Replication of Examples in Chapter 8: Nonlinear Regression Models

Zheng Tian

[2016-05-30 Mon]

1 Introduction

This document shows how to estimate nonlinear regression models. We will reproduce the estimation results of the application of test scores in California elementary school districts in Section 8.4 in the textbook.

The regression concerns the effect of class sizes represented by student-teacher ratios on test scores. The goals of this regression are to answer the following research questions:

1. After controlling for differences in economic characteristics of different districts, does the effect on test scores of reducing the student-teacher ratio depend on the fraction of English learners?
2. Does this effect depend on the value of the student-teacher ratio?

2 Read data

We read the data file, which is `caschool.dta`.

```
library(AER)
# read data
library(foreign)
classdata <- read.dta("./data/caschool.dta")
```

The regression models in this application involve the dependent variable of test scores (*TestScore*), and various independent variables, including the student-teacher ratio (*STR*), the percentage of English learners (*PctEL*), the percentage of students eligible for subsidized lunch (*PctLch*),

and average district income (*Income*). The corresponding variables in `classdata` are `str`, `el_pct`, `meal_pct`, and `avginc`.

We first get basic descriptive statistics of these variables.

```
var2use <- c("testscr", "str", "el_pct", "meal_pct", "avginc")
df2use <- classdata[var2use]
sumdf <- summary(df2use)
```

Usually, we present these descriptive statistics in a table. We can use the function of `stargazer` to generate Table 1.

```
varlabs <- c(
    "Average test scores",
    "Student-teacher ratio",
    "Percent of English learners",
    "Percent of students eligible for subsidized lunch",
    "Average district income"
)

library(stargazer)
stargazer(df2use, title = "Descriptive Statistics of All Variables",
  covariate.labels = varlabs,
  summary.stat = c("max", "mean", "median", "min", "sd"),
  digits = 2,
  label = "tab:destab"
)
```

Table 1: Descriptive Statistics of All Variables

Statistic	Max	Mean	Median	Min	St. Dev.
Average test scores	706.75	654.16	654.45	605.55	19.05
Student-teacher ratio	25.80	19.64	19.72	14.00	1.89
Percent of English learners	85.54	15.77	8.78	0.00	18.29
Percent of students eligible for subsidized lunch	100.00	44.71	41.75	0.00	27.12
Average district income	55.33	15.32	13.73	5.34	7.23

Finally, we can generate a scatterplot to visualize the relationship between test scores and student-teacher ratios.

```
plot(testscr ~ str, data = df2use,
```

```
main = "The scatterplot of test scores against student-teacher ratios",
xlab = "Student-teacher ratio", ylab = "Test scores",
bty = "l", col = "blue")
```

Figure 1: The scatterplot of test scores and student-teacher ratios

3 Regression models

We estimate seven regression models that appear in Table 8.3 in Chapter 8 of the textbook.

Create a binary variable *HiEL*

Since some regressions in this table use the independent variable of *HiEL* that is a binary variable for the districts with the percentage of English learners higher than 10%, we first need to generate such variable.

```
hiel <- ifelse(df2use$el_pct > 10, TRUE, FALSE)
table(hiel)
```

FALSE	228
TRUE	192

We can see that there are 370 districts categorized as having high percentage of English learners.

Set up regression models

We need to set up seven regression models as follows:

```
fm1 <- testscr ~ str + el_pct + meal_pct
fm2 <- testscr ~ str + el_pct + meal_pct + log(avginc)
fm3 <- testscr ~ hiel*str
fm4 <- testscr ~ hiel*str + meal_pct + log(avginc)
fm5 <- testscr ~ str + I(str^2) + I(str^3) + hiel + meal_pct + log(avginc)
fm6 <- testscr ~ hiel*str + hiel*I(str^2) + hiel*I(str^3) + meal_pct + log(avginc)
fm7 <- testscr ~ str + I(str^2) + I(str^3) + el_pct + meal_pct + log(avginc)
```

Since they are all linear with respect to the parameters, we can estimate them using OLS with the function `lm()` one by one. However, writing the commands of `lm()` for each specification is repetitive work, which is more easily done with a loop. Of course, we can use `for` loop to do so, but R has a family of `apply()` functions that make running a loop much easier. Here we use the `lapply()` function.

`lapply()` takes its first argument as a `list` object, and the second argument as the name of a function that is imposed on each item in the `list` object, and the third or more arguments as the other inputs for that function. Before using `lapply`, we use the `mget()` function to generate a `list` object consisting of all formula specified above. Also, we define a function of `allols()` that wraps the `lm()` function, using the default data set of `df2use`. Finally, `lapply()` returns a `list` object consisting of all the `lm` objects estimated by the `lm()` function.

```
fm.ls <- mget(paste("fm", 1:7, sep = ""))
```

```
allols <- function(x) lm(x, data = df2use)
ols.all <- lapply(fm.ls, allols)
```

We present all estimation results in Table 2 using the `stargazer` function. Since this function reports the homoskedasticity-only standard errors of the coefficients by default, we need to replace them with the heteroskedasticity-robust standard errors.

```
coef.all <- lapply(ols.all, coef)
hccm.all <- lapply(ols.all, vcovHC, type = "HC1")
seht.all <- lapply(hccm.all, function(x) sqrt(diag(x)))

indep.labels <- c("$STR$", "$STR^2$", "$STR^3$",
  "Percent of English learner",
  "High Percent of English learner",
  "$HiEL \\times STR$", "$HiEL \\times STR^2$",
  "$HiEL \\times STR^3$", "Percent of eligible for free lunch",
  "Average district income")

stargazer(ols.all, title = "Nonlinear regression models of test scores",
  coef = coef.all, se = seht.all,
  covariate.labels = indep.labels,
  dep.var.caption = "Dependent variable: Average test scores",
  dep.var.labels.include = FALSE,
  no.space = TRUE, df = FALSE,
  order = c(2, 4, 5, 3, 1, 8, 9, 10, 6, 7, 11),
  float.env = "sidewaystable",
  label = "tab:tab83")
```

4 Discussion of the results

The research questions

Keep in mind that we have two research questions to answer:

1. Does this effect depend on the value of the student-teacher ratio?
2. After controlling for differences in economic characteristics of different districts, does the effect on test scores of reducing the student-teacher ratio depend on the fraction of English learners?

Table 2: Nonlinear regression models of test scores

	Dependent variable: Average test scores						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
STR	-0.998^{***}	-0.734^{***}	-0.968	-0.531	64.339^{***}	83.701^{***}	65.285^{***}
	(0.270)	(0.257)	(0.589)	(0.342)	(24.861)	(28.497)	(25.259)
STR^2					-3.424^{***}	-4.381^{***}	-3.466^{***}
					(1.250)	(1.441)	(1.271)
STR^3					0.059^{***}	0.075^{***}	0.060^{***}
					(0.021)	(0.024)	(0.021)
Percent of English learner	-0.122^{***}	-0.176^{***}					-0.166^{***}
	(0.033)	(0.034)					(0.034)
High Percent of English learner			5.639	5.498	-5.474^{***}	816.075^{**}	
			(19.515)	(9.795)	(1.034)	(327.674)	
$HiEL \times STR$			-1.277	-0.578		-123.282^{**}	
			(0.967)	(0.496)		(50.213)	
$HiEL \times STR^2$						6.121^{**}	
						(2.542)	
$HiEL \times STR^3$						-0.101^{**}	
						(0.043)	
Percent of eligible for free lunch	-0.547^{***}	-0.398^{***}		-0.411^{***}	-0.420^{***}	-0.418^{***}	-0.402^{***}
	(0.024)	(0.033)		(0.029)	(0.029)	(0.029)	(0.033)
Average district income		11.569^{***}		12.124^{***}	11.748^{***}	11.800^{***}	11.509^{***}
		(1.819)		(1.798)	(1.771)	(1.778)	(1.806)
Constant	700.150^{***}	658.552^{***}	682.246^{***}	653.666^{***}	252.051	122.354	244.809
	(5.568)	(8.642)	(11.868)	(9.869)	(163.634)	(185.519)	(165.722)
Observations	420	420	420	420	420	420	420
R^2	0.775	0.796	0.310	0.797	0.801	0.803	0.801
Adjusted R^2	0.773	0.794	0.305	0.795	0.798	0.799	0.798
Residual Std. Error	9.080	8.643	15.880	8.629	8.559	8.547	8.568
F Statistic	476.306^{***}	405.359^{***}	62.399^{***}	325.804^{***}	277.212^{***}	185.777^{***}	276.515^{***}

Note: *p<0.1; **p<0.05; ***p<0.01

The polynomial terms of STR

Regression models (5), (6), and (7) include the polynomial terms of STR. The t-statistics for all these coefficients on STR, STR^2, and STR^3 are greater than the critical value of the normal distribution at the 1% level. Thus, they are all statistically significant individually.

The joint zero hypotheses for STR^2 and STR^3 can be tested using the F-statistics for all the three regression models.

```
testSTR23 <- function(ols.res, vcov.hc){
    test <- linearHypothesis(ols.res, c("I(str^2) = 0", "I(str^3) = 0"),
     vcov. = vcov.hc)
    fstat <- test[2, 3]
    pval <- test[2, 4]
    return(list(Fstat = fstat, Pval = pval, Test = test))
}


F5 <- testSTR23(ols.all[[5]], hccm.all[[5]])
F6 <- testSTR23(ols.all[[6]], hccm.all[[6]])
F7 <- testSTR23(ols.all[[7]], hccm.all[[7]])
```

The F statistics and their corresponding p-values are

- Regression (5): The F-statistic is `6.17` with the p-value of `0.0023`;
- Regression (6): The F-statistic is `5.81` with the p-value of `0.0033`;
- Regression (7): The F-statistic is `5.96` with the p-value of `0.0028`.

Therefore, the joint zero hypotheses are rejected for all three models at the 1% level. That means that there is nonlinear effect of STR on test scores.

The simplest way to interpret the nonlinear effect of STR on test scores is by plotting the estimated regression lines. To generate the estimated regression lines, we need to get the predicted values of test scores based on a range of the values of STR, holding other variable constant. Except for STR, we let all the continuous variables take their average values and let $HiEL$ be one. STR takes the values from its minimum value to its maximum value, with a step of 0.05.

The following codes get the predicted values of test scores and draw the regression lines.

```
str.sim <- with(df2use, seq(min(str), max(str), by = 0.05))
n.sim <- length(str.sim)
```

```
means <- lapply(df2use, mean)
means$hiel <- ifelse(mean(hiel) > 0.5, TRUE, FALSE)
newdf <- data.frame(str = str.sim,
    el_pct = rep(means$el_pct, n.sim),
    meal_pct = rep(means$meal_pct, n.sim),
    avginc = rep(means$avginc, n.sim),
    hiel = rep(means$hiel, n.sim))

yhat.2 <- predict(ols.all[[2]], newdata = newdf)
yhat.5 <- predict(ols.all[[5]], newdata = newdf)
yhat.7 <- predict(ols.all[[7]], newdata = newdf)

plot(testscr ~ str, data = df2use,
     xlab = "Student-teacher ratio", ylab = "Test scores",
     bty = "l", col = "gray")
lines(yhat.2 ~ str.sim, col = "black", lwd = 1.5)
lines(yhat.5 ~ str.sim, col = "red", lwd = 1.5)
lines(yhat.7 ~ str.sim, col = "blue", lty = 2, lwd = 1.5)
legend("topright", c("Linear regression (2)",
     "Cubic regression (5)",
     "Cubic regression (7)"),
       col = c("black", "red", "blue"),
       lty = c(1, 1, 2))
```

The interaction between *STR* and *HiEL*

Regression models (3), (4), and (6) include the interaction terms of STR and $HiEL$. Regressions (3) and (4) only have the interaction term of $HiEL$ and STR, which is neither significant at the 10% level, while Regression (6) has the interaction terms of $HiEL$ and STR, STR^2, and STR^3, which are individually significant at the 5% level. The joint hypothesis of all the interaction terms having zero coefficients can be tested as follows

```
F6.inter <- linearHypothesis(ols.all[[6]],
     c("hielTRUE:str=0", "hielTRUE:I(str^2)=0",
       "hielTRUE:I(str^3)=0"), vcov. = hccm.all[[6]])
```

The F-statistic is `2.69` with the p-value of `0.046` so that the coefficients on the three interaction terms are jointly significant at the 5% level but insignificant at the 1% level.

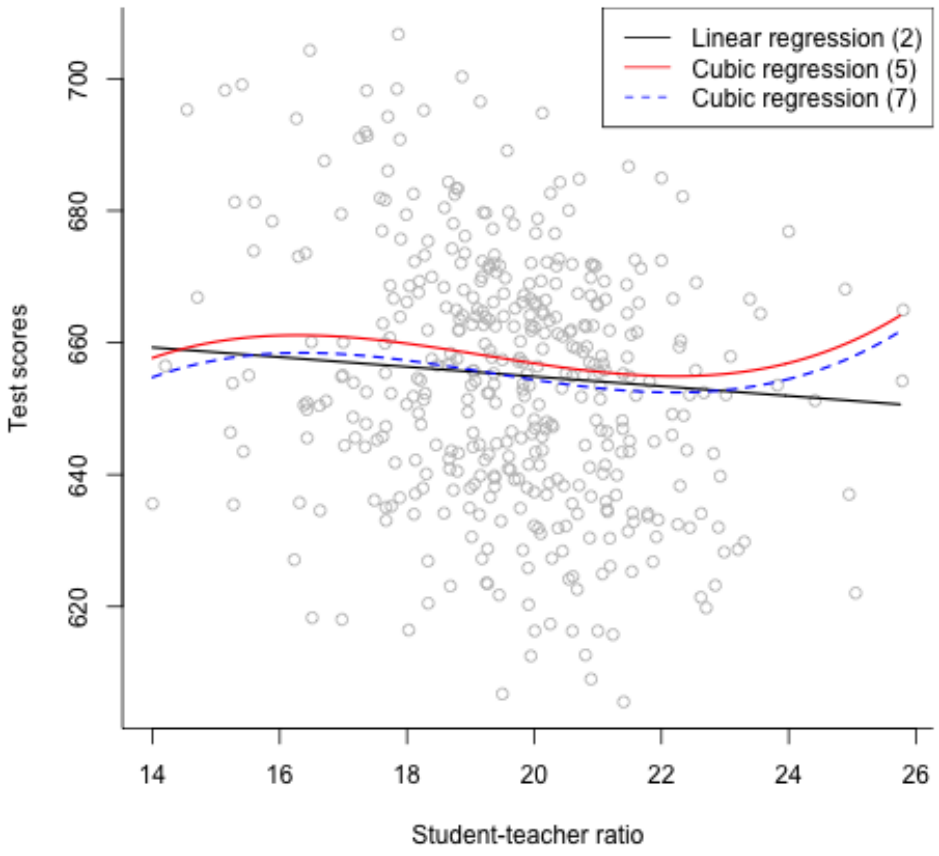

Figure 2: Three regression lines relating test scores and student-teacher ratios

Also, we can plot the regression lines in Regression (6) for $HiEL = 1$ and $HiEL = 0$.

```
# plot Figure 8.11
df2use.a <- df2use[hiel, ]
df2use.b <- df2use[!hiel, ]

newdf$hiel <- TRUE
yhat.6.T <- predict(ols.all[[6]], newdata = newdf)

newdf$hiel <- FALSE
yhat.6.F <- predict(ols.all[[6]], newdata = newdf)

plot(testscr ~ str, data = df2use.a,
     xlab = "Student-teacher ratio", ylab = "Test scores",
     bty = "l", col = "gray")
points(testscr ~ str, data = df2use.b, col = "orange")
lines(yhat.6.T ~ str.sim, col = "blue", lwd = 1.5)
lines(yhat.6.F ~ str.sim, col = "red", lwd = 1.5, lty = 2)
legend("topright", legend = c("High EL", "Low EL"),
       pch = c(1, 1), col = c("gray", "orange"))
legend(18, 615, legend = c("Regression with HiEL=1",
 "Regression with HiEL = 0"),
       col = c("blue", "red"), lty = c(1, 2))
```

Conclusion

5 Appendix: R codes

```
# read the data files into R
# read the dta file
library(AER)

library(foreign)
classdata <- read.dta("./data/caschool.dta")

# extract variables used in regression models
var2use <- c("testscr", "str", "el_pct", "meal_pct", "avginc")
```

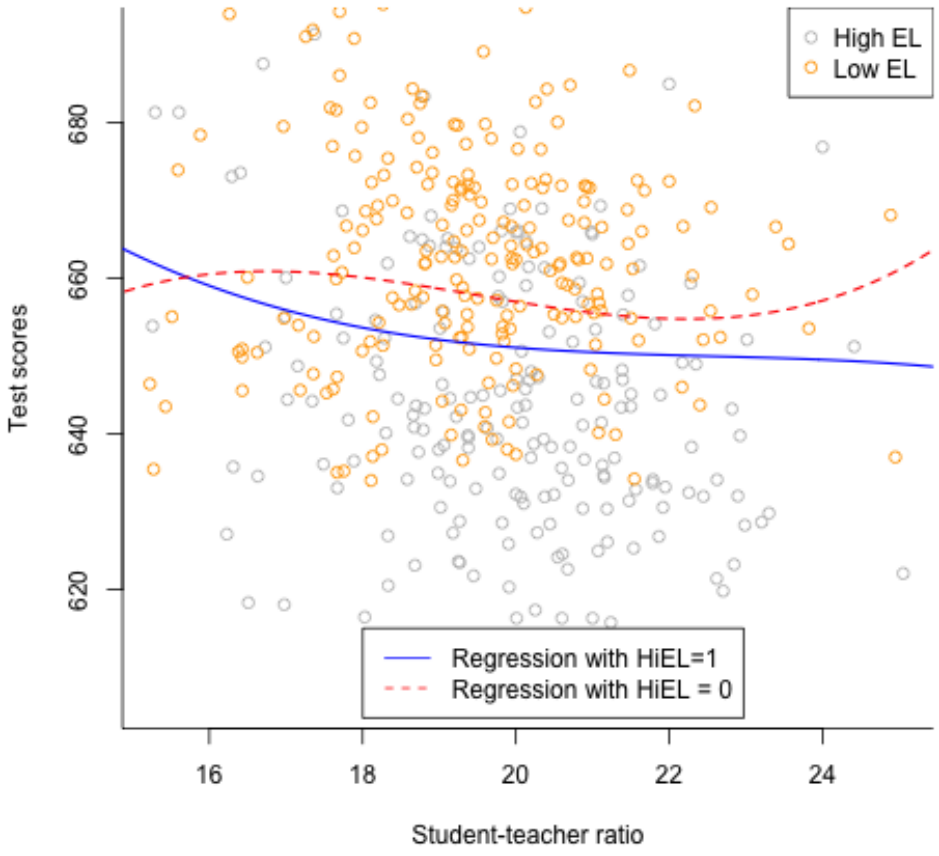

Figure 3: The regression lines for districts with high and low percentage of English learners

```
df2use <- classdata[var2use]

## descriptive statistics
sumdf <- summary(df2use)

varlabs <- c(
    "Average test scores",
    "Student-teacher ratio",
    "Percent of English learners",
    "Percent eligible for subsidized lunch",
    "Average district income"
)

library(stargazer)
stargazer(df2use, title = "Descriptive Statistics of All Variables",
          covariate.labels = varlabs,
          summary.stat = c("max", "mean", "median", "min", "sd"),
          digits = 2,
          label = "tab:destab"
)

plot(testscr ~ str, data = df2use,
     main = "The scatterplot of test scores against student-teacher ratios",
     xlab = "Student-teacher ratio", ylab = "Test scores",
     bty = "l", col = "blue")

## regressions

hiel <- ifelse(df2use$el_pct > 10, TRUE, FALSE)
table(hiel)

fm1 <- testscr ~ str + el_pct + meal_pct
fm2 <- testscr ~ str + el_pct + meal_pct + log(avginc)
fm3 <- testscr ~ str + hiel + hiel:str
fm4 <- testscr ~ str + hiel + hiel:str + meal_pct + log(avginc)
fm5 <- testscr ~ str + I(str^2) + I(str^3) + hiel + meal_pct + log(avginc)
fm6 <- testscr ~ hiel*str + hiel*I(str^2) + hiel*I(str^3) + meal_pct + log(avginc)
fm7 <- testscr ~ str + I(str^2) + I(str^3) + el_pct + meal_pct + log(avginc)
```

```
fm.ls <- mget(paste("fm", 1:7, sep = ""))

allols <- function(x) lm(x, data = df2use)
ols.all <- lapply(fm.ls, allols)

coef.all <- lapply(ols.all, coef)
hccm.all <- lapply(ols.all, vcovHC, type = "HC1")
seht.all <- lapply(hccm.all, function(x) sqrt(diag(x)))

indep.labels <- c("$STR$", "$STR^2$", "$STR^3$",
                  "Percent of English learner",
                  "High Percent of English learner",
                  "$HiEL \\times STR$", "$HiEL \\times STR^2$",
                  "$HiEL \\times STR^3$", "Percent of eligible for free lunch",
                  "Average district income")

stargazer(ols.all, title = "Nonlinear regression models of test scores",
          coef = coef.all, se = seht.all,
          covariate.labels = indep.labels,
          dep.var.caption = "Dependent variable: Average test scores",
          dep.var.labels.include = FALSE,
          no.space = TRUE, df = FALSE,
          order = c(2, 4, 5, 3, 1, 8, 9, 10, 6, 7, 11),
          label = "tab:tab83")


# Hypothesis tests
# STR^2 and STR^3

testSTR23 <- function(ols.res, vcov.hc){
    test <- linearHypothesis(ols.res, c("I(str^2) = 0", "I(str^3) = 0"),
                             vcov. = vcov.hc)
    fstat <- test[2, 3]
    pval <- test[2, 4]
    return(list(Fstat = fstat, Pval = pval, Test = test))
}
```

```
F5 <- testSTR23(ols.all[[5]], hccm.all[[5]])
F6 <- testSTR23(ols.all[[6]], hccm.all[[6]])
F7 <- testSTR23(ols.all[[7]], hccm.all[[7]])

# plotting
# Plot Figure 8.10
str.sim <- with(df2use, seq(min(str), max(str), by = 0.05))
n.sim <- length(str.sim)
means <- lapply(df2use, mean)
means$hiel <- ifelse(mean(hiel) > 0.5, TRUE, FALSE)
newdf <- data.frame(str = str.sim,
                    el_pct = rep(means$el_pct, n.sim),
                    meal_pct = rep(means$meal_pct, n.sim),
                    avginc = rep(means$avginc, n.sim),
                    hiel = rep(means$hiel, n.sim))

yhat.2 <- predict(ols.all[[2]], newdata = newdf)
yhat.5 <- predict(ols.all[[5]], newdata = newdf)
yhat.7 <- predict(ols.all[[7]], newdata = newdf)

F6.inter <- linearHypothesis(ols.all[[6]],
                             c("hielTRUE:str=0", "hielTRUE:I(str^2)=0",
                               "hielTRUE:I(str^3)=0"), vcov. = hccm.all[[6]])


plot(testscr ~ str, data = df2use,
     xlab = "Student-teacher ratio", ylab = "Test scores",
     bty = "l", col = "gray")
lines(yhat.2 ~ str.sim, col = "black", lwd = 1.5)
lines(yhat.5 ~ str.sim, col = "red", lwd = 1.5)
lines(yhat.7 ~ str.sim, col = "blue", lty = 2, lwd = 1.5)
legend("topright", c("Linear regression (2)",
                     "Cubic regression (5)",
                     "Cubic regression (7)"),
       col = c("black", "red", "blue"),
       lty = c(1, 1, 2))

# plot Figure 8.11
```

```
df2use.a <- df2use[hiel, ]
df2use.b <- df2use[!hiel, ]

newdf$hiel <- TRUE
yhat.6.T <- predict(ols.all[[6]], newdata = newdf)

newdf$hiel <- FALSE
yhat.6.F <- predict(ols.all[[6]], newdata = newdf)

plot(testscr ~ str, data = df2use.a,
     xlab = "Student-teacher ratio", ylab = "Test scores",
     bty = "l", col = "gray")
points(testscr ~ str, data = df2use.b, col = "orange")
lines(yhat.6.T ~ str.sim, col = "blue", lwd = 1.5)
lines(yhat.6.F ~ str.sim, col = "red", lwd = 1.5, lty = 2)
legend("topright", legend = c("High EL", "Low EL"),
       pch = c(1, 1), col = c("gray", "orange"))
legend(18, 615, legend = c("Regression with HiEL=1",
                           "Regression with HiEL = 0"),
       col = c("blue", "red"), lty = c(1, 2))
```