Estimating the Effect(s) of Florida's Family Transition Program Through Instrumental Variables

Daniel Chen

June 2021

Preparing the Data

Load in libraries.

```
library(tidyverse)
library(foreign)
library(zoo)
library(broom)
library(janitor)
library(estimatr)
library(kableExtra)
```

We'll need to merge and clean both the administrative data and survey data before estimating treatment effects, if any. The functions below load in the STATA files as tibbles, merge them, and remove duplicate columns.

```
data_loader <- function(path) {

  df <- as_tibble(read.dta(path))

  return(df)

}

data_merger <- function(dataframe_one, dataframe_two, by) {

  merged_df <- merge(dataframe_one, dataframe_two, by) %>%
    rename_at(vars(ends_with(".x")), ~str_replace(., "\\..$", "")) %>%
    select(-ends_with(".y")) %>%
    as_tibble()

  return(merged_df)

}

# Load data
admin <- data_loader(paste("/Users/danielchen/Desktop/UChicago/Year Two/Autumn 2020/",
                           "Program Evaluation/Problem Sets/Problem Set 2/ftp_ar.dta",
```

```
                              sep = ""))
survey <- data_loader(paste("/Users/danielchen/Desktop/UChicago/Year Two/Autumn 2020/",
                            "Program Evaluation/Problem Sets/Problem Set 2/ftp_srv.dta",
                            sep = ""))
# Merge data
merged_data <- data_merger(dataframe_one = admin, dataframe_two = survey, by = "sampleid")
```

The data should look like the following:

```
kable(head(merged_data[,1:11]))
```

sampleid	e	cflag	longtdec	b_aidst	gender	ethnic	marital	afdctime	afdchild	higrade
1	0	1	1	2	2	5	5	2	3	12
100	0	NA	2	2	2	1	1	2	2	12
1000	0	NA	1	2	2	1	5	2	3	12
1004	0	1	1	1	2	1	4	5	1	12
1007	0	1	5	1	2	1	3	4	3	12
1009	0	1	5	2	2	5	1	3	4	9

I've only shown the first 11 columns because thare are a total of 1,983 total variables in the merged data. In the next section, we'll determine which variables we're interested in.

Understanding Key Variables & Summary Statistics

- `e` is from the administrative data. It is the original treatment dummy where 0 indicates a family randomly assinged to the control group and 1 indicates a family randomly assigned to the treatment group. The treatment group received time limited welfare benefits.
- `fmi2` is from the survey data. Families were asked if they believed they were subject to the time limit or not. For now, we'll take a look at the number of families who did and did not believe in the time limit although respondents were allowed to skip the question or say "don't know".

```
# Define function to return a table summarizing number of families who believed
# in time limit vs. those who did not.
return_counts <- function(dataframe, variable) {

   # Adds quotes around user entered variable which can be acessed later
  var <- deparse(substitute(variable))

  counts <- dataframe %>%
    count({{ variable }}) %>%
    rename(category = {{ variable }},
           count    = n) %>%
    mutate(category = as.character(category),
           category = str_replace_na(category),
           category = str_replace_all(category, c("1"  = "Believed in time limit",
                                                  "2"  = "Didn't believe in time limit",
                                                  "8"  = "Don't know",
                                                  "NA" = "No response"))) %>%
    add_row(category = "Valid responses",
            count = length(which(!is.na(dataframe[[var]]))) == FALSE))
```

```
  return(counts)

}

# Show possible responses to fmi2 broken out
summary_counts <- return_counts(dataframe = merged_data, variable = fmi2)
kable(summary_counts, "pipe")
```

category	count
Believed in time limit	666
Didn't believe in time limit	365
Don't know	118
No response	580
Valid responses	1149

For this analysis, we'll only work with those who said they believed in the time limit or did not believe in the time limit. We'll filter everyone else out. Additionally, we want to use `fmi2` as our instrumental variable (more on that later), so we'll create a new dummy variable named `TLyes`. If `fmi2` equals 1 then `TLyes` is equal to 1, meaning the family believed in the time limit. Otherwise `TLyes` takes on the value of 0 meaning the family does not believe in the time limit.

```
# Define function that creates new dummy variable
dummy_creator <- function(dataframe, variable) {

  dataframe <- dataframe %>%
    filter({{ variable }} == 1 | {{ variable }} == 2) %>%
    mutate(TLyes = case_when({{ variable }} == 1 ~ 1,
                             {{ variable }} == 2 ~ 0))

  return(dataframe)

}

# Create new dataframe with TLyes dummy variable.
merged_data <- dummy_creator(dataframe = merged_data, variable = fmi2)

# Show that TLyes should line up with fmi2
merged_data %>%
  select(sampleid, fmi2, TLyes) %>%
  head(10) %>%
  kable()
```

sampleid	fmi2	TLyes
1007	1	1
1018	1	1
1022	1	1
1023	1	1
1024	1	1
1026	1	1
1028	1	1
103	2	0
1034	1	1
1039	1	1

We may, however, be worried that some families may have been assigned to the treatment group randomly, but they didn't believe in the time limit when in reality we know that they had time limited benefits because they were assigned to the treatment group. The opposite is true. It's possible that families assigned to the control group believed in the time limit when in reality their benefits were not subject to time. We can see this dynamic by generating a crosstab of those originally assigned to treatment/control (variable `e`) against whether they believed in the time limit or not (variable `TLyes`).

```
# Define function to generate crosstabs
generate_xtabs <- function(dataframe, variable_one, variable_two) {

  tabs <- dataframe %>%
    tabyl({{ variable_one}}, {{ variable_two }}) %>%
    as_tibble() %>%
    # NOTE: Typically bad practice to use spaces and punctuation when naming
    # variables, but I'm making an exception for a cleaner output for the reader.
    rename(`Original Assignment` = e,
           `Believe` = `1`,
           `Don't Believe` = `2`) %>%
    mutate(`Original Assignment` = as.character(`Original Assignment`),
           `Original Assignment` = str_replace_all(`Original Assignment`,
                                                   c("0" = "Control",
                                                     "1" = "Treatment")))

  return(tabs)

}

# Generate tabs
generate_xtabs(dataframe = merged_data, variable_one = e, variable_two = fmi2) %>%
  kable("pipe")
```

Original Assignment	Believe	Don't Believe
Control	205	300
Treatment	461	65

It's evident from the table above that there were some households (12.4%) assigned to the treatment group who didn't believe in the time limit (even though their benefits were indeed time limited) and some households (40.6%) assigned to the control group who believed in the time limit (even though their benefits were not time limited).

Estimating Effects through Ordinary Least Squares (OLS)

Through multivariate linear regression, we can estimate the effect of time limits on outcomes such as length of welfare receipt. Our main variable of interest is the assignment to treatment or control (`e`) and we'll control for number of variables that may otherwise bias our estimates if they aren't included in the regression. Before running the regression, we'll need to impute means for variables with `NaN` values.

```
# Define function to impute NA values
mean_imputer <- function(dataframe, variables, calculation) {

  imputed_data <- dataframe %>%
    mutate(across(all_of({{ variables }}), na.aggregate, FUN = calculation))
  # NOTE: mutate(across()) can also take in a vector of values referring to the
  # names of columns without the curly-curly notation of tidyverse

  return(imputed_data)

}

# Define vector containing names of variables that need means imputed
vars <- c("male",
          "agelt20",
          "age2534",
          "age3544",
          "agege45",
          "black",
          "hisp",
          "otheth",
          "marapt",
          "martog",
          "nohsged",
          "applcant")

# Call on function to impute means
merged_data <- mean_imputer(dataframe = merged_data, variables = vars, calculation = mean)

# Check out data to ensure no missing data
merged_data %>%
  select(all_of(vars)) %>%
  # the tilde denotes an unnamed lambda function. When using lambdas we must use
  # a period to reference the function's argument AND NOT anything else such as
  # an x or y.
  summarise(across(everything(), ~sum(is.na(.)))) %>%
  kable()
```

male	agelt20	age2534	age3544	agege45	black	hisp	otheth	marapt	martog	nohsged	applcant
0	0	0	0	0	0	0	0	0	0	0	0

From the table above, we can see that there are no longer any variables containing missing observations. With complete observations, OLS will estimate the coefficients below:

```
# Store regressors to vector
regressors <- c("TLyes",
                "male",
```

```
                "agelt20",
                "age2534",
                "age3544",
                "agege45",
                "black",
                "hisp",
                "otheth",
                "martog",
                "marapt",
                "nohsged",
                "applcant",
                "yremp",
                "emppq1",
                "yrearn",
                "yrearnsq",
                "pearn1",
                "recpc1",
                "yrrec",
                "yrkrec",
                "rfspc1",
                "yrrfs",
                "yrkrfs")

# Store dependent variables to vector
dependent_vars <- c("vrecc217",
                    "vrecc2t5",
                    "vrecc6t9",
                    "vrec1013",
                    "vrec1417")

# Create helper function to run regressions
run_ols <- function(dep_var, ind_var, dataframe) {
  suppressWarnings(
    lm(as.formula(paste(dep_var, "~", paste(ind_var, collapse = " + "))), data = dataframe)
  )
}

# Function that utilizes helper to loop through variables to run ols and return
# results summarized into a dataframe
return_ols_results <- function(dep_var, func, ind_var, dataframe) {

  results_df <- map(dep_var, func, ind_var, dataframe) %>%
    map(tidy) %>%
    bind_rows() %>%
    filter(term == "TLyes") %>%
    mutate(outcome = dep_var) %>%
    select(outcome, estimate:p.value)

  return(results_df)

}

# Run linear regression
```

```
return_ols_results(dep_var = dependent_vars,
                   func = run_ols,
                   ind_var = regressors,
                   dataframe = merged_data) %>%
  kable("pipe")
```

outcome	estimate	std.error	statistic	p.value
vrecc217	0.0273523	0.0171738	1.5926804	0.1115460
vrecc2t5	0.0340127	0.0190171	1.7885319	0.0739911
vrecc6t9	0.0078619	0.0278059	0.2827430	0.7774320
vrec1013	-0.0161295	0.0310710	-0.5191162	0.6037939
vrec1417	-0.0720054	0.0313818	-2.2944929	0.0219674

The table above shows OLS results where the `outcome` column refers to different economic measures throughout the study. For example, all the variables starting with "vrec" are binary outcome variables that indicate whether or not a family ever received benefits throughout different points of the four year study period. The results suggest that in year three (`vrec1013`) and year four (`vrec1417`), families who believed in the time limit were less likely to ever receive benefits relative to families who did not believe in the time limit by roughly one to seven percent.

The interpretation is plausible. If I believed that my benefits were restricted after a certain period of time, then I would be less likely to rely on these benefits as time passes. Though the interpretation is logical, this doesn't guarantee that our estimates are correct. In the next section, I explain problems with using ordinary least squares.

Limitations of Ordinary Least Squares

Ordinary Least Squares relies on the assumption that the independent variable is *exogenous* for valid estimates. However, the treatment variable is TLyes, and this variable is *endogenous*. As seen in the Understanding Key Variables + Summary Statistics section, there was clearly some confusion about who the time limits actually applied to. There may have been genuine confusion as to the treatment received by the treated, but families may also be *self-selecting* into treatment and control groups.

In other words, because individuals seek to maximize their outcomes, individuals and families may self-select into the group that does not believe in time limited benefits. Arguably, it's better to receive unlimited benefits as opposed to limited benefits, so individuals may not believe in the time limit even if they were randomly assigned to the treatment group which imposed a time limit.

In short, OLS will not return consistent estimates of the effect of believing in the time limit. In the next section, I explore the possibility of using an instrumental variable and elaborate on the required assumptions.

Instrumental Variable Approach

When the explanatory variable (in this case `TLyes`) is correlated with the error term, OLS will return biased results. The instrumental variable introduces a third variable that changes the explanatory variable but has no direct effect on the dependent variable. It is only through the explanatory variable that the instrument has an effect on the dependent variable.

In this scenario, we need a third variable, Z, that influences the exogenous part of treatment `TLyes` (or the part that is not correlated with the endogenous part of the error term). By this logic, Z must be uncorrelated with the error term.

In the following subsections, I will review the assumptions necessary to use an instrumental variable and evaluate whether or not they hold up in this context. I propose using `e` or the original experimental treatment indicator as an instrument for `TLyes`.

1. Exclusion Restriction

The instrument must not be correlated with the error term and must be exogenous. Mathematically speaking, we can say that $cor(z, u) = 0$ and $E(Y0_i|Z = 1, D = 0) = E(Y0_i|Z = 0, D = 0)$. In other words, the untreated potential outcome for an individual when the instrument is turned on is equal to the untreated potential outcome for an individual when the instrument is turned off. The instrument itself has no effect on the potential outcome conditional on the same unit being assigned to control or treatment.

This condition is fundamentally untestable because we never observe the error term, however, it is likely met. Because the original experimental dummy (`e`) is exogenous through random assignment, it's plausible that the assignment itself has no direct effect on any of the dependent variables. It is only through the belief (or disbelief) in the time limit (`TLyes`) that there is an effect on the variables of interest.

2. Relevance

The instrument must be correlated with the treatment dummy.

In this scenario, the assignment to treatment or control must be correlated with the belief in the time limit. If the instrument changes, then we also expect the belief in the time limit to also change.

This condition is satisfied. Below, I test the correlation between the two variables and test against the decision rule. When using an instrumental variable, the decision rule states that our F-stat must be greater than 10 when running a first-stage regression where the endogenous variable is explained by the instrument (`TLyes ~ e`)

```
# Define function to check if IV assumptions are met
iv_test <- function(dataframe, instrument, endogenous) {

  inst <- dataframe %>% select({{ instrument }})
  endo <- dataframe %>% select({{ endogenous }})

  # Instrument test for relevance
  correlation <- cor(inst, endo)
  cor_rounded <- round(correlation, 1)
  # Instrument first stage test. F-stat must be greater than 10
  eval(substitute(ols <- lm(endogenous ~ instrument, data = dataframe)))
  f_stat <- ols %>%
    tidy() %>%
    filter(term == "e") %>%
    mutate(f_stat = statistic * statistic) %>%
    select(f_stat)

  # Store results from correlation and first stage tests
  output_statement <- paste(" The correlation is:", correlation, "\n",
                            "The f-stat is:", f_stat)
  # Store decision statement based on correlation test and f-stat
  decision_statement <- if (cor_rounded >= .5 & f_stat > 10) {
    paste("The variables move together, and the f-stat is greater than 10",
          "which meets the IV criteria.")
  } else if (cor_rounded < .5 & f_stat > 10) {
```

```
    paste("The variables don't move together, but the f-stat is greater than 10",
          "failing to meet both IV criteria.")
  } else if (cor_rounded > .5 & f_stat < 10) {
    paste("The variables move together, but the f-stat is less than 10",
          "failing to meet both IV criteria.")
  } else {
    paste("The variables don't move together, and the f-stat is less than 10",
          "failing to meet any IV criteria.")
  }

  return(cat(output_statement, "\n", decision_statement))

}

# Call on function
iv_test(dataframe = merged_data, instrument = e, endogenous = TLyes)
```

```
##  The correlation is: 0.491814170149264
##  The f-stat is: 328.307021121776
##  The variables move together, and the f-stat is greater than 10 which meets the IV criteria.
```

From the output above, it's clear that both assumptions are met in using an instrumental variable.

Instrumental Variable Estimation

In this section, I estimate the effect of believing in the time limit on employment, welfare receipt, and income outcomes where `e` is the instrument for `TLyes`. The dataframe below displays the results for each regression.

```
# Helper function to run iv estimation using iv_robust() function
estimate_iv <- function(variable, ind_vars, dataframe) {

  tidy(iv_robust(as.formula(paste(variable, "~ ", paste(ind_vars, collapse = " +"),
                                  "| e + ", paste(ind_vars[-1], collapse = "+ "))),
                 data = dataframe))

}

# Function to return IV results in a dataframe
return_iv_results <- function(dep_vars, ind_vars, dataframe) {

  map(dep_vars, estimate_iv, ind_vars, dataframe) %>%
    bind_rows() %>%
    filter(term == "TLyes") %>%
    select(outcome, estimate:p.value)

}

# Store starting patterns for variables of interest into a vector of strings
patterns <- c("vempq",
              "vemp1",
              "vrecc",
```

```
              "vrec1")

# Grab the names of outcome variables of interest and store to vector
outcome_vars <- merged_data %>%
  select(starts_with(patterns),
         tinc217, tinc2t5, tinc6t9, tinc1013, tinc1417) %>%
  colnames()

# Return IV estimates as a dataframe
return_iv_results(dep_vars = outcome_vars,
                  ind_vars = regressors,
                  dataframe = merged_data) %>%
  kable("pipe")
```

outcome	estimate	std.error	statistic	p.value
vempq217	0.0548915	0.0428326	1.2815361	0.2003007
vempq2t5	0.0698992	0.0600985	1.1630777	0.2450738
vempq6t9	0.2365670	0.0607478	3.8942485	0.0001050
vemp1013	0.2678774	0.0606069	4.4199132	0.0000109
vemp1417	0.0798701	0.0602580	1.3254688	0.1853166
vrecc217	0.0027516	0.0357149	0.0770445	0.9386035
vrecc2t5	0.0087701	0.0389479	0.2251738	0.8218898
vrecc6t9	-0.0149909	0.0566620	-0.2645674	0.7913969
vrec1013	-0.1312344	0.0638887	-2.0541103	0.0402230
vrec1417	-0.3988559	0.0671881	-5.9364061	0.0000000
tinc217	4699.4564434	2090.6074912	2.2478904	0.0247991
tinc2t5	292.8061571	488.2033664	0.5997627	0.5487995
tinc6t9	1154.3221869	603.4980743	1.9127189	0.0560678
tinc1013	2002.3335106	701.4906981	2.8543978	0.0044004
tinc1417	1249.9945887	795.5167721	1.5712988	0.1164277

Effect on Employment

In analyzing the employment variables (those that start with "vemp"), the IV estimates reveal that those who believed in the time limit were more likely to be employed throughout the study. However, these differences are only statistically significant throughout the middle of the study. By the end, employment levels were no different - statistically speaking - between those who believed in the time limit versus those who did not.

Effect on Welfare Receipt

Over the course of the entire study, when looking at welfare receipt variables (those that start with "vrec"), there was no statistical difference in the amount of welfare received between those who believed in the time limit and those who did not (variable `vrecc217`). However, when looking at the period specific effects, by year three (`vrec1013`) and year four (variable `vrec1417`), those who believed in the time limit were between 13% and 40% more likely to be employed than those who did not believe in the limit. These differences are statistically significant at the 95% level of confidence.

When comparing these results to the OLS results above, it's clear that OLS has not only underestimated the effect of believing in the time limit in year three and year four, but OLS has also misidentified statistical significance.

A note on standard errors: Finally, when contrasting the IV estimates to OLS, it's clear that IV estimates will always return larger standard errors compared to OLS. In this scenario, the standard errors are larger by about two-fold. Intuitively, this makes sense because we're using less data to explain outcomes. We're using the variation in the perceived time limit explained by the assignment to treatment or control that is uncorrelated with the disturbance term. Since we're working with less data, we are less certain of our results and, consequently, the standard errors increase.

Effect on Income

Income is the last outcome of interest (variables starting with "tinc"). The data show that those who believed in the time limit had greater levels of income than those who did not. However, the increase is only statistically different from zero in year three of the study (`tinc1013`).

Conclusion

There are two key takeaways from this project:

1. Ordinary Least Squares returns biased results when our explanatory variable is endogenous. In this scenario, participant self-selection invalidates our estimates when using OLS. We need an exogenous, instrumental variable, in order to make causal claims.
2. From a causal inference perspective, at the end of the day, it appears that imposing a time limit on welfare benefits had no effect on employment, welfare, or income outcomes. After four years, participants who believed in the time limit fared no different from participants who did not believe in the time limit.

However, statistical magnitudes can be different from real world magnitudes. For example, in year four (`tinc1417`), individuals who believed in the time limit had a higher income, on average, by about $1250. Since 0 is included in the 95 percent confidence interval, statistical analyses would tell us that the increase is not meaningful. Even so, it's difficult to make the case in a real world setting that an added $1250 dollars is nothing - especially for individuals and families who lie on the lower end of the income distribution.