Quantitative Macroeconomics Homework 4

Zuzanna Brzóska-Klimek*

October 2019

*In collaboration with Jakub Bławat, Sebastian A. Roy, and Szymon Wieczorek

Exercise 1

We are given a representative household problem of a following form:

$$\max E\left\{\sum_{t=0}^{\infty}\beta^t u(c_t, h_t)\right\} \tag{1}$$

$$u(c_t, h_t) = \ln c_t - \kappa\frac{h_t^{1+\frac{1}{\nu}}}{1+\frac{1}{\nu}} \tag{2}$$

$$c_t + i_t = y_t \tag{3}$$

$$y_t = k_t^{1-\theta}h_t^{\theta} \tag{4}$$

$$i_t = k_{t+1} - (1-\delta)k_t \tag{5}$$

1

Recursive formulation of the problem given above means putting down a Bellman equation that allows to optimise using dynamic programming. One should note that, since labour supply is fixed ($h_t = h$), the **only** state variable in this problem is capital level k_t. Therefore, Bellman equation looks as following:

$$V_t(k_t) = \left(\max\left(\ln\left(k_t^{1-\theta}h_t^{\theta} - k_{t+1} + (1-\delta)k_t\right) - \kappa\frac{h_t^{1+\frac{1}{\nu}}}{1+\frac{1}{\nu}}\right) + V_{t+1}(k_{t+1})\right) \tag{6}$$

As a means of state space discretisation, capital grid $k\_grid = (0, 0.05, 0.10, \ldots, 50)$ is used. The grid consists of 1,001 elements.

a) Brute force VFI

Method	no. of iterations	time elapsed [s]
Brute force VFI	653	11.81

Table 1: Brute force VFI

Table 1 presents details of the baseline VFI method. Figure 1 depicts obtained, optimal values of the value function[1].

Obviously, standard, baseline VFI is a robust method of solving recursive, Bellman-type problems. It is also simple to implement in the majority of computational software. However, it comes at some cost - brute-force VFI is slow, even in case of a relatively uncomplicated problem with just one state variable, as the one solved in this example.

[1] Except for its first element, which is extremely negative and falls out from the y-axis range

b) VFI with monotonicity of the decision rule

Method	no. of iterations	time elapsed [s]
VFI & decision rule monotonicity	3,000	5.93

Table 2: VFI with monotonicity of the decision rule

Table 2 shows details of the improved VFI algorithm. As one can see, despite higher number of iterations, total time elapsed is lower by almost 50%.

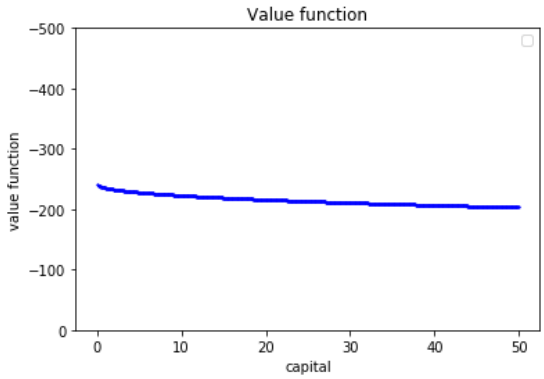

Figure 1: Representative agent model value function