Envelope Condition Method versus Endogenous Grid Method for Solving Dynamic Programming Problems

Lilia Maliar and Serguei Maliar
Los documentos de trabajo del Ivie ofrecen un avance de los resultados de las investigaciones económicas en curso, con objeto de generar un proceso de discusión previo a su remisión a las revistas científicas. Al publicar este documento de trabajo, el Ivie no asume responsabilidad sobre su contenido.

Ivie working papers offer in advance the results of economic research under way in order to encourage a discussion process before sending them to scientific journals for their final publication. Ivie’s decision to publish this working paper does not imply any responsibility for its content.

La Serie AD es continuadora de la labor iniciada por el Departamento de Fundamentos de Análisis Económico de la Universidad de Alicante en su colección “A DISCUSIÓN” y difunde trabajos de marcado contenido teórico. Esta serie es coordinada por Carmen Herrero.

The AD series, coordinated by Carmen Herrero, is a continuation of the work initiated by the Department of Economic Analysis of the Universidad de Alicante in its collection “A DISCUSIÓN”, providing and distributing papers marked by their theoretical content.

Todos los documentos de trabajo están disponibles de forma gratuita en la web del Ivie http://www.ivie.es, así como las instrucciones para los autores que desean publicar en nuestras series.

Working papers can be downloaded free of charge from the Ivie website http://www.ivie.es, as well as the instructions for authors who are interested in publishing in our series.

Versión: septiembre 2013 / Version: September 2013

Edita / Published by:
Instituto Valenciano de Investigaciones Económicas, S.A.
C/ Guardia Civil, 22 esc. 2 1º - 46020 Valencia (Spain)
Abstract

We introduce an envelope condition method (ECM) for solving dynamic programming problems. The ECM method is simple to implement, dominates conventional value function iteration and is comparable in accuracy and cost to Carroll’s (2005) endogenous grid method. Codes are available.

Keywords: Numerical dynamic programming; Value function iteration; Endogenous grid; Envelope condition; Curse of dimensionality; Large scale.

JEL classification numbers: C6, C61, C63, C68.

* Hoover Institution, Stanford University and University of Alicante. Corresponding author: Lilia Maliar, T24, Hoover Institution, 434 Galvez Mall, Stanford University, Stanford, CA 94305-6010, USA; tel 650-725-3416; email: maliarl@stanford.edu.
1 Introduction

Dynamic programming methods are an important tool in economics; see Judd (1998), Santos (1999), Rust (2008) and Stachursky (2009) for reviews. Conventional value function iteration (VFI) goes backward: we guess a value function in period $t+1$, and we solve for a value function in period $t$ using the Bellman equation. Conventional VFI is expensive: it requires us to find a root to a non-linear equation in all gridpoints, which involves interpolating value function off the grid and approximating conditional expectation in a large number of candidate solution points; see Aruoba et al. (2006) for examples assessing the cost of VFI.

Carroll (2005) introduces an endogenous grid method (EGM) that simplifies rootfinding under time iteration. The idea is to construct a grid on future endogenous state variables instead of current endogenous state variables, which are treated as unknowns. In a typical economic model, it is easier to solve for current endogenous state variables given the future state variables than to solve for future endogenous state variables given the current state variables. This is why EGM dominates conventional VFI.

In this paper, we have two contributions. First, we introduce an **envelope condition method (ECM)**, another solution method that simplifies rootfinding in dynamic programming problems. ECM does not perform conventional backward iteration on the Bellman equation but iterates forward. Also, to construct policy functions, ECM uses the envelope condition instead of the first-order conditions used by conventional VFI and EGM. We find that systems of equations produced by ECM are typically easier to solve than those produced by conventional VFI. In this sense, ECM is similar to EGM.

Second, we introduce versions of the EGM and ECM methods that approximate derivatives of value function instead of value function itself. We find that these versions produce far more accurate solutions than do similar methods that approximate value function itself.

We compare the EGM and ECM methods using both analytical arguments and numerical examples. We find that EGM and ECM are nearly identical in terms of accuracy and speed in our test problem, the neoclassical growth model with elastic labor supply. Codes are available at http://www.stanford.edu/~maliarl.

2 The model

We study the standard neoclassical growth model with elastic labor supply.
2.1 Bellman equation

We solve for value function $\mathcal{V}$ that satisfies the Bellman equation,

$$
V (k, a) = \max_{k', c, \ell} \{ u (c, \ell) + \beta E [V (k', a')] \}
$$

\[ \text{s.t. } k' = (1 - \delta) k + af (k, \ell) - c, \]

\[ \ln a' = \rho \ln a + \epsilon', \quad \epsilon' \sim \mathcal{N} (0, \sigma^2), \]

where $k$, $c$, $\ell$ and $a$ are capital, consumption, labor and productivity level, respectively; $\beta \in (0, 1)$; $\delta \in (0, 1]$; $\rho \in (-1, 1)$; $\sigma \geq 0$; the utility and production functions, $u$ and $f$, respectively, are strictly increasing, continuously differentiable and concave; the primes on variables denote next-period values, and $E [V (k', a')]$ is an expectation conditional on state $(k, a)$.

2.2 Optimality conditions

We divide the optimality conditions in two blocks. The first block identifies policy functions that correspond to a given value function $V$, and the second block identifies a value function that corresponds to given policy functions.

2.2.1 Block 1: Identifying policy functions given a value function

If a solution to Bellman equation (1)–(3) is interior, the optimal quantities satisfy first-order conditions (FOCs) with respect to labor and consumption and the envelope condition, which, respectively, are

$$
\begin{align*}
    u_\ell (c, \ell) &= -u_c (c, \ell) a f_\ell (k, \ell), \\
    u_c (c, \ell) &= \beta E [V_k (k', a')], \\
    V_k (k, a) &= u_c (c, \ell) [1 - \delta + af_k (k, \ell)],
\end{align*}
$$

as well as budget constraint (2). Here, $F_x (..., x, ...)$ denotes a first-order partial derivative of function $F (..., x, ...)$ with respect to variable $x$.

2.2.2 Block 2: Identifying a value function given policy functions

In the optimum, value function $V$ and its derivative $V_k$ satisfy

$$
\begin{align*}
    V (k, a) &= u (c, \ell) + \beta E [V (k', a')], \\
    V_k (k, a) &= \beta [1 - \delta + af_k (k, \ell)] E [V_k (k', a')].
\end{align*}
$$
Condition (7) is Bellman equation (1) evaluated under the optimal policy functions (which makes a maximization sign unnecessary), and condition (8) follows by combining (5) and (6).

2.3 Discussion

Envelope condition (6) is central to our analysis.\(^1\) Observe that we have two conditions that describe the relation between \(V_k\) and the policy functions: one is FOC (5) and the other is envelope condition (6). Conventional VFI and EGM of Carroll (2005) approximate policy functions using FOC (5), namely, they solve the system (2), (4) and (5). In contrast, our ECM method will approximate policy functions using envelope condition (6), namely, it will solve the system (2), (4) and (6). In Sections 3 and 4, we show that system of equations built on envelope condition (6) are easier to solve than system of equations built on conventional FOC (5), in which case ECM is a preferred choice.

Furthermore, the envelope condition provides a basis for condition (8). This condition allows us to approximate \(V_k\) without finding \(V\). Under our construction, all methods described in the paper can approximate a solution by iterating on either (7) or (8) or both, whereas the previous literature including conventional VFI and EGM of Carroll (2005) iterate only on Bellman equation (7). In Section 5, we show that iteration on (8) leads to far more accurate solutions than iteration on (7).

3 The model with inelastic labor supply

Consider first a model with inelastic labor supply under the following assumptions

\[
u (c, \ell) = \frac{c^{1-\gamma} - 1}{1 - \gamma} \quad \text{and} \quad f (k, \ell) = k^\alpha, (9)\]

where \(\gamma > 0\) and \(\alpha \in (0, 1)\). In this case, we have \(\ell = 1\), and FOC (4) is absent.

3.1 Conventional VFI

The conventional VFI method makes a guess on the future value function \(V (k', a')\) and identifies policy functions using budget constraint (2) and FOC (5). By substi-

\(^1\)Typically, the envelope condition is used to derive the Euler equation (namely, (6) is updated to get \(V_k (k', a')\) and the result is substituted into (5) to eliminate the unknown derivative of the value function). In the present paper, we do not derive the Euler equation but concentrate on the envelope condition in the form (6).
tuting $c$ from (2) into (5) under the assumptions (9), we obtain

$$\beta E[V_k (k', a')] = [k' - (1 - \delta) k - ak^\alpha]^{-\gamma}.$$  \hfill (10)

We must solve (10) for $k'$ in each grid point $(k, a)$. Finding a solution to (10) is expensive. For example, if we parameterize $V$ with a polynomial function, then solving (10) includes interpolation of $V_k$ to new values $(k', a')$, as well as approximation of conditional expectation $E[V_k (k', a')]$. We must explore many different candidate values of $(k', a')$ until we find a solution to (10).

### 3.2 Endogenous grid method

EGM of Carroll (2005) also makes a guess on the future value function $V(k', a')$ and identifies policy functions using budget constraint (2) and FOC (5). The difference is that EGM treats the future endogenous state variable as fixed, and it treats the current endogenous state variable as unknown. Since the values for $k'$ are fixed, it is possible to compute up-front $E[V (k', a')] \equiv W(k', a)$ and $E[V_k (k', a')] \equiv W_k(k', a)$.

Consider again the system (2), (4) and (5) under the assumptions (9). Now, we find $c$ directly from (5), $c = [\beta W_k (k', a')]^{-1/\gamma}$, and we are left to solve for $k$ that satisfies budget constraint (2) given $(k', a)$

$$(1 - \delta) k + ak^\alpha = [\beta W_k (k', a')]^{-1/\gamma} + k'. \hfill (11)$$

Observe that (11) is easier to solve numerically than (10) because it does not involve either interpolation or approximation of conditional expectation.

**Carroll’s (2005) change of variables.** Still, equation (11) must be solved numerically. However, Carroll (2005) finds a clever change of variables that makes unnecessary solving (11) on each iteration. He introduces a new variable $Y \equiv (1 - \delta) k + ak^\alpha = c + k'$, which allows us to rewrite Bellman equation (1) as

$$V(Y, a) = \max_{k'} \left\{ \frac{c^{1-\gamma} - 1}{1 - \gamma} + \beta W(Y', a) \right\}, \hfill (12)$$

where $W(Y', a) = E[V(Y', a')]$. The FOC of this problem is

$$c^{-\gamma} = \beta E\left[ V_Y (Y', a') \left( 1 - \delta + \alpha a' (k')^{\alpha-1} \right) \right]. \hfill (13)$$

Since we know $Y' \equiv (1 - \delta) k' + a' (k')^\alpha$, we can find the expectation in the right side of (13) and hence, we can compute $c$ and $Y = k' + c$. Therefore, we can iterate on Bellman equation (12) without using a solver. Once $V$ is computed, we find $k$ that corresponds to $Y = (1 - \delta) k + ak^\alpha$ using a numerical solver (just once).
3.3 Envelope condition method

Like conventional VFI, our ECM method operates on exogenous grid however makes a guess on the current value function $V(k, a)$ (or its derivative $V_k(k, a)$) instead of the future value function. This enables us to solves for $c$ using the envelope condition (6) instead of FOC (5). Under the assumptions (9), $c$ can be derived explicitly from (6),

$$c = \left( \frac{V_k(k, a)}{1 - \delta + \alpha a (k)^{a-1}} \right)^{-1/\gamma}. \quad (14)$$

We can next compute $k'$ directly from budget constraint (2). In this example, ECM is simpler than Carroll’s (2005) EGM as all policy functions can be constructed analytically and a solver must never be used (not even once).

3.4 Discussion

Four combinations are possible from two alternative conditions for $V_k$ (FOC (5) and envelope condition (6)) and two alternative grids (exogenous and endogenous). So far, we have distinguished two competitive methods: one is EGM of Carroll (2005) (FOC (5) and endogenous grid) and the other is our ECM (envelope condition (6) and exogenous grid). The conventional VFI (FOC (5) and exogenous grid) is not competitive. Therefore, we are left to explore the remaining combination (envelope condition (6) and endogenous grid). Combining (2) and (14) yields

$$(1 - \delta) k + ak^{\alpha} = \left( \frac{V_k(k, a)}{1 - \delta + \alpha a (k)^{a-1}} \right)^{-1/\gamma} + k'. \quad (15)$$

We must solve (15) for $k$ given $(k', a)$. This involves evaluations of $V_k(k, a)$ for many candidate solution points $(k, a)$, which is costly. We conclude that the combination of the envelope condition and endogenous grid does not lead to a competitive method. Our results are suggestive for other applications.

4 The model with elastic labor supply

We now consider the model with elastic labor supply under the following assumptions:

$$u(c, \ell) = \frac{c^{1-\gamma} - 1}{1 - \gamma} + B \frac{(1 - \ell)^{1-\alpha} - 1}{1 - \mu} \quad \text{and} \quad f(k, \ell) = k^{\alpha} \ell^{1-\alpha}, \quad (16)$$

where $\gamma > 0$, $\mu > 0$ and $\alpha \in (0, 1)$. We restrict attention to EGM and ECM that we found to be competitive.
4.1 Endogenous grid method

Under EGM, we must solve equations (2), (4) and (5) for $(c, \ell, k)$ given $(k', a)$. As in the model with inelastic labor supply, we compute $E[V(k', a')] \equiv W(k', a)$, $E[V_k(k', a')] \equiv W_k(k', a)$ given $V$, and we find $c = [\beta W_k(k', a)]^{-1/\gamma}$ using (4). Under (16), we can express $k$ from (4) and substitute it into (2) to get

$$k' = (1 - \delta) \left( \frac{B (1 - \ell)^{-\mu}}{\beta W_k(k', a) a (1 - \alpha)} \right)^{1/\alpha} \ell$$

$$+ \frac{B (1 - \ell)^{-\mu} \ell}{\beta W_k(k', a) (1 - \alpha)} - [\beta W_k(k', a)]^{-1/\gamma}. \quad (17)$$

The equation (17) that must be solved numerically for one unknown $\ell$. This equation is relatively cheap as it does not involve either interpolation or approximation of expectations.

4.2 Envelope condition method

Under ECM, we must solve equations (2), (4) and (6) for $(c, \ell, k')$ given $(k, a)$. By substituting $c^{-\gamma}$ from (4) into envelope condition (6), we obtain

$$V_k(k, a) = \frac{B (1 - \ell)^{-\mu}}{a (1 - \alpha) k^\alpha \ell^{-\alpha}} \left[ 1 - \delta + a \alpha k^{\alpha-1} \ell^{1-\alpha} \right]. \quad (18)$$

We must solve equation (18) for $\ell$. Like (17), the equation (18) does not involve either interpolation or approximation of expectations.

4.3 Discussion

Under our implementation, the rootfinding problems under EGM and ECM are comparable in their complexity. In both cases, we must find a solution to a non-linear equation in each grid point. Such an equation is relatively cheap to solve as it does not involve either interpolation or approximation of expectations.

In the model with elastic labor supply, Carroll’s (2005) change of variables does not avoid rootfinding. The variable $Y' = d' f (k', \ell') + k'$ depends on future labor $\ell'$, and $E[V(Y', a')]$ cannot be computed without specifying labor policy functions. Barillas and Fernández-Villaverde (2007) propose a way of extending EGM to the model with elastic labor supply. Namely, they fix a policy function for labor $\ell =$
construct the grid of \((Y', a)\), solve the model on that grid holding \(L\) fixed and use the solution to reevaluate \(L\); and they iterate on these steps until \(L\) converges.

Our implementation of EGM for the model with elastic labor supply differs from that in Barillas and Fernández-Villaverde (2007). First, we use future endogenous state variables for constructing gridpoints but we do not use Carroll’s (2005) change of variables. Second, to deal with rootfinding, we use a numerical solver while Barillas and Fernández-Villaverde (2007) iterate on a state contingent policy function for labor \(L(k', a)\).

5 Numerical analysis

We compare the performance of EGM and ECM in the context of the model with elastic labor supply.

5.1 Methodology

We calibrate the model (1)–(3) under (16) such that in the steady state, the capital-output ratio is \(\pi_k = 10\), the consumption-output ratio is \(\pi_c = 3/4\), the steady state labor is \(\ell = 1/3\) and \(\alpha = 1/3\); this implies \(\beta = 0.99\), \(\delta = 0.025\) and \(B = (1 - \alpha) \pi_k^{(1-\gamma)\alpha/(1-\alpha)} \pi_c^{-\gamma} (1 - \ell)^{\mu} \ell^{-\gamma}\). In the benchmark case, we use \((\gamma, \mu) = (2, 2)\). The parameters in (3) are \(\rho = 0.95\) and \(\sigma = 0.01\). Our design of EGM and ECM is similar. As a solution domain, we use a rectangular, uniformly spaced grid of 10 \(\times\) 10 points for capital and productivity within an ergodic range. We use a 3-node Gauss-Hermite quadrature rule for approximating integrals. We parameterize value function with complete ordinary polynomials of degrees up to 5. To solve for the polynomial coefficients, we use fixed-point iteration. To solve nonlinear equations (17) and (18), we use a solver \(csolve\) written by Christopher Sims. We use MATLAB software, version 7.6.0.324 (R2008a) and a desktop computer ASUS with Intel(R) Core(TM)2 Quad CPU Q9400 (2.66 GHz), RAM 4MB. A detailed description of the algorithms is provided in the Appendix A.

5.2 Results for the model with elastic labor supply

We first solve for \(V\) by iterating on Bellman equation (7); we refer to the corresponding methods as EGM-VF and ECM-VF. The results are shown in Table 1. The performance of EGM-VF and ECM-VF is very similar. EGM-VF produces slightly smaller maximum residuals, while ECM-VF produces slightly smaller average residuals. EGM-VF is somewhat slower than ECM-VF.
Table 1: Accuracy and speed of EGM-VF and ECM-VF in the model with elastic labor supply.\(^a\)

<table>
<thead>
<tr>
<th>Polynomial degree</th>
<th>EGM-VF</th>
<th>ECM-VF</th>
<th>EGM-VF</th>
<th>ECM-VF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(L_1)</td>
<td>(L_\infty)</td>
<td>CPU</td>
<td>(L_1)</td>
</tr>
<tr>
<td>1st</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2nd</td>
<td>-3.28</td>
<td>-2.81</td>
<td>8.3</td>
<td>-3.34</td>
</tr>
<tr>
<td>3rd</td>
<td>-4.31</td>
<td>-3.99</td>
<td>8.9</td>
<td>-4.38</td>
</tr>
<tr>
<td>4th</td>
<td>-5.32</td>
<td>-4.96</td>
<td>7.3</td>
<td>-5.45</td>
</tr>
<tr>
<td>5th</td>
<td>-6.37</td>
<td>-5.85</td>
<td>6.5</td>
<td>-6.57</td>
</tr>
</tbody>
</table>

\(a\) Notes: \(L_1\) and \(L_\infty\) are, respectively, the average and maximum of absolute residuals across optimality condition and test points (in log10 units) on a stochastic simulation of 10,000 observations; CPU is the time necessary for computing a solution (in seconds).

We next solve for \(V_k\) by iterating on (8); we call these methods EGM-DVF and ECM-DVF. The results are provided in Table 2. Again, EGM-DVF and ECM-DVF perform very similarly. Both methods deliver accuracy levels that are about an order of magnitude higher than those of EGM-VF and ECM-VF. Overall, we attain accuracy levels that are comparable to the best accuracy attained in the related literature.

Table 2: Accuracy and speed of EGM-DVF and ECM-DVF in the model with elastic labor supply.\(^a\)

<table>
<thead>
<tr>
<th>Polynomial degree</th>
<th>EGM-DVF</th>
<th>ECM-DVF</th>
<th>EGM-DVF</th>
<th>ECM-DVF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(L_1)</td>
<td>(L_\infty)</td>
<td>CPU</td>
<td>(L_1)</td>
</tr>
<tr>
<td>1st</td>
<td>-3.03</td>
<td>-2.87</td>
<td>8.1</td>
<td>-3.08</td>
</tr>
<tr>
<td>2nd</td>
<td>-4.13</td>
<td>-3.82</td>
<td>7.2</td>
<td>-4.18</td>
</tr>
<tr>
<td>3rd</td>
<td>-5.06</td>
<td>-4.77</td>
<td>7.3</td>
<td>-5.20</td>
</tr>
<tr>
<td>4th</td>
<td>-6.09</td>
<td>-5.64</td>
<td>7.4</td>
<td>-6.29</td>
</tr>
<tr>
<td>5th</td>
<td>-7.12</td>
<td>-6.26</td>
<td>7.6</td>
<td>-7.36</td>
</tr>
</tbody>
</table>

\(a\) Notes: \(L_1\) and \(L_\infty\) are, respectively, the average and maximum of absolute residuals across optimality condition and test points (in log10 units) on a stochastic simulation of 10,000 observations; CPU is the time necessary for computing a solution (in seconds).
polynomial, we effectively approximate $V_k$ with a polynomial which is one degree lower, i.e., we "lose" one polynomial degree.

We finally implement versions of EGM and ECM which approximate $V$ jointly with $V_k$ by iterating on both (7) and (8); we call them EGM-VF&DVF and ECM-VF&DVF. We specifically fit a polynomial approximation for $V$ on the grid using a constrained linear least-squares that imposes a linear restriction on the coefficients of a polynomial that approximates $V_k$. This procedure is similar in spirit to a Hermite interpolation method described in Cai and Judd (2012). In our simple example, approximating $V$ jointly with $V_k$ lead to the same results as those obtained approximating $V$ alone. However, in more complex models in which value function has many endogenous arguments, fitting both $V$ and $V_k$ on the grid may improve accuracy of solutions because it imposes consistency on cross derivatives of $V$.

6 Conclusion

Conventional VFI is expensive. Carroll (2005) introduces the EGM method that reduces the cost of value iteration dramatically. In this paper, we propose the ECM method that can compete with Carroll’s (2005) method. In our simple application, EGM and ECM perform similarly. But in more complex applications, one method may lead to a more simple system of equations and thus, be preferable to the other. One application in which ECM can be a useful choice is models of sovereign default; see, e.g., Villemot (2012).

In the paper, we build ECM and EGM using tensor product grids. However, ECM and EGM can be implemented using nonproduct techniques that are tractable in high dimensional applications; see Maliar and Maliar (2005) for a numerical method that solves for a value function on simulated series, and see Judd et al. (2011, 2012) for effective non-product grid constructions, low-cost monomial integration formulas and numerically stable fitting methods. In particular, Maliar and Maliar (2012) show versions of ECM that solves dynamic programming problems with up to 16 state variables.

References


Appendix A

We first provide a description of 3 versions of ECM used.

<table>
<thead>
<tr>
<th>ECM-VF, ECM-DVF, ECM-VF&amp;DVF</th>
</tr>
</thead>
</table>

**Step 0. Initialization.**

- a. Choose an approximating function \( \hat{V} (\cdot; b) \approx V \).
- b. Choose integration nodes, \( \epsilon_j \), and weights, \( \omega_j, j = 1, ..., J \).
- c. Construct a grid \( \Gamma = \{k_m, a_m\}_{m=1,...,M} \).
- d. Make an initial guess on \( b(1) \).

**Step 1. Computation of a solution for \( V \).**

At iteration \( i \), for \( m = 1, ..., M \),

- a. Solve for \( \eta_m \) that satisfies
  \[
  B (1 - \ell_m)^{-\mu} (1 - \delta + a\alpha k_m^{-1} \ell_m^{1 - \alpha}) = \hat{V}_k (k_m, a_m; b(i)) \ a (1 - \alpha) k_m \ell_m^{-\alpha};
  \]
- b. Compute \( c_m \equiv \left[ \frac{\hat{V}_k(k_m, a_m; b(i))}{1 - \delta + a\alpha k_m^{-1} \ell_m^{1 - \alpha}} \right]^{-1/\gamma};
- c. Compute \( k'_m = (1 - \delta) k_m + a_m k_m \ell_m^{1 - \alpha} - c_m; \)

**ECM-VF.** Find value function on the grid

\[
v_m \equiv \frac{d_m - 1}{1 - \gamma} + B \frac{(1 - \ell_m)^{1 - \mu}}{1 - \mu} + \beta \sum_{j=1}^J \omega_j \hat{V} \left( k'_m, a_m \exp (\epsilon_j) ; b(i) \right);
\]

**ECM-DVF.** Find derivative of value function on the grid

\[
d_m = \beta \left[ 1 - \delta + a\alpha k_m^{-1} \ell_m^{1 - \alpha} \right] \sum_{j=1}^J \omega_j \hat{V}_k \left( k'_m, a_m \exp (\epsilon_j) ; b(i) \right);
\]

**Step 2. Computation of \( b \) that fits the value function on the grid.**

- a. Run a regression to find \( \hat{b} \)
  
  ECM-VF. \( \hat{b} = \arg \min \sum_{m=1}^M \left\| v_m - \hat{V} (k_m, a_m; b) \right\| \).
  
  ECM-DVF. \( \hat{b} = \arg \min \sum_{m=1}^M \left\| d_m - \hat{V}_k (k_m, a_m; b) \right\| \).
  
  ECM-VF&DVF. \( \hat{b} = \arg \min \sum_{m=1}^M \left\| v_m - \hat{V} (k_m, a_m; b) \right\| \text{ s.t. } d_m = \hat{V}_k (k_m, a_m; b). \)

- b. Use damping to compute \( b(i+1) = (1 - \xi) b(i) + \xi \hat{b} \).

- c. Check for convergence: end Step 2 if \( \frac{1}{M} \sum_{m=1}^M \left| \frac{(k_m')'(i+1) - (k_m')'(i)}{(k_m')'(i)} \right| < \infty. \)

We now provide a description of 3 different version of EGM used (steps that are identical under ECM and EGM are omitted).
c. Construct a grid \( \Gamma = \{k'_m, a_m\}_{m=1}^{\ldots, M} \) 

At iteration \( i \), for \( m = 1, \ldots, M \),

a. Find \( c_m = \left( \beta \sum_{j=1}^{J} \omega_j \tilde{V}_k \left( k'_m, a^0_i \exp (\epsilon_j); b^{(i)} \right) \right)^{-1/\gamma} \)

b. Solve for \( \ell_m \) that satisfies \( k'_m = (1 - \delta) \left( \frac{B(1-\ell_m)^{-\mu}}{c^m_m a(1-\alpha)} \right)^{1/\alpha} \ell_m + \frac{B(1-\ell_m)^{-\mu} \ell_m}{c^m_m (1-\alpha)} - c_m \),

c. Compute \( k_m \equiv \left( \frac{B(1-\ell_m)^{-\mu}}{c^m_m a(1-\alpha) c^{m-\mu}_m} \right)^{1/\alpha} \); ...
Ivie
Guardia Civil, 22 - Esc. 2, 1º
46020 Valencia - Spain
Phone: +34 963 190 050
Fax: +34 963 190 055

Website: www.ivie.es
E-mail: publicaciones@ivie.es