Years in office, promotion incentives, and fiscal behavior

Siwei Dai

December 5, 2021

```
library(tidyverse)
library(estimatr)
library(modelsummary)
options(modelsummary_format_numeric_latex = "mathmode")
datapath <- './ps_data/FullData.csv'
df <- as_tibble(read.csv(datapath))
biodatapath <- './ps_data/BiographicalData.csv'
biodf <- as_tibble(read.csv(biodatapath))
fiscdatapath <- './ps_data/fiscal.dta'
fiscdf <- haven::read_dta(fiscdatapath)
```

Data description

This project is based on two datasets.

Two datasets are used for analysis. The first dataset is the "Chinese Political Elite Database" (CPED), compiled by Junyan Jiang (accessed at https://www.junyanjiang.com/data.html). This dataset includes the biographical and career information of all mayors and municipal party secretaries from 2000 to 2015, provincial governors and party secretaries (1995 - 2015).

Another dataset documents county-level fiscal revenue and expenditure from 1994 to 2007. The dataset is accessed from the replication data of Xu Xu (2021) who purchased and digitized the original "Fiscal Statistics of Cities and Counties" compiled by the Budget Department of the Ministry of Finance of China. The original paper concerns the effect of implementing digital surveillance programs on public security spending.

Before running analyses, I tidied and merged the two datasets to produce a leader-year dataset.

On the CPED dataset, I first created the key variables needed for analysis; I then converted the CPED dataset to a county-year level dataset using the start date and end date of politicians' terms.

On the county-level fiscal data, I dealt with certain coding discrepancies within the data. In some cases, missing data is coded `NA` and in other cases, it's coded 0. I recoded the missing data points to `NA`.

Finally, I merged the datasets together. As some politicians may only start their position closer to December, they may not have sufficient control over the fiscal expenditure and income that particular year. For politicians who came into office between July to December, I coded the first year of their tenure the next year.

The final dataset has 11,062 leader-year observations. Each observation documents who the leaders were at a particular city in a given year, how much was the fiscal expenditure and revenue.

```
df_tidy <- df %>%
    # translate the variables into English
```

```r
    # exper_num refers to the career stage
    rename('identifier' = 用户编码,
           'name' = 姓名,
           'job' = 标志位,
           'position_ori' = 级别,
           'exper_num' = 经历序号,
           'start_date' = 起始时间.YYYY.MM.DD.,
           'end_date' = 终止时间..YYYY.MM.DD..,
           'prefectural_code' = 二级关键词编码,
           'provincial_code' = 一级关键词编码,
           'position_code' = 职务一级关键词编码) %>%
    select(identifier, name, job, position_ori, exper_num, start_date, end_date,
           prefectural_code, provincial_code, position_code) %>%
    # impute prefectural code from provincial code if the prefectural code is missing due to being a pr
    mutate(prefectural_code = ifelse(is.na(prefectural_code), provincial_code, prefectural_code)) %>%
    # recode the position from strings into numeric; larger the number, higher the position
    mutate('position_numeric' = case_when(position_ori == '无级别' ~ 0,
                                          position_ori == '小于副处' ~ 1,
                                          position_ori == '副处' ~ 2,
                                          position_ori == '正处' ~ 3,
                                          position_ori == '副厅' ~ 4,
                                          position_ori == '正厅' ~ 5,
                                          position_ori == '副部' ~ 6,
                                          position_ori == '正部' ~ 7,
                                          position_ori == '副国' ~ 8,
                                          position_ori == '正国' ~ 9)) %>%
    # Code the term length
    mutate('term_length' = (as.numeric(as.Date(end_date) - as.Date(start_date)))/365,
           'start_year' = ifelse(as.numeric(format(as.Date(start_date), '%b')) >= 7, as.numeric(format(a
           'end_year' = as.numeric(format(as.Date(end_date), '%Y'))) %>%
    # Create a `leader` variable to code if the official is a municipal/provincial governor or party se
    mutate(governor = job %in% c(" 市长", " 省长"),
           mayor = (job == " 市长"),
           party_secretary = job %in% c(" 市委书记", " 省委书记"),
           leader = governor | party_secretary) %>%
    # Create promotion or demotion
    group_by(name) %>%
    # rank the experiences in the temporal order
    arrange(exper_num, by_group = TRUE) %>%
    mutate(dem = (lag(position_numeric) > position_numeric),
           # or being moved to an inconsequential position
           prom = (lead(position_numeric) > position_numeric))
```

Converting to county-year dataset

```r
df_tidy_year <- df_tidy %>%
    # filter out NAs, otherwise there would be errors in mutating new variables
    filter(!is.na(start_year) & !is.na(end_year) & !is.na(prefectural_code)) %>%
    mutate('year' = map2(start_year, end_year, `:`)) %>%
    unnest(cols = c(year)) %>%
    # filter out the experiences after the fiscal data is available
```

```r
    filter(year >= 1994) %>%
    # Create two dummies
    mutate('term_year' = year - start_year + 1,
           'first_year' = ifelse(year == start_year, 1, 0),
           'last_year' = ifelse(year == end_year, 1, 0))
```

Merge the dataset with spending data and biographical information

```r
# merge with fiscal data using regional code and year
# first, filter only include those who have ascended to the position of making decisions
biodf_tidy <- biodf %>%
    rename('bio_identifier' = X,
           'birth_date' = 出生日期.YYYY.MM.DD.,
           'discipline_date' = '查处. 罢黜时间',
           'discipline_cause' = '查处. 罢黜原因') %>%
    select(bio_identifier, birth_date, discipline_date, discipline_cause) %>%
    mutate('birth_year' = as.numeric(format(as.Date(birth_date), '%Y')),
           'disciplined' = (discipline_cause == '贪污腐败' | discipline_cause == '违纪'),
           'discipline_year' = as.numeric(format(as.Date(discipline_date), '%Y')))

fiscdf_tidy <- fiscdf %>%
    filter(!is.na(admcode) & admcode %in% unique(df_tidy$prefectural_code)) %>%
    # Code the income and expenditure observations that are 0 to NA
    mutate(income_ttl = ifelse(income_ttl == 0, NA, income_ttl),
           exp_ttl = ifelse(exp_ttl == 0, NA, exp_ttl),
           exp_shbz = ifelse(exp_shbz == 0, NA, exp_shbz))

df_merged <- fiscdf_tidy %>%
    left_join(df_tidy_year, by = c('admcode' = 'prefectural_code',
                                   'year' = 'year')) %>%
    left_join(biodf_tidy, by = c('identifier' = 'bio_identifier')) %>%
    mutate('income_log' = log(income_ttl),
           'exp_log' = log(exp_ttl),
           'social_exp_log' = log(exp_shbz),
           'edu_exp_log' = log(exp_jy),
           'age' = year - birth_year,
           'age50' = age >= 50,
           'completeterm' = (term_year >= 3)) %>%
    # filter out the leaders
    filter(leader == 1) %>%
    arrange(admcode, year)
```

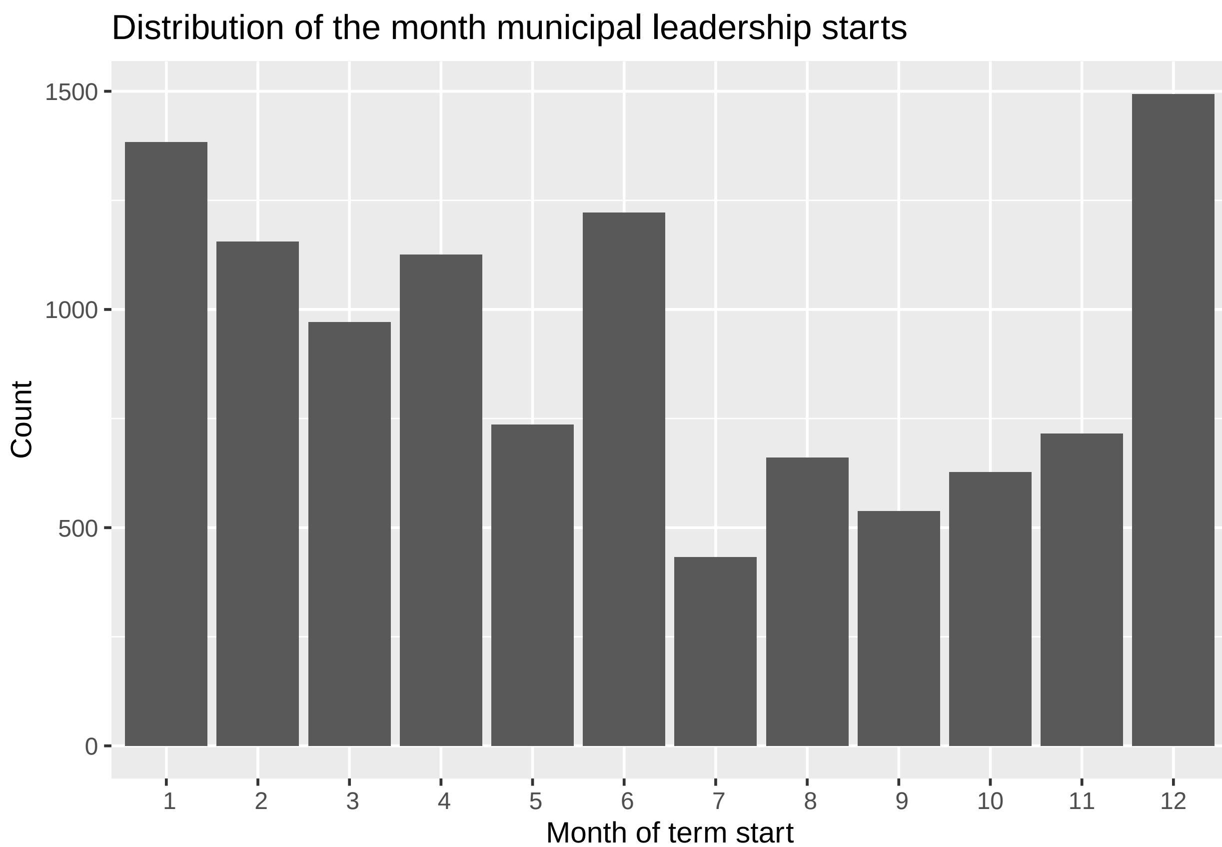
Summary Statistics & Visualizations

How many observations

Distribution of start and end date

One concern with merging fiscal data with political cycles is how much influence a politician has over fiscal spending and revenue. If a politican is appointed later in the year, it would be improbable

```
fig_distofstartdate <- df_merged %>%
    filter(!is.na(format(as.Date(start_date), '%b'))) %>%
    ggplot(aes(x = format(as.Date(start_date), '%b'))) +
    geom_bar() +
    labs(x = 'Month of term start',
         y = 'Count',
         title = "Distribution of the month municipal leadership starts")
fig_distofstartdate
```

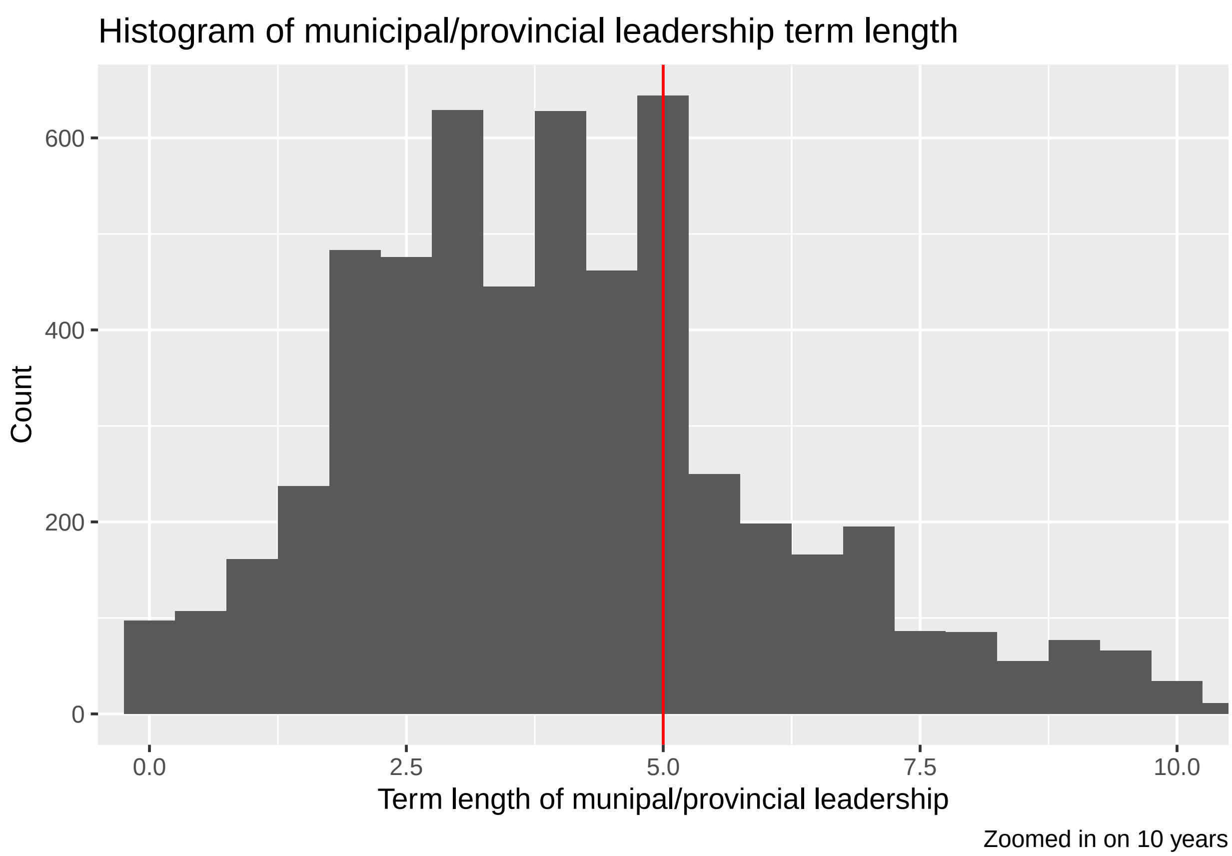



```
# ggsave("./FinalPJ/fig_distofstartdate.png")
```

Average Term length

There is a formal term limit for Chinese mayors and provincial governors: 5 years. Yet this formal term limit is not strictly enforced. As shown in the following figure, most of Chinese mayors are promoted or transferred to other positions before reaching the 5-year term limit(the red vertical line marks the fifth year in office). This means that politicians face substantial uncertainty on how long they could stay at a particular position.

```
fig_tenuretime <- df_merged %>%
    group_by(name) %>%
    filter(governor)  %>%
    ggplot(aes(x = term_length)) +
    geom_histogram(binwidth = 0.5) +
    coord_cartesian(xlim = c(0, 10)) +
    geom_vline(color = "red", xintercept = 5) +
    # Change x coordinates
    labs(x = "Term length of munipal/provincial leadership",
         y = "Count",
         title = "Histogram of municipal/provincial leadership term length",
         caption = "Zoomed in on 10 years")
fig_tenuretime
```

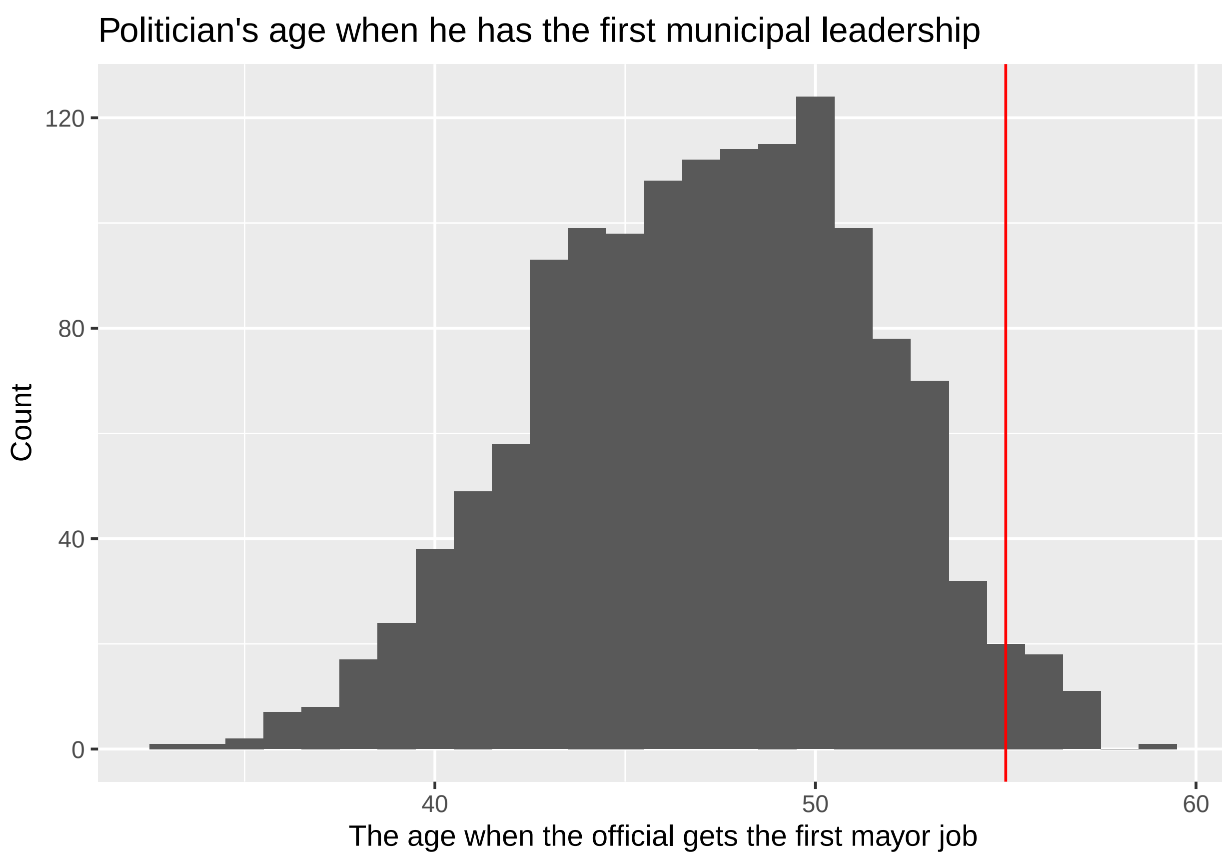



```
# ggsave("./FinalPJ/fig_tenuretime.png")
```

How much time does it take for one to become a municipal leader

Another factor that impacts Consider the age constraint

```r
fig_promotiontime <- df_merged %>%
    group_by(name) %>%
    filter(job == " 市长") %>%
    filter(age == min(age)) %>%
    ggplot(aes(x = age)) +
    geom_histogram(binwidth = 1) +
    geom_vline(color = "red", xintercept = 55) +
    labs(x = "The age when the official gets the first mayor job",
         y = "Count",
         title = "Politician's age when he has the first municipal leadership")
fig_promotiontime
```



Political cycles on spending

Given the age constraint on future promotions and substantial uncertainty on how long a politician can serve as a mayor or provincial governor, we would predict that in the starting years of their terms, politicians are incentivised to signal their competence to the upper-level organizational department for promotion. As a politician serves longer on a particular position and her age goes up, promotion incentives may decrease and fiscal necessity pushes the politician back to responsible behavior.

The time period for the data .

This difference in promotion incentives could be reflected in fiscal behaviors. Building on existing literature, I predict that politicians have two objectives: boost economic performance and maintain social stability (or prevent collective action). To boost economic performance, politicians may create tax exemptions or sell the state-owned land cheaply to developers for development; this would result in lower fiscal revenue. On the social stability side, politicians may opt to increase social expenditure to reduce social grievances and prevent collective actions.

Hypothesis 1: social speding is positively correlated with

Hypothesis 2:

Dependent variables

The key dependent variables are city-level social expenditure and fiscal revenue. I logged the dependent variables.

Independent variables

The key independent variables are `years in current office` and `age`.

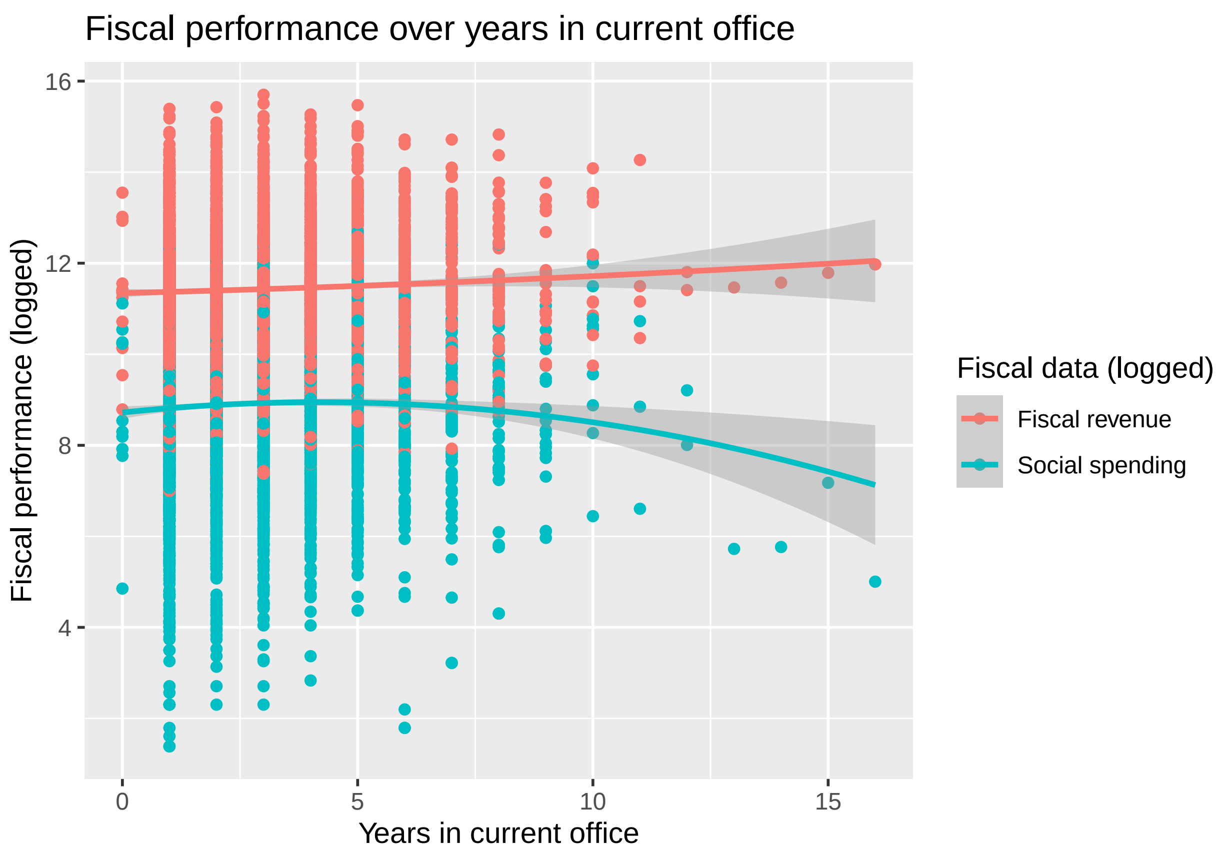
I also control for year fixed effects in the models to capture for the macro

To account for the effect of age on promotion incentives, I have also

Graph

The following graph shows

```r
# filter out who the mayors/leaders are
fig_fisc <- df_merged %>%
    filter(mayor) %>%
    pivot_longer(cols = c(social_exp_log, income_log),
                 names_to = "fiscal_type",
                 values_to = "values") %>%
    ggplot(aes(x = term_year, y = values, color = fiscal_type)) +
    geom_point() +
    geom_smooth(method = 'lm', formula = y ~ poly(x, 2)) +
    labs(x = "Years in current office",
         y = "Fiscal performance (logged)",
         title = "Fiscal performance over years in current office") +
    scale_color_discrete(name = "Fiscal data (logged)",
                         labels = c("Fiscal revenue", "Social spending"))
fig_fisc
```




```
# ggsave("./FinalPJ/fig_fisc.png")
```

Regression analysis

```
df_merged_mayor <- df_merged %>%
    filter(mayor)
# This set of regressions use social expenditure (logged) as DV
model_socexp <- lm(social_exp_log ~ term_year  + factor(year) + factor(name),
                   data = df_merged_mayor)
# Produce robust standard errors
modelr_socexp <- lm_robust(social_exp_log ~ term_year, fixed_effects = ~ year + name,
                            data = df_merged_mayor)
# Add a non-linear term
modelr_socexp2 <- lm_robust(social_exp_log ~ poly(term_year, 2), fixed_effects = ~ year + name,
                    data = df_merged_mayor)
# add an interaction with age
modelr_socexp_interage <- lm_robust(social_exp_log ~ age50*term_year, fixed_effects = ~ year + name,
                            data = df_merged_mayor)
mdlist_socexp <- list(model_socexp, modelr_socexp, modelr_socexp2, modelr_socexp_interage)
# Producing the regression tables
tbl_fe <- tibble("variable" = c("Leader fixed effects", "Year fixed effects"),
                 "values" = "Y") %>%
    cbind(replicate(.$values, n = length(mdlist_socexp) - 1))
```

	Model 1	Model 2	Model 3	Model 4
Years in office	−0.117***	−0.117***	−14.317***	−0.140***
	(0.012)	(0.018)	(2.389)	(0.021)
Years in office (squared)			0.641	
			(1.551)	
Age				−0.141**
				(0.052)
Age X Years in office				0.027+
				(0.014)
Num.Obs.	4425	4425	4425	4123
R2	0.942	0.942	0.942	0.942
R2 Adj.	0.917	0.917	0.917	0.918
AIC	7359.3			
BIC	15768.8			
Log.Lik.	−2364.655			
F	38.372			
Std.Errors		HC2	HC2	HC2
Leader fixed effects	Y	Y	Y	Y
Year fixed effects	Y	Y	Y	Y

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

```
modelsummary(mdlist_socexp,
             stars = TRUE,
             # output = "./FinalPJ/tbl_socexp.jpg",
             coef_map = c("term_year" = "Years in office",
                          "poly(term_year, 2)1" = "Years in office",
                          "poly(term_year, 2)2" = "Years in office (squared)",
                          "age50TRUE" = "Age",
                          "age50TRUE:term_year" = "Age X Years in office"),
             add_rows = tbl_fe)
```

```
## Warning: In version 0.8.0 of the `modelsummary` package, the default significance markers produced by
## This warning is displayed once per session.
```

Interpretation part: coef, standard errors, p-values

Model 1 is a linear model of regressing social expenditure on years in current office and year fixed effects. The coefficient on the `Years in office` variable is ‘

Model 2 adds a second-degree term of years in office to the regression.

Talk about the relative changes

Model 3 is similar to model 2 but is estimated with robust standard errors. Compared to model 2, the standard error

Model 4 adds an interaction term of years in office and age, and is estimated with robust standard errors. The coefficient on the `Years in office` variable

		2nd-degree polynomial	Robust standard errors	
Years in office	0.035***	4.352***	−16.747***	−0.256*
	(0.010)	(1.239)	(1.320)	(0.117)
Years in office (squared)		0.455	1.015**	
		(1.237)	(0.317)	
Age				0.064***
				(0.007)
Age X Years in office				0.005+
				(0.002)
Num.Obs.	5089	5089	5089	4683
R2	0.002	0.002	0.969	0.077
R2 Adj.	0.002	0.002	0.957	0.076
AIC	16556.3	16558.2		
BIC	16575.9	16584.3		
Log.Lik.	−8275.154	−8275.086		
F	12.336	6.235		
Std.Errors			HC2	HC2
Leader fixed effects	Y	Y	Y	Y
Year fixed effects	Y	Y	Y	Y

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

Income trends

```
# This regression is for mayors
model_inc <- lm(income_log ~ term_year, fixed_effects = ~ year + name,
                data = df_merged_mayor)
# Add non-linear terms in the
model_inc2 <- lm(income_log ~ poly(term_year, 2), fixed_effects = ~ year + name,
                 data = df_merged_mayor)
# Produce robust standard errors
modelr_inc2 <- lm_robust(income_log ~ poly(term_year, 2), fixed_effects = ~ year + name,
                         data = df_merged_mayor)
# add an interaction with age
model_inc_interage <- lm_robust(income_log ~ age*term_year,
                                data = df_merged_mayor)
mdlist_inc <- list(model_inc, "2nd-degree polynomial" = model_inc2, "Robust standard errors" = modelr_in
# Producing the regression tables
modelsummary(mdlist_inc,
             stars = TRUE,
             coef_map = c("term_year" = "Years in office",
                          "poly(term_year, 2)1" = "Years in office",
                          "poly(term_year, 2)2" = "Years in office (squared)",
                          "age" = "Age",
                          "age:term_year" = "Age X Years in office"),
             add_rows = tbl_fe)
```

Table 2 shows that when using fiscal revenue as the dependent variable.

	Model 1	Model 2
(Intercept)	1.467***	0.290***
	(0.090)	(0.087)
lag_socialexp	0.893***	
	(0.009)	
term_year	−0.029***	−0.004*
	(0.007)	(0.002)
lag_income		0.989***
		(0.007)
Num.Obs.	4001	4770
R2	0.800	0.958
R2 Adj.	0.800	0.958
Std.Errors	HC2	HC2

+ p < 0.1, * p < 0.05, ** p < 0.01, *** p < 0.001

Alternative specifications

In the following sections, I use different specifications for the regression. First I created a lagged dependent variable. ### Using lagged DV in the estimation

```
# Creating lagged DV in the dataset
df_lagged <- df_merged %>%
    group_by(admcode, year) %>%
    summarise(sumed_disciplined = any(disciplined == 1),
              sumed_exp_log = mean(exp_log),
              sumed_income_log = mean(income_log),
              sumed_social_exp_log = mean(social_exp_log)) %>%
    mutate(treatment = lag(sumed_disciplined),
           lag_exp = lag(sumed_exp_log),
           lag_income = lag(sumed_income_log),
           lag_socialexp = lag(sumed_social_exp_log)) %>%
    select(-sumed_disciplined, -sumed_exp_log) %>%
    right_join(df_merged, by = c('admcode' = 'admcode',
                                 'year' = 'year'))
```

```
## `summarise()` has grouped output by 'admcode'. You can override using the `.groups` argument.
```

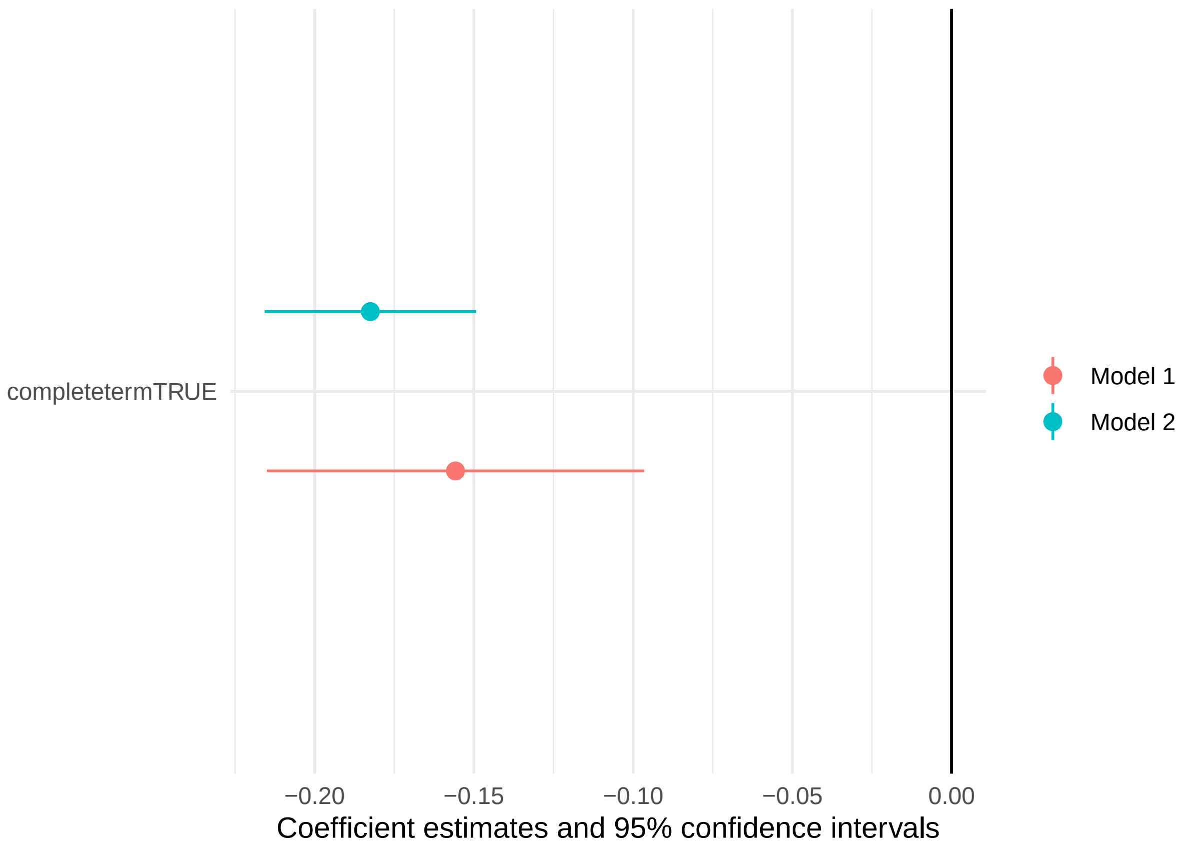
```
# Add a lagged DV to the regression
model_lag_socexp <- lm_robust(social_exp_log ~ lag_socialexp + term_year,
                              data = df_lagged %>% filter(mayor))

model_lag_income <- lm_robust(income_log ~ lag_income + term_year,
                              data = df_lagged %>% filter(mayor))
tbl_lag <- list(model_lag_socexp, model_lag_income)
modelsummary(tbl_lag, stars = TRUE)
```

Interpretation

Using indicator variables

```
# Using indicator variables
model_fac_socexp <- lm_robust(social_exp_log ~ completeterm, fixed_effects = ~ year + name,
                              data = df_merged %>% filter(mayor))
model_fac_income <- lm_robust(income_log ~ completeterm, fixed_effects = ~ year + name,
                              data = df_merged %>% filter(mayor))
mdlist_fac <- list(model_fac_socexp, model_fac_income)
modelplot(mdlist_fac) +
    geom_vline(xintercept = 0)
```



Bootstrap the standard errors