
                size = 1) +
  # To compute the kernel distribution
  geom_density(adjust = adjust)  +
  labs(title = paste('Within transformation for ', col_name))

between_plot <- ggplot(between_data, aes( {{ col }} )) +
  geom_histogram(aes(y = ..density..)) +
  stat_function(fun = dnorm,
                args = list(mean = mean(between_data %>% pull()),
                            sd = sd(between_data %>% pull())),
```

```
                col = "#1b98e0",
                size = 1) +
  geom_density(adjust = adjust)  +
  labs(title = paste('Between transformation for ', col_name))

within_plot + between_plot
}
```

```
within_between_plot(confirmed, adjust = 10)
```




```
within_between_plot(E_LoanToValue, adjust = 10)
```



6. Plot boxplot of within distribution and between distribution for the dependent variable and the key explanatory variables. Comment that you find the same insights from question 5.

7. Compute univariate descriptive statistics (min, Q1, median, Q3, max, mean, standard error) for Within and Between transformed variables. Is the mean different from the median and why? How many standard errors from the mean are the min and max extremes (report (MAX-average)/standard error and (MIN-average)/standard error in the tables)?

8. Plot the boxplot of within transformed dependent variable and the key explanatory variable by a few individual (all of them if N around 50) and only the first 20 of them for larger data set. Comment on their differences of standard errors and means for each individuals

9. Compare and comment the within and between transformed bivariate correlation matrix for all variables (include a time trend 1,2,.,T). Check poor simple correlation with the dependent variables and high correlation between explanatory variables.

7. Compute univariate descriptive statistics (min, Q1, median, Q3, max, mean, standard error) for Within and Between transformed variables. Is the mean different from the median and why? How many standard errors from the mean are the min and max extremes (report (MAX-average)/standard error and (MIN-average)/standard error in the tables)?

8. Plot the boxplot of within transformed dependent variable and the key explanatory variable by a few individual (all of them if N around 50) and only the first 20 of them for larger data set. Comment on their differences of standard errors and means for each individuals

9. Compare and comment the within and between transformed bivariate correlation matrix for all variables (include a time trend 1,2,.,T). Check poor simple correlation with the dependent variables and high correlation between explanatory variables.

10. Comment the bivariate auto-correlation and trend-correlations (check the number of observations).

11. Comment the bivariate graphs with linear, quadratic and Lowess fit for dependent and key explanatory variable (aid/gdp and growth of gdp): Within transformed, Between transformed.

12. Comment the results of estimations of Between, Within (fixed effects, (fe)) and Mundlak (random effects (re) including all X(i.) as regressors), two-way fixed effects (add year dummies in fe regression) and First differences, including all explanatory variables except the ones with high near-multicollinearity in their respective between or within space.

13. If one of your variable is time-invariant z(i) (Institutional quality ICRG for Burnside Dollar), run a baseline Hausman Taylor estimation including all X(i.) as instruments. Comment the results.

14. If one of your variable is time-invariant z(i) (Institutional quality ICRG for Burnside Dollar), run a between regression on z(i) explained by X(i.) and other time invariant variable (only with N observations). If the R2 is low, this may signal X(i.) are weak instruments poorly correlated with the variable z(i) to be instrumented. Comment.

15. Optional: mention or propose improvements to the Python, STATA, SAS or R code (copy it here). Optional: propose improvements, additional insights, and you do not know how to code them.

PART B (update results)

1. Download 5 panel data variables from World Bank and/or IMF and/or FRED databases for the recent period (1990-2020) and for the largest coverage of emerging economies: GDP/head, GDP/head PPP-adjusted (very last update), Log(population), Foreign aid/GDP (ODA), of log an index of corruption (or good public sector governance) from the World Bank. From now on, consider as your sample only country-year observations which are available for ALL the 5 variables for at least TWO CONSECUTIVE years for a given country. The full class may coordinate for this updated database. In all the following questions except perhaps the last one, the PPP adjusted GDP is not used. So we consider 4 variables excluding GDP/head PPP adjusted. We download yearly data for 88 countries over 23 periods (between 1996 and 2019) for the following variables: (i) the World Bank corruption index, (ii) GDP per capita in euros, (iii) PPP-adjusted GDP per capita (iv) foreign aid - as % of Gross National Income and per capita - and (v) population.

```
# Import Data
# Inside the Data folder, get all the .RDS files except MSA_Large
panel_data <- list.files(path =  'Data/', pattern="*[^(MSA_Large)].RDS") %>%
  map(., ~read_rds(paste0('Data/', .))) %>%
  reduce(inner_join, by = c('iso2c', 'country', 'year'))

panel_data
```

```
#> # A tibble: 1,848 x 9
#>    iso2c country corruption  year gdp_ppp gdp_per_cap oda_gni oda_net population
#>    <chr> <chr>        <dbl> <int>   <dbl>       <dbl>   <dbl>   <dbl>      <dbl>
#>  1 AL    Albania     -0.533  2019  13657.       4549.   0.187    9.95    2854191
#>  2 AL    Albania     -0.479  2018  13317.       4434.   2.27   120.      2866376
#>  3 AL    Albania     -0.421  2017  12771.       4250.   1.29    58.6     2873457
#>  4 AL    Albania     -0.405  2016  12292.       4090.   1.42    59.5     2876101
#>  5 AL    Albania     -0.479  2015  11878.       3953.   2.91   116.      2880703
#>  6 AL    Albania     -0.548  2014  11587.       3856.   2.11    97.3     2889104
#>  7 AL    Albania     -0.698  2013  11361.       3781.   2.08    93.3     2895092
#>  8 AL    Albania     -0.726  2012  11228.       3736.   2.86   120.      2900401
#>  9 AL    Albania     -0.683  2011  11053.       3678.   2.94   131.      2905195
#> 10 AL    Albania     -0.525  2010  10749.       3577.   3.09   125.      2913021
#> # ... with 1,838 more rows
```

Based on the below table, as we lack observations for the years 1997, 1999 and 2001, we restrict our analysis to the 2002-2019 period for those 88 countries.

```
panel_data %>%
  count(year)
```

```
#> # A tibble: 21 x 2
#>     year     n
#>    <int> <int>
#>  1  1996    88
#>  2  1998    88
#>  3  2000    88
#>  4  2002    88
#>  5  2003    88
#>  6  2004    88
#>  7  2005    88
#>  8  2006    88
#>  9  2007    88
#> 10  2008    88
#> # ... with 11 more rows
```

We therefore remove the years 1996, 1998 and 2000 from our sample.

```r
panel_data <- panel_data %>%
  filter(!year %in% c(1996, 1998, 2000))

panel_data %>%
  count(country)
```

```
#> # A tibble: 88 x 2
#>    country                 n
#>    <chr>               <int>
#>  1 Albania                18
#>  2 Algeria                18
#>  3 Antigua and Barbuda    18
#>  4 Argentina              18
#>  5 Bangladesh             18
#>  6 Belize                 18
#>  7 Benin                  18
#>  8 Bhutan                 18
#>  9 Bolivia                18
#> 10 Botswana               18
#> # ... with 78 more rows
```

As we are concerned with the impact of foreign aid on economic growth, we also remove from our sample countries that have a negative net ODA.

```r
# We remove them because they have negative ODA for at least one year
# We would need to find Gross ODA in order to have only positives values
country_to_remove <- panel_data %>%
  group_by(country) %>%
  filter(oda_net < 0) %>%
  distinct(country) %>%
  pull()

country_to_remove
```

```
#>  [1] "Argentina"   "China"       "Gabon"       "Indonesia"   "Malaysia"
#>  [6] "Mauritius"   "Panama"      "Peru"        "Philippines" "Sri Lanka"
#> [11] "Thailand"
```

We therefore arrive to the following final dataset, which comprises 77 countries over 18 years (2002-2019).

```r
panel_data <- panel_data %>%
  filter(!country %in% country_to_remove)

panel_data %>%
  count(country)
```

```
#> # A tibble: 77 x 2
#>    country                 n
#>    <chr>               <int>
#>  1 Albania                18
#>  2 Algeria                18
#>  3 Antigua and Barbuda    18
#>  4 Bangladesh             18
#>  5 Belize                 18
#>  6 Benin                  18
#>  7 Bhutan                 18
#>  8 Bolivia                18
#>  9 Botswana               18
```

```
#> 10 Brazil                 18
#> # ... with 67 more rows
```

```
panel_data <- panel_data %>%
  arrange(country, year) %>%
  group_by(country) %>%
  mutate(g_gdp_per_cap = log(gdp_per_cap) - dplyr::lag(log(gdp_per_cap)),
         g_population = log(population) - dplyr::lag(log(population)),
         g_oda_net = log(oda_net) - dplyr::lag(log(oda_net))) %>%
  ungroup()
```

2. Compute 2 growth rates using the difference of log: the growth of GDP/head (difference of log, denoted GDPg), the growth of foreign aid ODAg (but NOT the growth for foreign aid/GDP: remove the difference of log of GDP from the difference of log of foreign aid/GDP).

3. Compute the between average over time for the first period and for the second period for the 6 variables. Provide the top 10 of countries for ODA/GDP with average over time for each period. We separate our dataset in two periods of approximately equal size in terms of available observations: 2002-2013 for Period 1 and 2014-2019 for Period 2.

As in Jia and Williamson (2018), we compare the countries that are the top recipients of foreign aid compared to GDP in those two subsets of the database.

```
Period_1 <- subset(panel_data, year == 2002:2013)

Period_1 %>%
  group_by(country) %>%
  mutate(oda_net_gdp_cap = oda_net / gdp_per_cap) %>%
  # Summarise is used  to transform our dataframe and calculate the mean for each country
  summarise(across(where(is.double), ~mean(., na.rm = T))) %>% # across() applies a function (here the mea
  arrange(desc(oda_net_gdp_cap)) %>%
  relocate(country, oda_net_gdp_cap) %>%
  slice_max(oda_net_gdp_cap, n = 10)
```

```
#> # A tibble: 10 x 11
#>    country        oda_net_gdp_cap corruption gdp_ppp gdp_per_cap oda_gni oda_net
#>    <chr>                    <dbl>      <dbl>   <dbl>       <dbl>   <dbl>   <dbl>
#>  1 Malawi                  0.171    -0.589     1234.        322.   15.8     55.7
#>  2 Rwanda                  0.147    -0.00580   1339.        533.   17.4     79.8
#>  3 Cabo Verde              0.146     0.757     5610.       2724.   14.9    404.
#>  4 Sierra Leone            0.134    -0.919     1381.        526.   19.3     69.1
#>  5 Burkina Faso            0.107    -0.272     1617.        549.   11.0     59.2
#>  6 Niger                   0.0904   -0.746     1002.        429.    9.62    38.8
#>  7 Ethiopia                0.0840   -0.652     1076.        387.   13.2     32.4
#>  8 Central Afric~          0.0836   -1.11      1091.        483.    9.59    40.7
#>  9 Lesotho                 0.0812    0.0429    2181.        925.    6.51    78.9
#> 10 Tanzania                0.0799   -0.561     1864.        691.    9.46    55.5
#> # ... with 4 more variables: population <dbl>, g_gdp_per_cap <dbl>,
#> #   g_population <dbl>, g_oda_net <dbl>
```

```
Period_2 <- subset(panel_data, year == 2014:2019)

Period_2  %>%
  group_by(country) %>%
  mutate(oda_net_gdp_cap = oda_net / gdp_per_cap) %>%
  # Summarise is used  to transform our dataframe and calculate the mean for each country
  summarise(across(where(is.double), ~mean(., na.rm = T))) %>% # across() applies a function (here the mea
  arrange(desc(oda_net_gdp_cap)) %>%
  relocate(country, oda_net_gdp_cap) %>%
  slice_max(oda_net_gdp_cap, n = 10)
```

```
#> # A tibble: 10 x 11
#>    country        oda_net_gdp_cap corruption gdp_ppp gdp_per_cap oda_gni oda_net
#>    <chr>                    <dbl>      <dbl>   <dbl>       <dbl>   <dbl>   <dbl>
#>  1 Central Afric~           0.324    -1.23      892.        395.   28.4    128.
#>  2 Malawi                   0.178    -0.743    1479.        386.   16.3     68.9
#>  3 Vanuatu                  0.172    -0.0283   3031.       2799.   16.0    480.
#>  4 Burundi                  0.164    -1.32      799.        296.   18.9     48.5
#>  5 Sierra Leone             0.148    -0.675    1688.        643.   17.6     95.3
#>  6 Solomon Islan~           0.138    -0.144    2615.       2250.   13.8    311.
#>  7 Rwanda                   0.122     0.636    1985.        789.   12.9     96.0
#>  8 Guinea-Bissau            0.117    -1.52     1862.        624.   10.4     72.8
#>  9 Gambia, The              0.114    -0.610    2098.        673.   11.3     76.8
```

```
#> 10 Niger                    0.104    -0.632    1166.        499.    9.35    51.8
#> # ... with 4 more variables: population <dbl>, g_gdp_per_cap <dbl>,
#> #   g_population <dbl>, g_oda_net <dbl>
```

We notice that the 10 countries with the highest ODA/GDP average over the period differ between those two samples. While Malawi, Rwanda, Sierra Leone, Niger and the Central African Republic are top recipients in foreign aid / GDP on average for both periods 1 and 2, Burkina Faso, Cabo Verde and Tanzania have a lower relative to period 1 and are no more among the top 10 recipients in period 2.

Average ODA/GDP are higher in period 2 - ranging from 10.4% to 32% compared to 7.9% and 17% in period 1 - explained by the presence of other countries compared to period 1, namely the Gambia, Guinea-Bissau, the Solomon Islands, Burundi and Vanuatu.

The magnitude of the change in the average ODA/GDP ratio for the Central African republic is the most striking, from 17% in period 1 to 32% on average over the period 2.

4. Compute the proportion of country-years observations in your database such that 0<=ODA/GDP<0.5% Now turning to the countries with the lowest ODA/GDP ratios, 23 countries of our dataset have ratios between 0% and 0.5%. Notably, Algeria, Brazil, Costa Rica, Dominican Republic, India, Mexico and Turkey have a ratio lower than 0.5% for all the 18 years of observations.

Albania, Côte d'Ivoire, Iraq and Vietnam, while having among the lowest ODA/GDP ratios, have an ODA/GDP lower than 0.5% for only 1 year of observation.

```
panel_data %>%
  group_by(country) %>%
  mutate(oda_net_gdp_cap = oda_net / gdp_per_cap) %>%
  filter(oda_net_gdp_cap >= 0,
         oda_net_gdp_cap < 0.005) %>%
  count(country) %>%
  mutate(prop = round(n / 18, 2)) %>%
  arrange(desc(n))
```

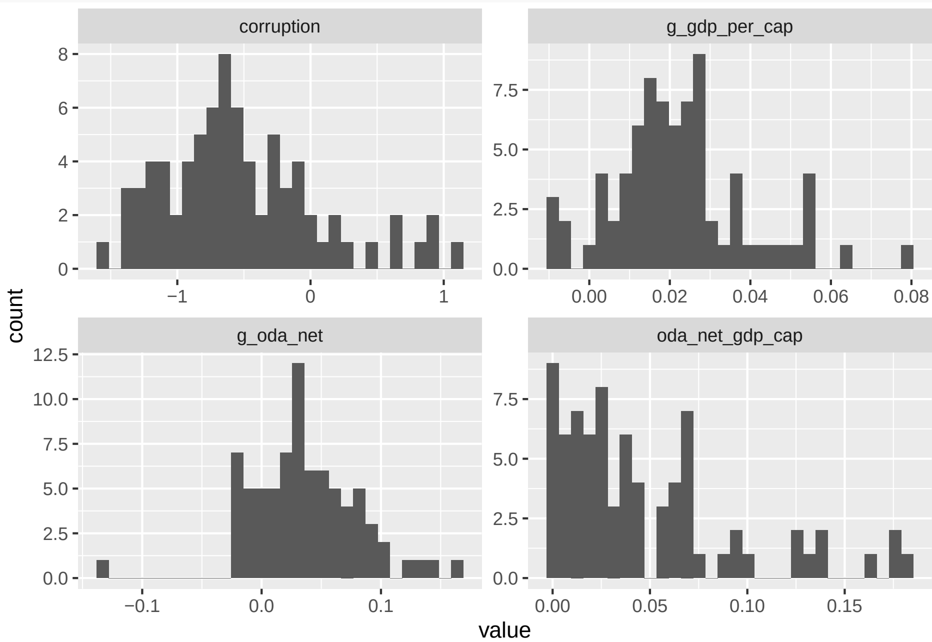
```
#> # A tibble: 23 x 3
#> # Groups:   country [23]
#>    country                n  prop
#>    <chr>              <int> <dbl>
#>  1 Algeria               18  1
#>  2 Brazil                18  1
#>  3 Costa Rica            18  1
#>  4 Dominican Republic    18  1
#>  5 India                 18  1
#>  6 Mexico                18  1
#>  7 Turkey                18  1
#>  8 Ecuador               17  0.94
#>  9 Colombia              16  0.89
#> 10 Equatorial Guinea     16  0.89
#> # ... with 13 more rows
```

5. Compute the between and within transformations of the 6 variables over the full period. Provide the 4 histograms for ODA/GPD, growth of ODA, growth of GDP/head, corruption index for both between and within transformed variables (hence 8 histograms). Comment. Definitions of the between and within transformation: XXX.

```
# Between transformation
between_transformation <- panel_data %>%
  group_by(country) %>%
  mutate(oda_net_gdp_cap = oda_net / gdp_per_cap) %>%
  summarise(across(where(is.double), mean, na.rm = T))
```

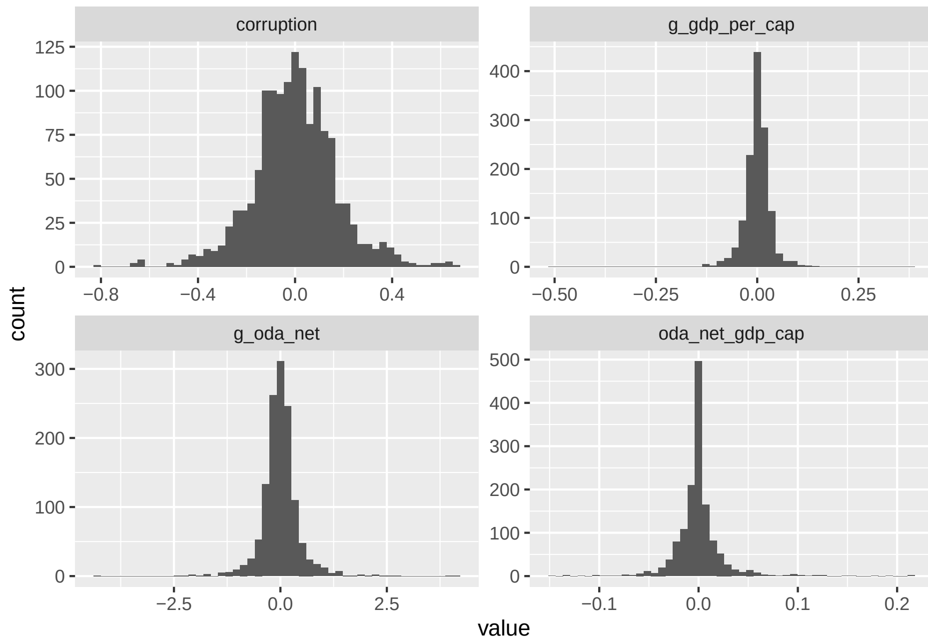
The important message of the below charts in the non-normality of the 'Between-transformed' variables. While we have no numeric information regarding the parameters of those distributions, we notice that the Between-transformed 'corruption' variable displays a positive skewness - it has a relatively fatter right tail. The Between-transformed 'ODA / GDP per capita' rather has the shape of a gamma distribution.

```
between_transformation %>%
  select(country, oda_net_gdp_cap, g_oda_net, g_gdp_per_cap, corruption) %>%
  pivot_longer(-country) %>%
  ggplot(aes(x = value)) +
  geom_histogram() +
  facet_wrap(~name, scales = 'free')
```



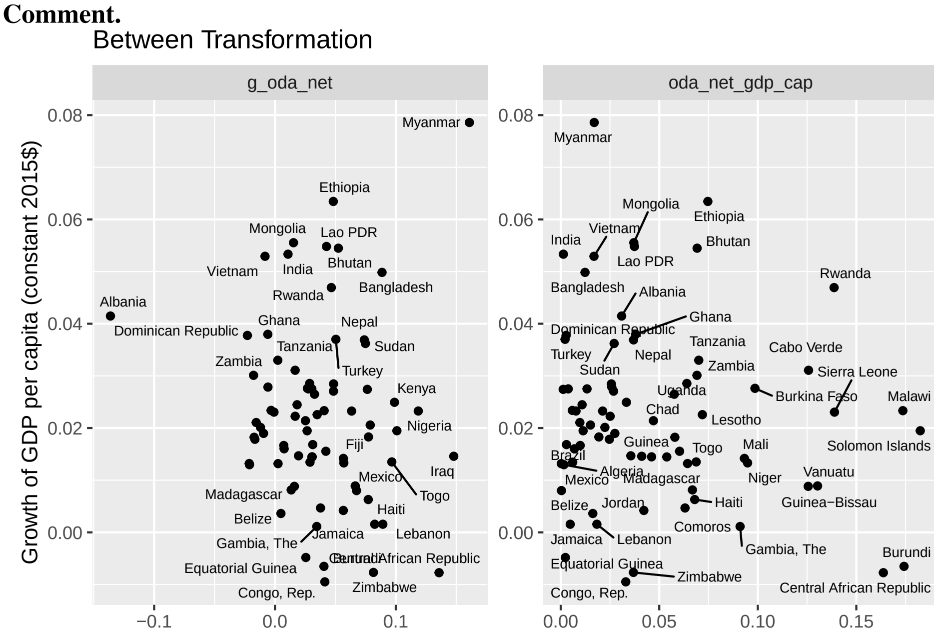
The most striking characteristic in the 'within-transformed' case is the apparent normality of the corruption variable and the 'ODA / GDP per capita' variable. Although those charts give no precisions about the kurtosis of those within-transformed variable, we can at least conclude that those variables are evenly-distributed around the mean.

```
# Within transformation
panel_data %>%
  group_by(country) %>%
  mutate(oda_net_gdp_cap = oda_net / gdp_per_cap) %>%
  mutate(across(where(is.double),~ . -  mean(., na.rm = T))) %>%
  select(country, oda_net_gdp_cap, g_oda_net, g_gdp_per_cap, corruption) %>%
  pivot_longer(-country) %>%
  ggplot(aes(x = value)) +
  geom_histogram(bins = 50) +
  facet_wrap(~name, scales = 'free')
```

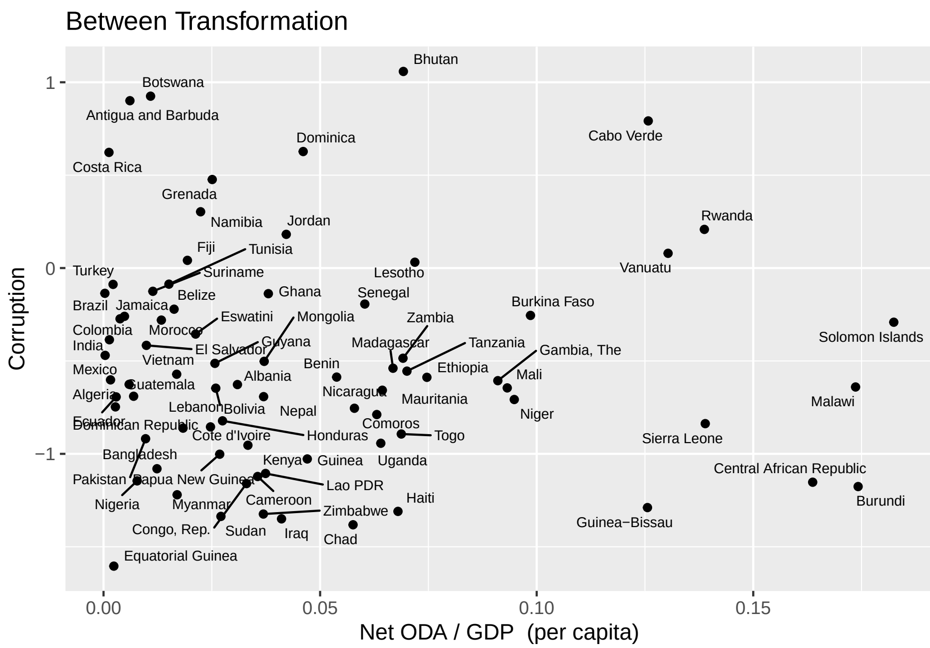

corruption
g_gdp_per_cap
g_oda_net
oda_net_gdp_cap
count
value

```r
between_transformation %>%
  pivot_longer(-c(country, g_gdp_per_cap)) %>%
  filter(name %in% c('g_oda_net', 'oda_net_gdp_cap')) %>%
  ggplot(aes(x = value, y = g_gdp_per_cap, label = country)) +
  geom_point() +
  facet_wrap(~name, scales = 'free') +
   geom_text_repel(size = 2.5) +
  labs(title = 'Between Transformation',
       x = '',
       y = 'Growth of GDP per capita (constant 2015$)')
```

6. Provide the 3 bivariate graphs (with acronyms for observations NIC12, for Nicaragua 2012) for between and within (hence 6 graphs) of growth of GDP/head (vertical axis) with (1) ODA/GDP, (2) the growth of ODA; of corruption index with ODA/GDP. Comment.




```r
between_transformation %>%
  ggplot(aes(x = oda_net_gdp_cap, y = corruption, label = country)) +
  geom_point() +
  geom_text_repel(size = 2.5) +
  labs(title = 'Between Transformation',
       x = 'Net ODA / GDP  (per capita)',
       y = 'Corruption')
```




```r
within_bivariate <- panel_data %>%
  group_by(country) %>%
  mutate(oda_net_gdp_cap = oda_net / gdp_per_cap) %>%
  mutate(across(where(is.double),~ . -  mean(., na.rm = T)),
         # '..$' : Regex pour sélectionner les deux (un point = n'importe quelle terme)
         # derniers ($) charactères du vecteur Year
         # '^..' : ^ pour selectionner les deux premiers charactères
         # '..$' : $ pour selectionner les deux derniers charactères
         country_year = paste0(country, '-', str_extract_all(year, '..$'))) %>%
  ungroup() %>%
  select(country_year, country, year, gdp_per_cap, oda_net_gdp_cap, g_oda_net, g_gdp_per_cap, corruption)
```

```r
fatal_btw_mod <- plm(corruption ~ gdp_per_cap,
                     data = within_bivariate,
                     index = c("country", "year"),
                     model = "within")

summary(fatal_btw_mod)
```

```
#> Oneway (individual) effect Within Model
#>
#> Call:
#> plm(formula = corruption ~ gdp_per_cap, data = within_bivariate,
#>     model = "within", index = c("country", "year"))
#>
#> Balanced Panel: n = 77, T = 18, N = 1386
#>
#> Residuals:
#>       Min.    1st Qu.     Median    3rd Qu.       Max.
#> -0.8239265 -0.1005177 -0.0037068  0.0999840  0.6496487
#>
#> Coefficients:
#>               Estimate Std. Error t-value Pr(>|t|)
#> gdp_per_cap 1.3025e-05 7.8949e-06  1.6499  0.09921 .
```

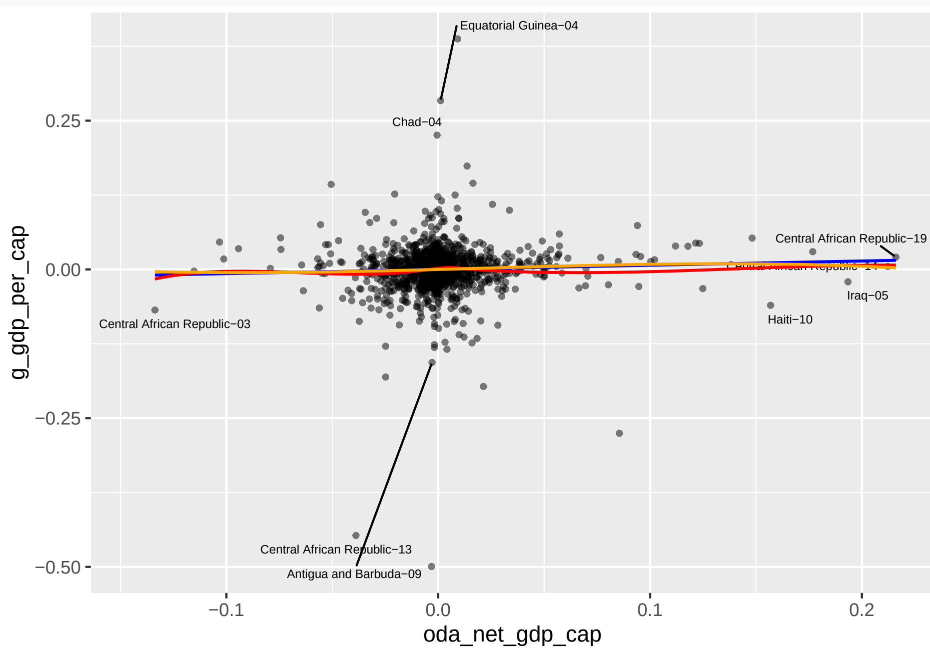
```
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Total Sum of Squares:    41.368
#> Residual Sum of Squares: 41.282
#> R-Squared:      0.0020768
#> Adj. R-Squared: -0.056669
#> F-statistic: 2.72207 on 1 and 1308 DF, p-value: 0.09921
```

```
within_bivariate %>% lm(corruption ~ -1 + gdp_per_cap, data = .)
```

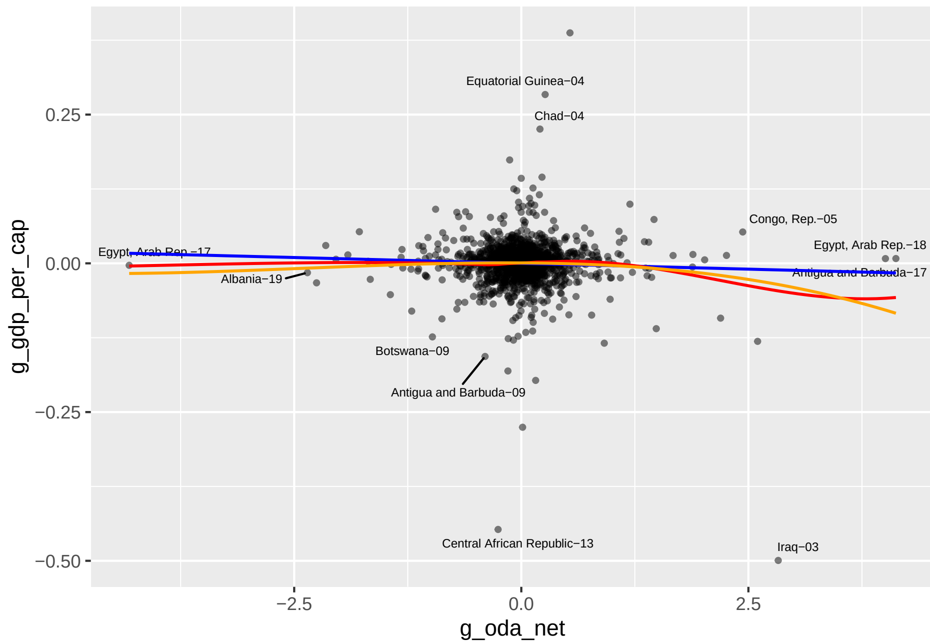
```
#>
#> Call:
#> lm(formula = corruption ~ -1 + gdp_per_cap, data = .)
#>
#> Coefficients:
#> gdp_per_cap
#>   1.303e-05
```

```
plot_biv <- function(data, x, y) {

  data %>%
    ggplot(aes(x = {{ x }}, y = {{ y }}, label = country_year)) +
    geom_point(size = 1, alpha = 0.5) +
    geom_text_repel(size=2) +
    geom_smooth(method = lm, se = FALSE, color = 'blue', size = 0.7) +
    geom_smooth(method = loess, se = FALSE, color = 'red', size = 0.7) +
    geom_smooth(method = lm, formula = y ~ splines::bs(x, 3), se = FALSE, color = 'orange', size = 0.7)

}
```

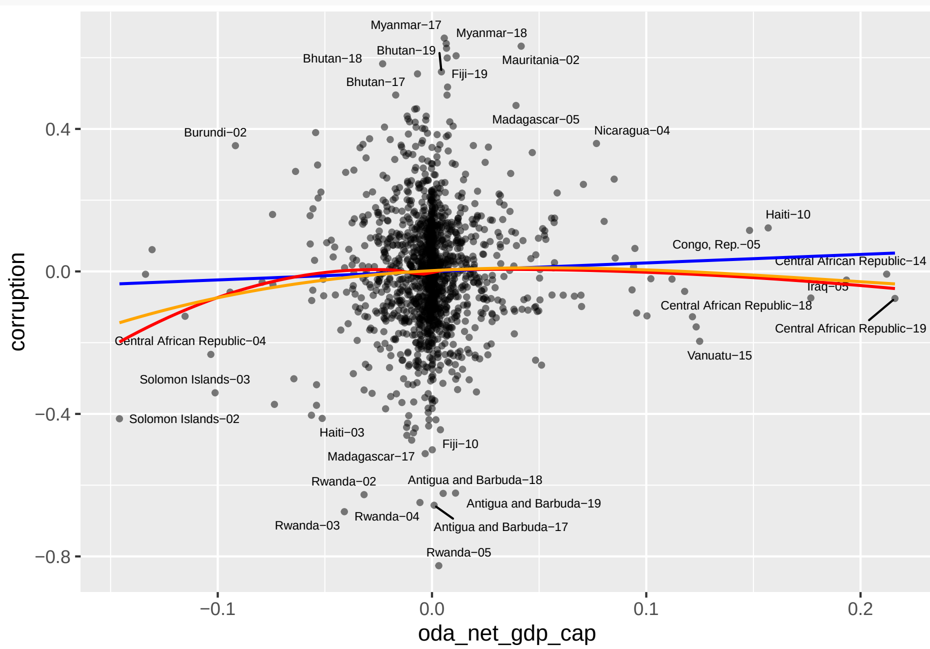
```
within_bivariate %>%
  plot_biv(x = oda_net_gdp_cap, y = g_gdp_per_cap)
```




```
within_bivariate %>%
  plot_biv(x = g_oda_net, y = g_gdp_per_cap)
```




```
within_bivariate %>%
  plot_biv(x = oda_net_gdp_cap, y = corruption)
```



```
between_transformation %>%
  select(c(g_oda_net, g_gdp_per_cap, gdp_per_cap, oda_net_gdp_cap, corruption, population)) %>%
  #cor()%>%
datasummary_correlation(title = 'Correlation matrix') %>%
kable_styling(latex_options = c("striped", "hold_position"))
```

Table 1: Correlation matrix

	g_oda_net	g_gdp_per_cap	gdp_per_cap	oda_net_gdp_cap	corruption	population
g_oda_net	1	.	.	.	.	.
g_gdp_per_cap	−0.07	1	.	.	.	.
gdp_per_cap	−0.04	−0.17	1	.	.	.
oda_net_gdp_cap	0.14	−0.15	−0.52	1	.	.
corruption	−0.16	0.14	0.43	−0.05	1	.
population	−0.04	0.26	−0.06	−0.20	−0.02	1

7. Comment the between versus within correlation matrix for the 6 variables in this order

```
fatal_fe_mod <- plm(corruption ~ gdp_per_cap,
                    data = panel_data,
                    index = c("iso2c", "year"),
                    model = "between")

summary(fatal_fe_mod)
```

```
#> Oneway (individual) effect Between Model
#>
#> Call:
#> plm(formula = corruption ~ gdp_per_cap, data = panel_data, model = "between",
#>     index = c("iso2c", "year"))
#>
#> Balanced Panel: n = 77, T = 18, N = 1386
#> Observations used in estimation: 77
#>
#> Residuals:
#>      Min.    1st Qu.    Median   3rd Qu.      Max.
#> -1.817354 -0.365234 -0.038984  0.263649  1.652166
#>
#> Coefficients:
#>                Estimate  Std. Error t-value  Pr(>|t|)
#> (Intercept) -7.7825e-01  8.7716e-02 -8.8724 2.604e-13 ***
#> gdp_per_cap  8.2596e-05  2.0191e-05  4.0908 0.0001071 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Total Sum of Squares:    26.267
#> Residual Sum of Squares: 21.475
#> R-Squared:      0.18242
#> Adj. R-Squared: 0.17152
#> F-statistic: 16.7344 on 1 and 75 DF, p-value: 0.00010711
```

```
between_transformation %>% lm(corruption ~ gdp_per_cap, data = .)
```

```
#>
#> Call:
#> lm(formula = corruption ~ gdp_per_cap, data = .)
#>
#> Coefficients:
#> (Intercept)  gdp_per_cap
#>  -0.7782526    0.0000826
```

8. Run a one-way fixed effect foreign aid regression on ODA/GDP function of Ln(Population) and Ln(GDP/head). Comment.

9. Run a one-way fixed effect of Corruption Index function of Ln(GDP/head), of ODA/GDP and the growth of ODA. Comment.

10. Run a one-way fixed effect with the growth of GDP/head function of Ln(GDP/head), ODA/GDP, the growth of ODA and the Corruption index.

11. Propose an additional interesting estimation using this database.

```
panel_data %>%
  ggplot(aes(x = year, y = gdp_ppp)) +
  geom_point() +
  geom_line(data = panel_data %>% group_by(year) %>% summarise(gdp_mean = mean(gdp_ppp)),
            aes(x = year, y = gdp_mean), col = "blue") +
  scale_x_continuous(labels = as.character(unique(panel_data$year)),
                     breaks = unique(panel_data$year)) +
  labs(x = "Year", y = "GDP Per capita") +
  theme(axis.text.x = element_text(angle = 90))
```

12. Compute the between and within transformations of the 11 variables over the full period. Provide histograms for ODA/GPD, growth of ODA, growth of GDP/head for both between and within transformed. Comment.