Homework 2: . . . But We Make It Up in Volume

36-402, Spring 2016

Due at 11:59 pm on Wednesday, 27 January 2016

"Gross domestic product" is a standard measure of the size of an economy; it's the total value of all goods and services bought and solid in a country over the course of a year. It's not a perfect measure of prosperity[1], but it is a very common one, and many important questions in economics turn on what leads GDP to grow faster or slower.

One common idea is that poorer economies, those with lower initial GDPs, should grower faster than richer ones. The reasoning behind this "catching up" is that poor economies can copy technologies and procedures from richer ones, but already-developed countries can only grow as technology advances. A second, separate idea is that countries can boost their growth rate by under-valuing their currency, making the goods and services they export cheaper.

This week's data set contains the following variables:

- Country, in a three-letter code (see `http://en.wikipedia.org/wiki/ISO_3166-1_alpha-3`).
- Year (in five-year increments).
- Per-capita GDP, in dollars per person per year ("real" or inflation-adjusted).
- Average percentage growth rate in GDP over the next five years.
- An index of currency under-valuation[2]. The index is `0` if the currency is neither over- nor under- valued, positive if under-valued, negative if it is over-valued.

Note that not all countries have data for all years. However, there are no missing values in the data table.

[1] A standard example: if vandals break all the windows on a street, a town, GDP goes *up* by the cost of the repairs.

[2] The idea is to compare the actual exchange rate with the US dollar to what's implied by the prices of internationally traded goods in that country — the exchange rate which would ensure "purchasing power parity". The details are in the paper this assignment is based on, which will be revealed in the solutions.

1. (5) Linearly regress the growth rate on the under-valuation index and the log of GDP. Report the coefficients and their standard errors. Do the coefficients support the idea of "catching up"? Do they support the idea that under-valuing a currency boosts economic growth?

2. Repeat the linear regression but add as covariates the country, and the year. Use `factor(year)`, not `year`, in the regression formula.

   (a) (1) Report the coefficients for log GDP and undervaluation, and their standard errors.

   (b) (2) Explain why it is more appropriate to use `factor(year)` in the formula than just `year`.

   (c) (2) Plot the coefficients on year versus time.

   (d) (5) Does this expanded model support the idea of catching up? Of undervaluation boosting growth?

3. Does adding in year and country as covariates improve the predictive ability of a linear model which includes log GDP and under-valuation?

   (a) (1) What are the R^2 and the adjusted R^2 of the two models?

   (b) (5) Use leave-one-out cross-validation to find the mean squared errors of the two models. Which one actually predicts better, and by how much? *Hint:* Use the code from lecture 3.

   (c) (4) Explain why using 5-fold cross-validation would be hard here. (You don't need to figure out how to do it.)

4. *Kernel smoothing* Use kernel regression, as implemented in the `np` package, to non-parametrically regress growth on log GDP, under-valuation, country, and year (treating year as a categorical variable). *Hint:* read chapter four carefully. In particular, try setting `tol` to about 10^{-3} and `ftol` to about 10^{-4} in the `npreg` command, and allow several minutes for it to run. (We suggest caching this part of your code.)

   (a) (4) Give the coefficients of the kernel regression, or explain why you can't.

   (b) (2) Plot the predicted values of the kernel regression, for each country and year, against the predicted values of the linear model.

   (c) (4) Plot the residuals of the kernel regression against its predicted values. Should these points be scattered around a flat line, if the model is right? Are they?

   (d) (4) The `npreg` function reports a cross-validated estimate of the mean squared error for the model it fits. What is that? Does the kernel regression predict better or worse than the linear model with the same variables?

5. *Time courses and interactions* In this question, use the kernel regression you fit in the previous problem.

   (a) (5) Plot the predicted growth rate, as a function of the year, in five year increments from 1955 to 2000, if the initial GDP (not log GDP!) is \$10,000 in each period, the under-valuation index is 0 (i.e., no under- or over- valuation), and the country is Turkey.

   (b) (1) Re-do the plot but change the under-valuation index to +0.5.

   (c) (1) Re-do the plot but hold the initial GDP at \$1,000 and the under-valuation index at 0.

   (d) (1) Re-do the plot with the initial GDP at \$1,000 and the under-valuation index at +0.5.

   (e) (5) Is there evidence of an interaction between initial GDP and under-valuation? Explain.

   (f) (5) For the same kernel regression, plot the predictions for growth against each variable, holding the other variables fixed at their medians. You can obtain these plots using the `plot` command and the kernel regression object from `npreg`. Is there evidence that $\log(GDP)$ or under-valuation are strongly related to growth? Which variables do show a strong relationship with growth?

6. (15) Chapter 3, Exercise 4. (5 for each part.)

7. *Kernel regression and varying smoothness.* Starter code for this problem is in `hw2prob7.R` on the course website. That code will generate a data set to be used for this problem, and will also provide a true mean function `mu(x)`. The resulting data frame has a `x` column (your predictor) and a `y` column (your response).

   (a) (3) Plot y versus x. Overlay the true mean function `mu(x)` using the `curve` function in `R`. What do you notice for $x < 4\pi$ and $x > 4\pi$?

   (b) (5) Using the `np` library in `R`, fit a kernel regression on each of the following datasets:

      i. Only those data points with $x < 4\pi$.
      ii. Only those data points with $x > 4\pi$.
      iii. All the data points

      For each of these regressions, what is the optimal bandwidth? How does the optimal bandwidth for the overall data set compare to the optimal bandwidth for each of the halves?

   (c) (5) For each of the three selected bandwidths, make a plot showing:

      - The true mean $\mu(x)$.
      - The data points.
      - The kernel regression predictions, with the bandwidth specified to be the selected bandwidth

The result should be three plots, each tuned to one of the selected bandwidths. Give these plots clear titles to distinguish them. *Hint:* to see the important details of your plot, you will probably need to make them reasonably large. Use the zoom button in the plotting frame in RStudio to see a bigger plot. To include a bigger plot in your homework, use the `fig.width` and `fig.height` settings of Rmarkdown. An example code chunk would look like:

````
```{r, fig.width=7, fig.height=9}
  CODE
```
````

(You will probably want to change the numbers 7 and 9.)

(d) (5) How do these three plots differ? In particular, how well do the regressions trained on the left and right halves do on each half of the data set? How well does the bandwidth fit on the overall data set do on each half? (Be specific about the types of problems that occur.) What lesson might this tell about functions of varying smoothness and kernel regression, if any?

RUBRIC (10): The text is laid out cleanly, with clear divisions between problems and sub-problems. The writing itself is well-organized, free of grammatical and other mechanical errors, and easy to follow. Questions which ask for a plot or table are answered with both the figure itself and the command (or commands) use to make the plot. Plots are carefully labeled, with informative and legible titles, axis labels, and (if called for) sub-titles and legends; they are placed near the text of the corresponding problem. All quantitative and mathematical claims are supported by appropriate derivations, included in the text, or calculations in code. Numerical results are reported to appropriate precision. Code is properly integrated with a tool like R Markdown or knitr, and both the knitted file and the source file are submitted. The code is indented, commented, and uses meaningful names. All code is relevant to the text; there are no dangling or useless commands. All parts of all problems are answered with actual coherent sentences, and never with raw computer code or its output.