AAE 637 Lab 5: Intermediate MATLAB programming*

2/25/2015

Advanced control flow

- `switch` - generalization of `if` so more than two choices can be specified
- `continue` - start back at the top of the loop without executing any instructions remaining in the loop
- `break` - exit the loop
- `return` - Similar to `break`, but used within a function. Stops the function from making more computations and then yields the current values of the output
- `&&` and `||` vs. `&` and `|` - the double symbol means only evaluate a statement in the "chain" if it is needed to assess the truth value of the whole statement. Don't use the double symbol in matrix subsetting.

Error handling

- `assert` - produce an error if a specified condition is false
- `error` - produce an error
- `try` & `catch` - redirect your program when an error is encountered rather than halting the program

Understanding the call stack

	Call Stack			
	Function name	Purpose	Arguments	Global variables
|	`NR`	Estimate a nonlinear least squares model	various	various
|	`hessian_bwg`	Compute the hessian of the SSE function	function name; betas	none
|	`sse_fn`	Compute the SSE of the desired model	betas	`lhsvar`; `func_name`
↓	`cobb_douglas`	Compute the predicted values for the model	betas	`rhsvar`

*prepared by Travis McArthur, UW-Madison (http://www.aae.wisc.edu/tdmcarthur/teaching.asp)

Strategies for restricting parameter values during estimation

You will want to use a monotonic transformation that maps the domain $\mathbb{R}$ into a restricted range that is a subset of $\mathbb{R}$.

- Say your restriction has this form: $a < \beta < \infty$. Let $\dot{\beta}$ be the value that you input into your estimation function.
  - One solution: $\beta = h(\dot{\beta}) = \exp\left\{\dot{\beta}\right\} + a$
  - And $-\infty < \beta < b$ works similarly: $\beta = h(\dot{\beta}) = -\exp\left\{\dot{\beta}\right\} + b$
- Your restriction has this form: $a < \beta < b$
  - Use the inverse tangent function
  - $\beta = h(\dot{\beta}) = \arctan(\dot{\beta}) \cdot \dfrac{(b-a)}{\pi} + \dfrac{(b+a)}{2}$

Warning: your usual estimation procedure will yield a table with an estimate of $\dot{\beta}$, not β. You will have to fix this somehow.

This is not the same as testing a restriction, since that involves restriction of a parameter or functions of parameters to a single value. The technique above is used for avoiding invalid parameter values.

Miscellaneous

- Use `rng` to set the "seed" for the random number generator so that a random sequence is reproducible
- Use `strcmp` and/or `pull_data` to select matrix columns by name
- Use wolframalpha.com to double-check your analytical derivatives

Adjustments to reach convergence

- Selecting good starting values is important: use OLS, restricted model, randomized starting values, or grid search
- Alter or remove second condition here:
  `while (ss1 < ss2) && (s >=.2); % ** Loop to determine step length`
- Scale down variables
- Set `dh` lower, say to 10^{-10}. This affects the precision of numerical calculus of `Grad`, but not `hessian_bwg`
- Try Gauss-Newton when Newton-Raphson fails, and vice versa.
- Make sure you are feeding a column beta vector to your optimization function