Homework 7

36-467, Fall 2018

Due at 6 pm on Wednesday, 24 October 2018

AGENDA: Familiarization with autoregressive models, and with (part of) the data for the final.

The file `sial.csv` contains part of the data described by Prof. Manning in his guest lecture of October 9, which will be used in your take-home final. This records historical estimates for the number of slaves exported, per decade, from various regions of Africa, for every decade from 1650 to 1890. (Some of the entries are "100" and "10"; these indicate "an unknown but small (or very small) number".) Column names have the format "Territory - Region", e.g., "Mauritania - Senegambia" indicates data for the territory of Mauritania in the region of Senegambia.

1. *Descriptive statistics*

   (a) (3) Create a plot showing all the time series together. (There are 64 columns, so it won't be practical to give each one a label in a legend on the graph.)

   (b) (3) Create a plot showing the time series for the territories of Southern Dahomey, Sierra Leone, Akan, and Western Nigeria. Include, for each time series, a spline smooth of the trend.

   (c) (5) Plot the autocorrelation functions for those four time series.

   (d) (5) Plot the cross-correlation functions among those four time series. (There will be six of them.)

2. *Autoregressive modeling of one time series* In this problem, "the time series" refers to the series for South Dahomey, one of the four which you worked with in Problems 1b–1d.

   (a) (5) Fit an AR(1) model to the time series using ordinary least squares. (There are several functions for doing this in base R, using `ar`, `arima0`, `arima`, etc., as well as just using `lm`.) What are the intercept and slope?

   (b) (5) What stationary mean is implied by your parameter estimate from the previous part? What stationary variance is implied? How well do these match the actual mean and variance of this time series?

(c) (5) Simulate 20 new trajectories from your estimate AR(1) model, all starting from the same initial value as the actual data. (You can do this with your own code, as in the slides for lecture 12, or use a function like `arima.sim`.) Plot these together with the data. (Make sure that the plot clearly distinguishes data from simulation.) Does the data look like a typical run of the simulation? If not, how does it differ?

(d) (5) Fit an AR(2) model to the time series using ordinary least squares. What are the coefficients?

(e) (5) Re-fit the AR(1) and AR(2) models to the data < 1780. Then, for each decade in $[1780, 1890]$, calculate a prediction using the fixed model coefficients and the values of the time series up to that decade (so the prediction for 1780 uses all data ≤ 1770, the prediction for 1790 uses all data ≤ 1780, etc.). Report the root-mean-squared errors. Which model seems to be doing better?

3. *Vector autoregressive modeling of multiple time series* In this problem, "the time series" refers to the set of four series which you worked with in Problems 1b–1d.

(a) (5) Use ordinary least squares to fit a VAR(1) model to these time series. Report all the coefficients — both a vector of 4 intercepts and a 4×4 matrix of slopes. (Note that if you use the `ar` function, it may fit an intercept *as well as* de-meaning — see `help(ar)` and especially `help(ar.ols)`.)

(b) (5) What are the eigenvalues of the estimated slope matrix? What do they imply about the dynamics of the process?

(c) (5) Explain, in words, the meaning of the coefficients used to predict South Dahomey.

(d) (5) Suppose one extra slave was exported from Sierra Leone in the decade beginning 1700. What would be the change in the expected number of slaves exported from Akan in the decade beginning 1710? What would be the change in the expected number of slaves exported from Sierra Leone in the decade beginning 1720?

4. *Many autoregressive models*

(a) (4) Fit an AR(1) model, by ordinary least squares, to each of the 64 time series for the 64 territories. Report the estimated coefficients in a suitable figure.

(b) (4) How many of the estimated models *could* be stationary? How many of the estimated models *could not possibly* be stationary?

5. *Detrending with PCA*

(a) (3) Do PCA on these data, treating them as 25 observations of a 64-dimensional vector. How much of the variance does the first principal component capture?

(b) (3) Plot PC1 scores as a function of time, and describe its shape in words.

(c) (5) Let $X_i(t)$ be the time series for territory i, $P_1(t)$ be the PC1 scores as a function of time, and w_{i1} be the weight of territory i in PC1. Now define

$$Y_i(t) \equiv X_i(t) - w_{i1} P_1(t)$$

Explain why $Y_i(t)$ can be seen as a de-trended version of $X_i(t)$.

(d) (5) Create a new data-frame consisting of the 64 $Y_i(t)$ time-series, and plot the series for the four territories used in Problem 1b. Compare this plot, in words, to the one from Problem 1b. Are the new series more plausibly stationary than the old ones, less plausibly stationary, or about the same?

(e) (5) Estimate an AR(1) model for the new, de-trended time series for South Dahomey. Generate 20 new trajectories from this model, as in Problem 2c, and, as in that problem, plot the simulated trajectories along the real time series. Do the simulations match the data better after de-trending, or worse, or about the same? Explain.

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, without dangling or useless commands. All parts of all problems are answered with coherent sentences, and raw computer code or output are only shown when explicitly asked for.