Problem Set 4: Root-Finding and Optimization

Albert Rodriguez-Sala and Jacob Adenbaum
Programming for Economics—ECNM10106
The University of Edinburgh

2023

Exercise 1. Testing algorithms with the Rosenbrock function (30 points). Let's use again the Rosenbrock function to test the different algorithms seen in class. For each algoritm report: the value $\mathbf{x}^*$ that minimizes the function and how much time it took to compute the solution. If the computation takes too much time (let's say 5 minutes) stop the computation and report that the algorithm took too much time. [1] The Rosenbrock function is

$$f(\mathbf{x}) = \sum_{i}^{N-1} \left((1-x_i)^2 + (x_{i+1} - x_i^2)^2\right)$$

Where N is the number of variables.

a. define the Rosenbrock function for a general number of variables N. [2]

b. **Testing the Brute-force algorithm**. Using a range of $(-2, 2)$ for all x_i, find the minimum of the Rosenbrock function using the brute-force method for the three following cases: $N = 3$, $N = 4$, and $N = 5$. How does computational time increases as the number of variables N increases? linearly or exponentially on N?

   For the rest of the exercises use an initial value of zero for all the variables. That is $\mathbf{x}_0 = [x_{01}, x_{02}, .., x_{0N}] = [0, 0, ..., 0]$.

c. Find the minimum of the Rosenbrock function with $N = 30$ for the following three cases: using the BFGS method, using the Nelder-Mead method and using the Powell method. Which algorithm is faster?

d. **Let's test the algorithms further**. Minimize the Rosenbrock function for 100 variables, N=100. Use the BFGS method, the Nelder-Mead method and the Powell method. If N=100 with N=100 you can use another N like N=50.

e. **Algorithms comparison.** For this exercise we see that there is one algorithm that tends to to do better than the rest (especially when N is large). Which algorithm is and why? you might want to comment on the properties of the function we are minimizing and the characteristics of the algorithm.

[1] In Spyder you can stop the computation by clicking on the squared button in the console (or just close the console). If some of the fans in your laptop do not work and your laptop gets too hot, also stop the computation.

[2] You need to create the function, if you directly import it from somewhere else you will get a 0 for the whole exercise.

Exercise 2. Steady states and transitions in a Solow economy (30 points). Consider a basic Solow model with a Cobb-Douglass production function $Y_t(K_t, L_t) = AK_t^{\alpha} L_t^{1-\alpha}$, no population growth and depreciation rate δ. The capital per worker law of motion is

$$k_{t+1} = sAk_t^{\alpha} + (1-\delta)k_t \tag{1}$$

And the capital per worker in steady state is

$$k^* = \left(\frac{sA}{\delta}\right)^{\frac{1}{1-\alpha}} \tag{2}$$

The parameter values are $\alpha = 0.3$, $\delta = 0.1$.

a. **Calibration exercise**. What are the values of the saving rate, s, and the TFP, A, such that in steady state the output per worker is equal to 10, and the captial-output ratio is equal to 4?

$$y^* = 10$$
$$\frac{k^*}{y^*} = 4$$

b. **A new steady state.** Now, suppose there is a positive TFP shock and A doubles with respect to the value you found on a). Keeping everything else unchanghed, that is with the s you found on a) and $\alpha = 0.3$, $\delta = 0.1$, what is the new steady state capital per worker, k^*, and output per worker, y^*?

c. **Computing the transition to the new steady state.** Plot the capital per worker and output per worker dynamics from the steady state in a) to the new steady state in b). To do so,

   – Notice that the capital per worker dynamics under Solow are given by the law of motion given by equation (1). Where now $A = 2A_{old}$.

   – You can use a while loop that starts with the capital value in the steady state at a), and in each iteration computes the next capital value k_{t+1} given by the law of motion (1) until it reaches the point where the next capital value capital k_{t+1} is approximately equal to the steady state in b). [3]

   – To find the approximately equal value we can use a small number $\varepsilon = 0.5$ so that your while loop looks something like this

```
eps=0.5
t=0
k = k_s #the steady state k in a)
while np.abs(k-k_s2)>= eps:  #k_s2 is the steady state k in b)
```

   – For each iteration store the iteration number (t), the capital k_t, and the output y_t.

   – Plot the capital per worker and output per worker paths you found with the while loop.

[3] This is a suggestion. It is perfectly fine to solve the exercise using a different code like a for loop with some stopping criteria.

Exercise 3. Optimal life-cycle consumption paths: (40 points). Consider the problem of an individual that maximizes his/her lifetime utility by choosing consumption levels across periods. The per period utility funciton takes a CRRA form

$$u(c_t) = \frac{c_t^{1-\theta}}{1-\theta}$$

where we assume a constant relative risk aversion parameter of $\theta = 1.5$. The individual discounts future consumption at discount rate $\beta = 0.96$. The individual can save and borrow in a risk-free asset a_t at an interest rate r. For simplicity, let's assume $r = \frac{1}{\beta} - 1$.

a. **A two-period model**: consider the problem of an individual that lives for two periods ($t = 0, t = 1$) and maximizes his/her lifetime utility, subject that the budget constraint at each period hold. The maximization problem is

$$\max_{c_0, c_1, a_1} \{u(c_0) + \beta u(c_1)\}$$
$$\text{st:}$$
$$c_0 + a_1 = a_0 + y_0$$
$$c_1 = (1+r)a_1 + y_1$$

Where a_0 is an initial assets endowment, y_0, y_1 are the income flows at period 0 and period 1 respectively. Using the following parameter values $a_0 = 0, y_0 = 10, y_1 = 5$, find the optimal consumption c_0^*, c_1^* that solves the previous problem. plot the life-cycle of income and optimal consumption. That is a line plot where the x-axis are the time periods: $t = 0, 1$, line-1 is y_0, y_1, and line-2 is c_0^*, c_1^*.

b. **A life-cycle model with 4 periods**. Solve the following maximization problem

$$\max_{\mathbf{c},\mathbf{a}} \{u(c_0) + \beta u(c_1) + \beta^2 u(c_3) + \beta^3 u(c_2)\}$$
$$\text{st:}$$
$$\begin{aligned} c_0 + a_1 &= a_0 + y_0 \\ c_1 + a_2 &= (1+r)a_1 + y_1 \\ c_2 + a_3 &= (1+r)a_2 + y_2 \\ c_3 &= (1+r)a_3 + y_3 \end{aligned}$$

and plot the life-cycle of income and optimal consumption. Use the following parameter values: $a_0 = 0$, and $\mathbf{y} = [y_0, y_1, y_2, y_3] = [5, 10, 15, 0]$

c. **Comparative statics.** Solve the problem in b) for r=0.01 (case 1), for r=0.04 (case 2) and for r=0.08 (case 3). Plot the optimal consumption path along the 4 periods for each of the three cases. How does the optimal life-cycle consumption changes as r changes?

d. **Optimal consumption paths under income risk**. Now let's solve the original 2-period model with uncertainty. Suppose that period 1 income, y_1, is unknown: with probability $p^l = 0.5$ it takes a low realization $y_1^l = 2.5$, with probability $p^h = 0.5$ it takes a high realization $y_1^h = 7.5$. The rest of parameters and assumptions are exactly as in exercise a). solve the optimal consumption and saving decisions that maximize the expected lifetime utility. That is c_0, c_1, s_1 that solves the following problem

$$\max_{c_0, c_1, a_1} \{u(c_0) + \beta \mathbb{E}\left[u(c_1)\right]\} =$$
$$= \max_{c_0, c_1, a_1} \{u(c_0) + \beta \left[p^l u(c_1(y_1^l)) + p^h u(c_1(y_1^h))\right]\}$$
$$\text{st:}$$
$$c_0 + a_1 = a_0 + y_0$$
$$c_1 = (1+r)a_1 + y_1$$

Plot the income and optimal consumption paths when y_t takes a low realization and when y_t takes a high realization.

e. **Precautionary Savings.** Compare the savings in a world with certainty (case a), with respect to the world with uncertainty (d). Why do savings increase as there is income risk? Provide an economic or "real life" intuition.