Problem Set 2 solutions

June 7, 2012

1. Practice with Matlab's functional optimizers using `nlsw88.mat` (from PS1)

   (a) Estimate the following model:

   $$\ln(wage_i) = \beta_1 + \beta_2 age_i + \beta_3 black_i + \beta_4 other_i + \beta_5 collgrad_i + \varepsilon_i \tag{1}$$

   under the assumption that ε_i is mean-zero and well-behaved. Note that $black_i$ corresponds to $race_i = 2$ and $other_i$ corresponds to $race_i = 3$. *Be sure to drop observations for all variables where any of the variables are missing.*

   i. see m-file code
   ii. see m-file code
   iii. see m-file code
   iv.

Variable	$\hat{\beta}_{fminsearch}$	$\hat{\beta}_{fminunc}$	$\hat{\beta}_{fmincon}$	$\hat{\beta}_{closed\_form}$
Constant	1.998	2.000	1.998	1.998
age	-0.005	-0.005	-0.005	-0.005
black	-0.134	-0.134	-0.134	-0.134
other	0.022	0.023	0.022	0.022
collgrad	0.422	0.422	0.422	0.422
s^2	0.293	0.293	0.293	0.293
SSE	656.578	656.578	656.578	656.578
N		2,244		

   Answers are very similar regardless of which optimizer is used.

   (b) Now estimate the same model from (a), but assuming $\varepsilon_i \overset{iid}{\sim} N(0,\sigma)$. In this case, the log likelihood function looks like

   $$\ell(X_i;\beta,\sigma) = \sum_{i=1}^{n}\left\{-\frac{1}{2}\ln\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2}\left(\ln(wage_i) - X_i\beta\right)^2\right\} \tag{2}$$

   where $X_i\beta$ is the right-hand side of equation (1) (except ε, of course).

   i. See m-file code
   ii. See m-file code

iii. See m-file code

iv.

Variable	$\hat{\beta}_{fminsearch}$	$\hat{\beta}_{fminunc}$	$\hat{\beta}_{fmincon}$	$\hat{\beta}_{closed\_form}$
Constant	1.998	1.999	1.998	1.998
age	-0.005	-0.005	-0.005	-0.005
black	-0.134	-0.134	-0.134	-0.134
other	0.022	0.027	0.022	0.022
collgrad	0.422	0.422	0.422	0.422
$\hat{\sigma}^2$	0.293	0.293	25.047	0.293
SSE or log likelihood	-1,805.189	-1,805.190	-5,688.904	656.578
N		2,244		

The normality assumption doesn't do much in terms of point estimates (even the error variance). `fminunc` and `fminsearch` were very similar, but `fmincon` had a hard time estimating the error variance. s^2 and $\hat{\sigma}^2$ are identical.

(c) Now estimate the following model:

$$\begin{aligned}\ln(wage_i) = {} & \beta_1 + \beta_2 age_i + \beta_3 black_i + \beta_4 other_i + \beta_5 collgrad_i + \\ & \beta_6 grade_i + \beta_7 married_i + \beta_8 south_i + \beta_9 c\_city_i + \\ & \beta_{10} union_i + \beta_{11} ttl\_exp_i + \beta_{12} tenure_i + \beta_{13} age_i^2 + \\ & \beta_{14} hours_i + \beta_{15} never\_married_i + \varepsilon_i\end{aligned} \quad (3)$$

i. See m-file code

ii. See m-file code

iii. See m-file code

iv. See m-file code

v.

	OLS		MLE		OLS
Variable	$\hat{\beta}_{fminsearch}$	$\hat{\beta}_{fminunc}$	$\hat{\beta}_{fminsearch}$	$\hat{\beta}_{fminunc}$	$\hat{\beta}_{closed\_form}$
Constant	0.578	0.489	0.368	0.611	0.629
age	0.000	0.002	0.000	0.001	0.000
black	-0.106	-0.058	-0.167	-0.108	-0.108
other	0.038	0.014	0.040	0.031	0.031
collgrad	0.012	0.014	0.019	0.044	0.044
grade	0.069	0.074	0.072	0.065	0.065
married	-0.032	-0.032	-0.036	-0.035	-0.035
south	-0.133	-0.118	-0.199	-0.135	-0.135
c_city	0.085	0.092	0.119	0.085	0.085
union	0.124	0.063	0.196	0.123	0.123
ttl_exp	0.033	0.031	0.034	0.033	0.033
tenure	0.009	0.010	0.004	0.009	0.009
age^2	0.000	0.000	0.000	0.000	0.000
hours	0.003	0.003	0.006	0.003	0.003
never_married	-0.102	-0.076	-0.044	-0.107	-0.107
s^2 $\left(\hat{\sigma}^2\right)$	0.163	0.164	0.167	0.162	0.163
SSE or log likelihood	301.477	304.098	-977.320	-946.618	301.352
N			1,865		

`fminsearch` outperformed `fminunc` under OLS, but the opposite was true for MLE (you can tell primarily by looking at the SSE or log likelihood value. A look at the parameter estimates reinforces this conclusion).

vi.

	OLS		MLE		OLS
Variable	$\hat{\beta}_{fminsearch}$	$\hat{\beta}_{fminunc}$	$\hat{\beta}_{fminsearch}$	$\hat{\beta}_{fminunc}$	$\hat{\beta}_{closed\_form}$
Constant	0.054	0.011	0.083	0.011	0.629
age	0.001	0.027	0.064	0.032	0.000
black	-0.104	-0.086	-0.265	0.009	-0.108
other	0.013	0.016	-0.322	0.010	0.031
collgrad	0.009	0.014	-0.074	0.013	0.044
grade	0.040	0.069	0.016	0.047	0.065
married	0.069	0.005	-0.101	0.011	-0.035
south	-0.050	-0.156	0.190	0.008	-0.135
c_city	-0.027	0.047	0.295	0.011	0.085
union	0.082	0.107	0.496	0.012	0.123
ttl_exp	0.055	0.033	0.052	0.044	0.033
tenure	0.000	0.009	0.006	0.032	0.009
age^2	0.000	0.000	-0.001	-0.001	0.000
hours	0.007	0.003	-0.001	0.039	0.003
never_married	-0.025	-0.022	-0.361	0.010	-0.107
s^2 $\left(\hat{\sigma}^2\right)$	0.190	0.164	0.176	0.087	0.163
SSE or log likelihood	351.806	303.333	-1,390.852	-3,733.363	301.352
N			1,865		

When we start the optimization at .01 for all parameters, `fminunc` now outperforms `fminsearch` for OLS, but `fminsearch` outperforms `fminunc` for MLE. For the MLE, however neither optimizer was close at all. I'm not sure why this is the case. It could be an issue with the data.

2. Practice with Matlab's functional optimizers using `nhanes2d.mat` (data from the National Health and Nutritional Examination Survey—NHANES).

   (a) Estimate the following model:

$$
\begin{aligned}
hct_i \quad = \quad & \beta_1 + \beta_2 age_i + \beta_3 black_i + \beta_4 other_i + \beta_5 heartatk_i + \\
& \beta_6 female_i + \beta_7 highbp_i + \beta_8 northeast_i + \beta_9 midwest_i + \\
& \beta_{10} south_i + \beta_{11} non\_central\_city_i + \beta_{12} rural_i + \beta_{13} height_i + \\
& \beta_{14} weight_i + \beta_{15} houssiz_i + \varepsilon_i
\end{aligned} \tag{4}
$$

   Be sure to drop observations for all variables where any of the variables are missing. Also, report the sum of squared residuals and/or log likelihood at convergence, number of iterations to convergence, and the estimation sample size.

      i. See m-file code
      ii. See m-file code
      iii. See m-file code
      iv. See m-file code
      v.

	`fminsearch`		`fminunc`		OLS
Variable	$\hat{\beta}_{looseTol}$	$\hat{\beta}_{strictTol}$	$\hat{\beta}_{looseTol}$	$\hat{\beta}_{strictTol}$	$\hat{\beta}_{closed\_form}$
Constant	39.515	42.138	44.621	44.621	44.622
age	-0.001	0.000	-0.002	-0.002	-0.002
black	-2.676	-1.804	-1.633	-1.633	-1.633
other	0.146	0.166	0.341	0.341	0.341
heartatk	-0.025	0.308	0.086	0.085	0.086
female	-3.466	-3.650	-3.868	-3.868	-3.868
highbp	0.995	0.366	0.286	0.286	0.286
NE	0.007	0.087	0.038	0.038	0.038
MW	-0.066	-0.093	-0.092	-0.092	-0.092
S	0.059	-0.010	0.022	0.022	0.022
non central city	-0.169	-0.679	-0.137	-0.137	-0.137
rural	0.205	0.086	0.109	0.109	0.109
height	-0.001	0.001	-0.013	-0.013	-0.013
weight	0.063	0.028	0.029	0.029	0.029
household size	0.011	-0.017	-0.068	-0.068	-0.068
$\hat{\sigma}^2$ (s^2)	9.498	9.088	9.017	9.017	9.030
SSE or log likelihood	-26,332.9	-26,104.6	-26,063.9	-26,063.9	93,316.7
iterations	2,239	6,935	55	91	—
N			10,349		

The tolerances made a huge difference with `fminsearch`, but not with `fminunc`. This is probably because `fminunc`'s default is 10^{-6} instead of 10^{-4} (like `fminsearch`). The answers for `fminsearch` are quite different from `fminunc`, especially under the looser convergence tolerances. This could also be a function of the dimension of the problem, though.

3. `fminsearch` performs better than I had originally thought it would, especially in 1(c). I had expected it to perform much more poorly when the dimension of the parameter vector increased so much. `fminunc` did a lot better in question 2 than in question 1 (at least for the normal MLE case). Starting values appear to play a much larger role than convergence tolerances, at least for `fminunc`. `fminsearch` appears to benefit extensively from tightening its convergence tolerances. At the end of the day, optimization is more of an art than a science, so users should be aware of the relative strengths and weaknesses of each optimizer.