第 3 回：動的計画法

初稿・再校：反映済み

1 局所的近似と大域的近似

前回は、実際に 2 期間モデルを近似的に解く事によって、数値計算を使ってどのような事が出来るのかと数値計算手法の基本的な使い方について概観を示した。今回は本格的な動学的マクロ経済モデルを導入しながら、その代表的な解法の一つである**動的計画法 (dynamic programming)** について説明する。

現在のマクロ経済モデルは**新古典派成長モデル (neoclassical growth model)** に支えられているといっても過言ではない。新古典派成長モデルは、代表的個人と代表的企業が経済活動を無限期間行うと想定した際の動学的最適化問題として定義され、実物的景気循環理論やニューケインジアン DSGE モデルなど現在のマクロ経済モデルのベースとなっている。

前回紹介したとおり、似たような経済問題であっても、モデルの特性や知りたいことに応じて解法はいくつか存在していて、常にどれかの方法が「最適」になるわけではない。例えば、景気循環の研究では定常状態の近傍を**局所的 (local)** に (対数) 線形近似で解くアプローチがいまでも頻繁に用いられている[1]。線形近似は現在でも重要な手法の 1 つであるが、近年は金融政策のゼロ金利制約や家計の借り入れ制約といった非線形な現象にも注目することが多くなったため、モデルを**大域的 (global)** に解く必要に迫られる場合も多々ある。

動的計画法は大域的に消費者や企業の意思決定問題を解くことができ、強い非線形性が存在するモデルにおいて強みを発揮する。また、**構造推計 (structural estimation)** と呼ばれる数値計算と推計を組み合わせた手法のパーツとしても利用される汎用性の高いアプローチである

2 ロビンソン・クルーソーとベルマン方程式

様々なマクロ経済モデルの基礎になっている**代表的個人モデル (representative agent model)** を設定しよう。代表的個人モデルは、無人島で生活するロビンソン・クルーソーに例えられる。自分しかいない孤島で暮らしていくことになったロビンソン・クルーソーは、生きていくために一人で生産と貯蓄の計画を建てないといけない。無人島なので、他の経済主体との市場における取引は

[1] 景気循環研究における線形近似アプローチについては既に良書が存在している。例えば、加藤 (2006) や廣瀬 (2012) を参照せよ。

存在しないが、生産と消費・貯蓄だけでも立派な経済活動である。彼がやるべきことは、助けが来るまでの期間について、消費と貯蓄の**最適経路 (optimal path)** を考えて実行することである。

時間は離散的に進むと考えて、現時点でのカレンダー年を $t \geq 0$ と書くことにする。また、話を簡単にするために必ず T 期終了後に助けが来る事がわかっているとしよう ($t = 0, ..., T$)。無人島にたどり着いたロビンソン・クルーソーが考えるべきことは、0 期時点で手元にある資本 (種) k_0 を使って、$T+1$ 期間の最適な消費・貯蓄計画を立てることである。

この種はそのまま食べてしまうことも出来るが、植えて収穫をすれば増やすことも出来る。全て食べきってしまうと次期に食べるものがなくなって死んでしまうが、全ての種を蒔いてしまうと、今日ひもじい思いをすることになる。そのため、ちょうどよいバランスを考えなくてはいけない。

この状況を予算制約として表現すると、

$$c_t + k_{t+1} \leq f(k_t) \tag{1}$$

となる。ただし、c_t は t 期における消費である。

$f(k_t)$ は生産関数で、k_t 単位の種を蒔くと $y_t = f(k_t)$ だけの種が収穫できる。標準的な仮定として、生産関数は $f'(k_t) > 0$ 及び $f''(k_t) < 0$ を満たすとする。なお、(1) 式では生産した財を消費も貯蓄もせず捨ててしまうことを許容するために不等式 $\leq$ を用いているが、種の効用が正である限り予算制約は等式で成立する。

ロビンソン・クルーソーの目的は無人島生活での消費から得られる効用の割引現在価値を最大化することなので、**目的関数 (objective function)** は、

$$\max_{\{c_t, k_{t+1}\}_{t=0}^{T}} \sum_{t=0}^{T} \beta^t u(c_t) \tag{2}$$

となる。効用関数 $u(c_t)$ にも、$u'(c_t) > 0$ 及び $u''(c_t) < 0$ を仮定しておく。

ロビンソン・クルーソー経済において我々が知りたいことは、(2) 式の目的関数を最大にする消費及び資本の経路 $\{c_t, k_{t+1}\}_{t=0}^{T}$ である。T 期に必ず救出されるのであれば、無人島に種を残していっても仕方がないので、T 期にある種は全て食べてしまうのが最適な計画である。すなわち、最適貯蓄計画では必ず $k_{T+1} = 0$ となる[2]。また無人島では誰かから種を借りることは出来ないので、必ず $k_t \geq 0$ でなければならない。

予算制約式 (1) のもとで (2) 式を最大化するという動学的最適化問題を解く方法はいくつか存在

[2] 最終期における貯蓄の割引現在価値がゼロになる条件は**横断条件 (transversality condition)** と呼ばれている。また、最終期に負債を残す事を排除する条件を**非ポンジゲーム条件 (no Ponzi game condition)** と呼ぶ。両者の直観的な説明については齊藤 (2006) を参照せよ。

している[3]。学部生向けの経済数学でもおなじみのラグランジュ未定乗数法を用いると、前回も出てきた一階条件 (オイラー方程式)

$$u(c_t) = \beta u'(c_{t+1}) f'(k_{t+1}) \tag{3}$$

を導出することが出来る。このような定式化を**逐次問題 (sequence problem)** と呼ぶ。

(3) 式を直接解いていく事も出来るが、今回は別のアプローチを紹介する。ロビンソン・クルーソーの立場にたつと、最適計画を立てる際に必要な情報は、無人島で生活する残存期間 T とスタート時点で保有している資本 k_0 である。そこで問題を組み立てなおそう。

$V_0(k_0)$ を 0 期において k_0 だけ資本を保有していた際に達成できる最大効用と定義する。これを**価値関数 (value function)** と呼ぶ。目的関数は、価値関数を使って次のように書き直すことが出来る。

$$\begin{aligned} V_0(k_0) &= \max\{u(c_0) + \beta u(c_1) + \beta^2 u(c_2) + ... + \beta^T u(c_T)\}, \\ &= \max\{u(c_0) + \beta[u(c_1) + \beta u(c_2) + ... + \beta^{T-1} u(c_T)]\}, \\ &= \max\{u(c_0) + \beta V_1(k_1)\}. \end{aligned} \tag{4}$$

ただし、$V_1(k_1)$ は 1 期に資本を k_1 だけ資保有していた際の最大効用である。また、同じだけ資本を保有していても残存期間が異なれば価値関数は異なる ($V_0(k) \neq V_1(k)$) ため、経過期間も状態変数の一つとして明示的に書く必要がある。

0 期に k_0 だけの資本を保有していた経済主体が $T+1$ 期間を通じた最適貯蓄計画を解く問題を、0 期の消費・貯蓄選択問題と「1 期に資本 k_1 だけ保有している際の最適貯蓄問題」に分割した。言い換えると、次期以降は k_1 からスタートして最適な貯蓄計画をするとして、今期にどれだけ食べてどれだけ残して次期の初期資本とするかという問題に設定し直すのである。これにより、$T+1$ 期間の消費と貯蓄の流列 $\{c_t, k_{t+1}\}_{t=0}^T$ を探す問題を、k_0 を所与として最適な k_1 を探す問題に変換出来る。このように再定義する事が出来る背景には、0 期に最適計画が出来ているのであれば、それ以降の部分問題も最適計画になっているという前提がある。この性質を**最適性原理 (principle of optimality)** と呼ぶ。また、(4) 式は**ベルマン方程式 (Bellman equation)** と呼ばれている。

逐次問題の解とベルマン方程式を用いて導出した解は、緩やかな条件のもとで一致することが知られている[4]。前回学んだように同じ問題でも解法は様々であり、以下ではこの動的計画法を使っ

[3] 連続時間モデルであれば、**最大値原理 (maximum principle)** や**変分法 (calculus of variation)** などを用いて解くことが出来る。例えば、Wickens (2012) を参照せよ。
[4] 詳細は、Stokey et al. (1989) の第 4 章を参照せよ。

た解法を紹介していく。動的計画法は理系分野で発達した手法で、さまざまな専門書が存在している[5]。

3　有限期間モデルの解法

3.1　後ろ向き帰納法

有限期間のロビンソン・クルーソー経済を実際に解いてみよう。我々が知りたいのは、価値関数 $V_t(k)$ と、前回同様に今期の資本と次期の最適な資本を結びつける**政策関数 (policy function)** $k_{t+1} = g(k_t)$ である。政策関数が得られれば、初期資本 k_0 のもとで、$k_1 = g(k_0) \rightarrow k_2 = g(k_1) \rightarrow$ ... と解いていける。数値計算の基本的な考え方も前回と同様で、後ろ向きに解いていけばよい。

今考えている経済は仮想的であるがカリブレーションによって、パラメータの数値を設定しておこう。ロビンソン・クルーソーは 10 期間後に助けが来る事を知っており、割引因子 β は 0.96 とする。効用関数は対数型 ($u(c) = \ln c$) とする。また、生産関数は $f(k) = k^\alpha$ と特定化をして、$\alpha = 0.4$ と設定しよう。

状態変数 k を離散化して、$k^i \in \{k^1, ..., k^I\}$ を資本に関するグリッドとする。最終期である T 期においては、資本 (種) を残しておく必要はない。すなわち、

$$c_T^i = f(k^i), \quad V_T(k^i) = \ln c_T^i$$

をグリッド上で計算すればよい。

1 期さかのぼって、$T-1$ 期における最適化問題を考えてみよう。$T-1$ 期と T 期の 2 期間の最適化問題なので、こちらは前回勉強した 2 期間モデルと構造はまったく同じである。前回紹介した最適化によって政策関数と価値関数を計算すれば、近似的な価値関数 (点の集まり)$\{V_{T-1}(k^i)\}_{i=1}^I$ を得られる。

さらにさかのぼって、$T-2$ 期における最適化問題を考えよう。解きたい問題は、

$$V_{T-2}(k_{T-2}^i) = \max_{k_{T-1}} \{u([k_{T-2}^i]^\alpha - k_{T-1}) + \beta V_{T-1}(k_{T-1})\} \tag{5}$$

である。

ここで問題が発生する。我々が知っている右辺第 2 項の価値関数は連続的な関数ではなく、$\{V_{T-1}(k^i)\}_{i=1}^I$ という離散個の点の集まりである。しかし、(5) 式において、最適な次期の資本

[5] 代表的なテキストとして、Bertsekas (2012, 2017) がある。

k_{T-1} が必ずしもグリッド上の値を取る保証はない。もし、最適な資本が $k^*_{T-1} \in [k^i, k^{i+1}]$ にある場合にどうすればよいのであろうか? そこで必要になってくるのが**内挿法 (interpolation)** という考え方である。

3.2 内挿法と外挿法

内挿法にはいくつかの方法が存在している。一番シンプルな内挿は直線で結んだ間の点を取るというものである。例えば、k^i と k^{i+1} の間に知りたい k_{T-1} が存在しているのであれば、価値関数の値も $V_{T-1}(k^i)$ と $V_{T-1}(k^{i+1})$ の間にあるはずと考えるのは自然である。直線的に間の値を取るという方法を**線形補間 (linear interpolation)** と呼ぶ。

$k_{T-1} \in [k^i, k^{i+1}]$ であれば、

$$V_{T-1}(k_{T-1}) = V_{T-1}(k^i)\frac{k^{i+1} - k_{T-1}}{k^{i+1} - k^i} + V_{T-1}(k^{i+1})\frac{k_{T-1} - k^i}{k^{i+1} - k^i}$$

が線形補間によって近似した貯蓄水準となる。

線形補間は直観的で強力な方法であるが、欠点も存在する。証明は省略するが、真の価値関数 V は微分可能な関数である。それを線形補間してしまうと、線と線のつなぎ目 (グリッド上) で微分不可能になるため、価値関数の微分可能性という重要な性質の一つを落としてしまうという問題が生じる。

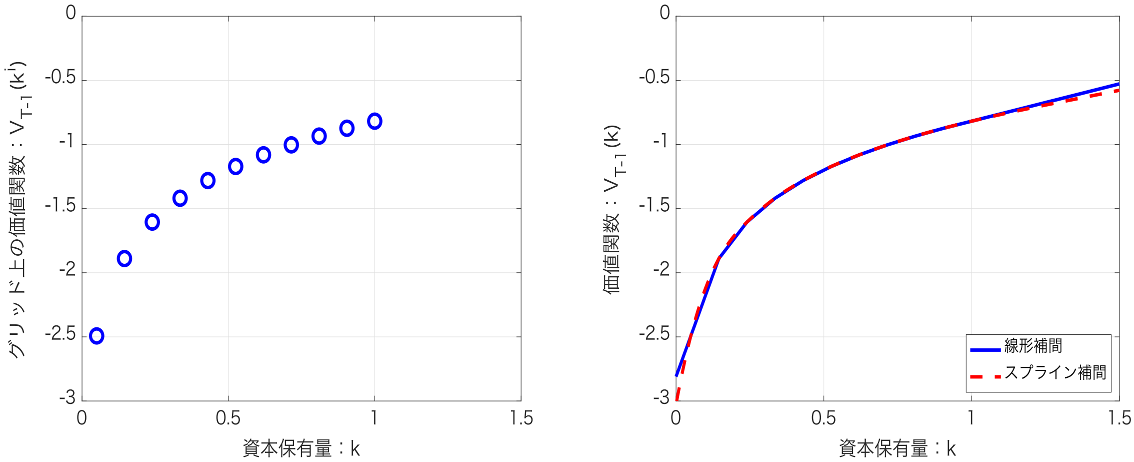
微分可能性を維持する方法はいくつか考えられるが、頻繁に用いられる方法の 1 つとして **3 次のスプライン補間 (cubic spline interpolation)** がある。3 次のスプライン補間とは、グリッド間を 3 次関数で近似しつつ、グリッド上で導関数が一致するようにつなげることによって、全体として微分可能になるように補完する方法である。

図 1(a) は価値関数 $V_{T-1}(k^i)$ のグリッド上の値である。グリッドは 0.05 から 1 の間に 11 個の点を取っている[6]。この点を線形補間およびスプライン補間したのが図 1(b) である。線形補間のほうが若干、カクカクしていて、一方でスプライン補間はスムーズな曲線である事が見て取れる。幸い価値関数は複雑な形ではないため、どちらも価値関数をうまく近似しているように見えるが、定義域 $[0.05, 1]$ の外側を補間 (**外挿 (extrapolation)** と呼ぶ) する場合には、線形補間とスプライン補間で違いが出ているのに気がついてほしい。

現在の資本 k が小さくなると消費も 0 に近づくため、対数型効用関数のもとで限界効用はどんどん大きくなり、価値関数の傾きは ∞ に近づいていく。どちらも、その性質を完全に再現できてい

[6] 資本が 0 だと生産が出来なくなるため、定義域の下限は必ず正値である必要がある。

ない。関数を近似する際には、近似しやすい形状とそうではない形状が存在していることに注意しながら利用する必要がある。



(a) 価値関数：$V_{T-1}(k^i)$　　(b) 線形補間とスプライン補間

図 1: 内挿法と外挿法

MATLAB 関数

- `interp1`：1 次データの内挿・外挿を計算する関数。線形補間及びスプライン補間にも対応している。グリッドが 2 次元 (例えば、資本が 2 種類あるモデル) の場合には `interp2` が用意されている。

3.3　計算結果

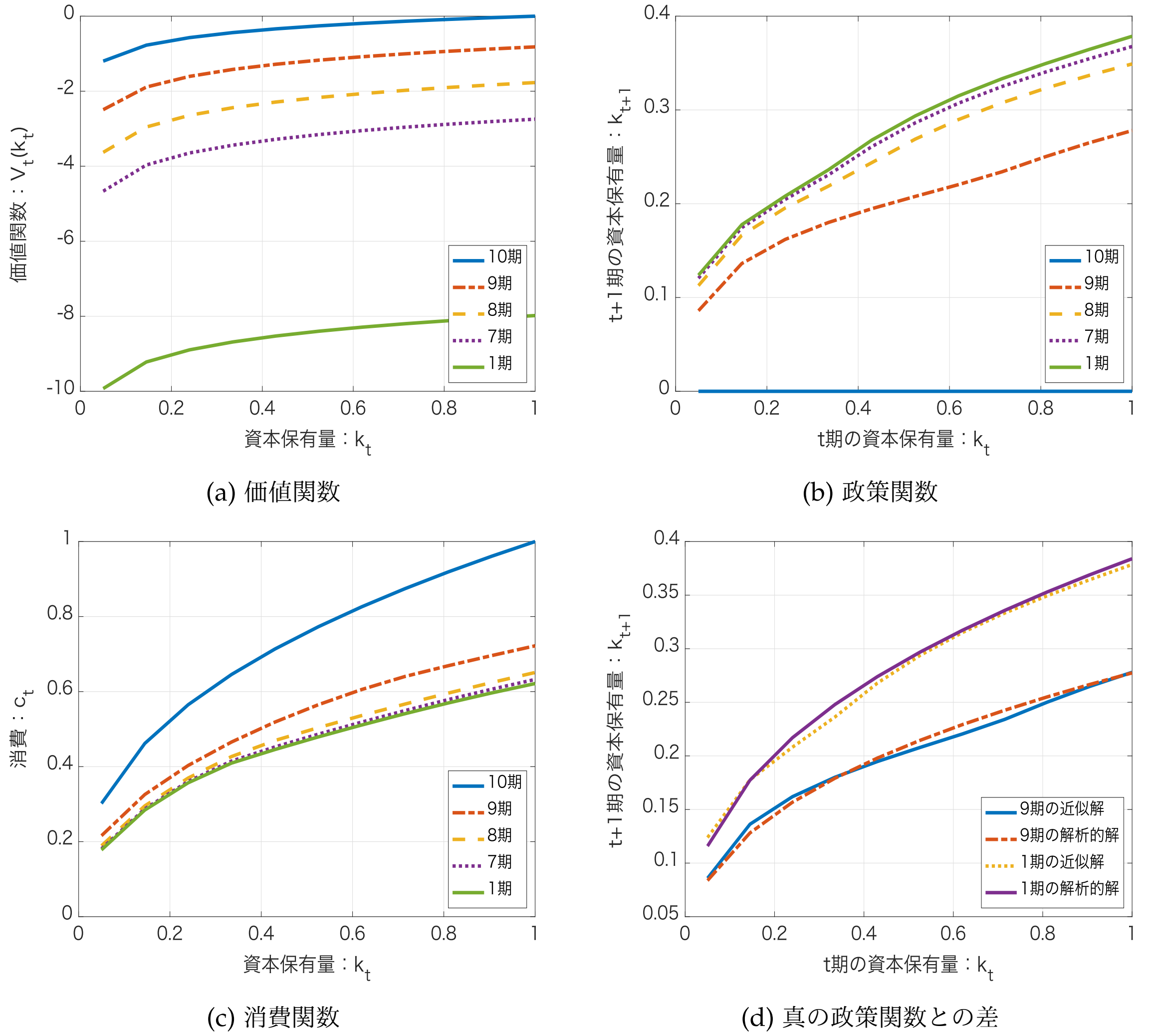
内挿法と前回学んだ最適化を組み合わせることによって、グリッド上で (5) 式を解くことが出来るようになる。後は、計算を繰り返し $t = 0$ 期までさかのぼっていけば良い。

図 2 が実際に価値関数と政策関数を計算した結果である。グリッドの数がわずか 11 個のため滑らかな曲線には見えないが、図 2(a) の通り、価値関数は凹関数であり、効用関数の性質を引き継いでいる。同じ資本保有量でも、残存期間に応じて最適な貯蓄額は異なるため、政策関数は時間の経過とともに下方シフトしていく事が読み取れる (図 2(b))。また図 2(c) の通り、消費関数も凹関数の性質を有しており、ここから消費者行動の分析につなげることも可能である。

有限 T 期間のロビンソン・クルーソーモデルには解析的解が存在している。

$$k_{t+1} = \alpha\beta \frac{1-(\alpha\beta)^{T-t}}{1-(\alpha\beta)^{T-t+1}} k_t^{\alpha},\ t = 0, 1, \ldots, T-1.$$

図 2(d) は解析的に得られる真の政策関数と比較したものであるが、グリッドが 11 個程度だと、多少の誤差が生じることが確認できる[7]。



(a) 価値関数　(b) 政策関数　(c) 消費関数　(d) 真の政策関数との差

図 2: ロビンソン・クルーソーモデル

[7] 前回同様に連載で使用したコードは全て GitHub 上に公開されている。読者はグリッドの数を増やすことによって真の値に近づくことを自ら確認してみてほしい。今回使用したコードは `https://github.com/keizai-seminar-quant-macro/chapter3` からダウンロード出来る。

MATLAB Code

- `main_robinson_crusoe.m`

4　無限期間モデル (新古典派成長モデル)

ロビンソン・クルーソーはいずれ無人島生活から脱出するので有限期間の経済活動を考えることが出来たが、われわれが生活する社会では明確な最終期は存在していない。あるいは、自分の世代から子供、孫、またその子供と続く王朝 (dynasty) 的な経済活動を考えるのであれば永遠に続く。そのため永続的な経済活動を考える必要がある。それが**無限期間モデル (infinite horizon model)** である。

上でセットアップした代表的個人 (ロビンソン・クルーソー) が存在する経済を無限期間に拡張しよう。

$$\max_{\{c_t,k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t u(c_t),$$
$$\text{subject to}$$
$$c_t + k_{t+1} = f(k_t) + (1-\delta)k_t,\ k_0 \text{ given.}$$

ただし、無限期間モデルなので目的関数が発散しないように $\beta \in (0,1)$ とする。

基本的な考え方は前と同じである。代表的個人は無限期間の消費から生じる自身の効用の割引現在価値を最大化するように、消費と貯蓄を選択する。ここでは一般的に、資本 k_t は生産に投下されて財 $(f(k_t))$ を生み出すが、δ の割合で減耗すると考える[8]。

これをベルマン方程式の形で書き換えると、

$$V(k) = \max_{k'}\{u(f(k) + (1-\delta)k - k') + \beta V(k')\}, \tag{6}$$

となる。

(6) 式はこれまでのベルマン方程式とほぼ同じ表記に見えるが、わずかに異なる点が見つけられるだろうか。先ほどは時間をあらわすサブスクリプト t が書いてあったが、それが消えて、次期の変数を k' とプライムを付けて表記している。有限期間モデルでは、「今が何期目か」あるいは「残存期間は何期間あるか」という情報が意思決定を考える上で重要なため、状態変数の一つとなっていた。また、価値関数も何期目かによって変わるため、各期毎に区別されていた。しかし、無限期

[8] ロビンソン・クルーソー経済では、種は蒔くとなくなってしまうので、資本減耗率 100% を暗黙のうちに仮定していた。

間モデルの場合には、意思決定に必要な情報は状態変数 k に集約されており、時間の情報を明示的に書く必要はなくなる[9]。言い換えると、経済主体の意思決定を**再帰的 (recursive)** に書き直したのである。

5　無限期間モデルの解法

5.1　ベルマン方程式の特徴付け

ロビンソン・クルーソー経済では、価値関数を繰り返し計算してどこに行きつくかを気にする必要はなかった。しかし、無限期間モデルの場合には数学的な取り扱いに注意をする必要がある。

動的計画法及びその数値計算を正しく使うためには、理論的な理解が重要になってくる。しかし、そのためには数学的な準備が必要になるため、ここではベルマン方程式の重要な性質について、直観的な結果だけ証明なしで紹介する。動的計画法について詳細な理解が得たいのであれば、Stokey et al. (1989) にじっくりと取り組むことをお勧めする。

(6) 式で定義されるベンチマークモデルの解法は、前節で紹介した後ろ向き帰納法と似ている。それが、**価値関数反復法 (value function iteration: VFI)** である。価値関数の性質及び VFI には昔から多くの研究の蓄積があり、その長所と短所もはっきりしている。以下がその特徴である。

有限期間モデルの場合には後ろ向きに計算することで、全期間の価値関数を計算することが出来た。(6) 式では明示的な終わりはないが、同じことが出来る。任意に価値関数 $V^{(0)}$ を推測 (initial guess) する。そして、

$$V^{(1)}(k) = \max_{k'}\{u(c) + \beta V^{(0)}(k')\}$$

を計算する。その後、得られた $V^{(1)}(k)$ を右辺に代入して、同じようにベルマン方程式を解くのである。一定の条件下において、この繰り返し計算が真の価値関数に**収束 (converge)** する事は、**縮小写像の性質 (contraction mapping property)** として知られている。 すなわち、真の価値関数の形状を知らなくても、繰り返し計算によって安定的に真の値に近い値を計算できるのである。

多くの数値計算手法では、実際にもっともらしい値が得られていても本当に真の値に収束しているのか、理論的にきちんと確認出来ない場合が多い。 その中で、VFI は理論的に収束する事が確認されていることから、信頼性がある手法である。VFI の縮小写像の性質は、モデルが複雑であってもそのモデルをベルマン方程式の形にすることが出来れば、数値的に分析可能にしてくれるとい

[9] あらゆるモデルがこのような表現が出来るわけではない。定式化によっては、過去の歴史 (history) の経路が状態変数となり、現在の経済を定める上で重要になることがある。その場合には、単純な再帰的表現は出来なくなる。

う意味で極めて強い性質である。一方で、VFI の収束スピードは遅いことが知られており、何らかのスピードアップの工夫がなければ、真の解を得るためには時間がかかる。

5.2 数値計算

価値関数を VFI によって求める際に用いられる方法は、大きく分けて 2 種類がある。

1. 状態変数と操作変数を共に離散化して解く方法
2. 状態変数は離散個で近似をして、操作変数のみ自由な値を認める方法

どちらも前回 2 期間モデルで学習した数値計算法の応用である。

5.2.1 状態変数と操作変数を離散化する

まず、状態空間と操作変数が共に離散的な場合の VFI からスタートしよう。このアプローチは、十分な精度で計算をしようとすると非常に遅くコンピュータのメモリも大量に消費するが、安定しており、強い非線形性があるモデルでも扱うことが可能な点が長所である[10]。

ベンチマークモデルは、$u(c) = \log c$、$f(k) = k^{\alpha}$、$\delta = 1$ と特定化した場合、次のような解析的解が存在していることが知られている。

$$
\begin{aligned}
V(k) &= A + B \log k, \\
A &= (1-\beta)^{-1}\left[\ln(1-\alpha\beta) + \frac{\alpha\beta}{1-\alpha\beta}\ln\alpha\beta\right],\ B = \frac{\alpha}{1-\alpha\beta}, \\
k' &= \alpha\beta k^{\alpha}.
\end{aligned}
$$

本来、解析的解が存在しているモデルであれば、わざわざ数値計算を使って近似的に解く必要はない。ここでは、近似的な計算によって得られた解と真の解がどの程度近い (離れている) かを確認するために、あえて解析的解が存在するモデルで近似計算を行っている。

まず、状態変数 k および操作変数 k' を離散 I 個に区切る ($k^i \in \{k^1, ..., k^I\}$)。資本 k の状態は離散個に区切った値しかとらないとする。グリッドを設定する際に、状態変数の上限 $\overline{k} = k^I$ 及び下限 $\underline{k} = k^1$ を設定する必要がある。下限については、モデルによっては借り入れ制約などが考えられるが、我々のモデルでは資本がなくては生産が行えなくなってしまうので、正値とする ($k^1 > 0$)。上限の設定は、場合によっては試行錯誤が必要になる。理論的な上限がある場合は良いが、経済成

[10] 最近の論文でこのアプローチをそのまま使っている研究はあまりないかもしれないが、簡単なモデルを始めに解いてみる手法としては有用である。Rust (1997) は状態空間を小さくする事で次元の呪い (curse of dimensionality) を緩和する方法を提案している。

長モデルなどでは明示的な上限は必ずしも存在しない。我々のモデルの場合、後で図 3(b) で示すように、政策関数と 45 度線が交わる点で傾きが 45 度より緩やかとなる。これは、45 度線との交点より右の領域では、時間とともに資本は減少していくことを意味しており、交点が定常状態となる。そのため、45 度線との交点の右側で最大値を設定した。

以下、アルゴリズムの概観である。

■アルゴリズム

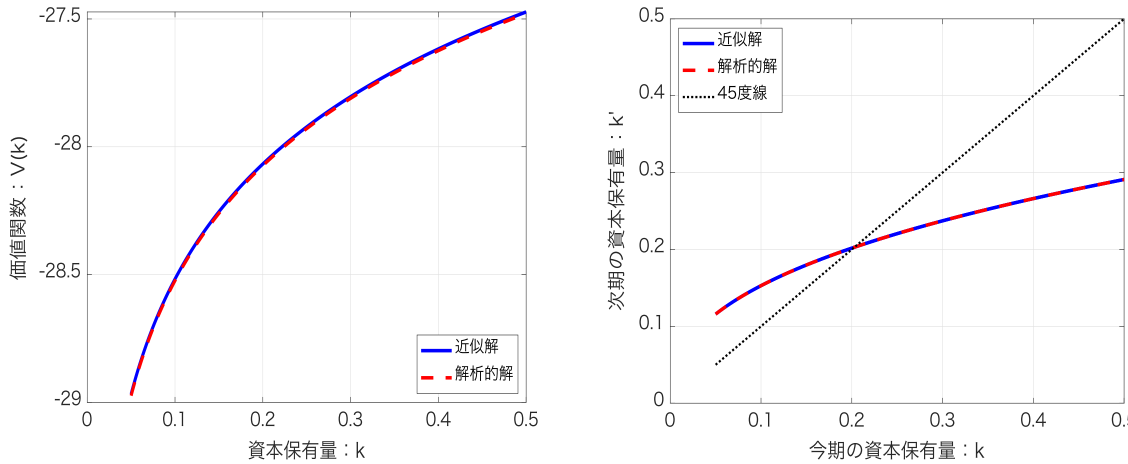
1. **グリッド生成**　状態空間及び操作変数を有限個のグリッドに区切る。今回は 10001 個とした。グリッド上における価値関数 $V^{(0)}(k^i)$ の初期値を当て推量 (initial guess) する。前述の通り、価値関数は収束するので、始めは全て 0 からスタートしても問題ないし、よりよい初期値を知っている場合にはそれを利用する。
2. **収束の基準**　収束の基準になるパラメータ ε を与えておく。どのような値にするかはモデルによるが、あまり緩くすると問題を解いたことにならないし、厳しくしすぎるといつまでたっても収束しない。今回は $\varepsilon = 10{-}5$ とした。
3. **効用関数を設定**　効用関数 $u([k^i]^\alpha + (1-\delta)k^i - k^j)$ を状態変数と操作変数の全グリッド上で評価する。よって、$I \times I$ の行列になる。
4. **価値関数の組み合わせを計算**　各 k_i について

$$V^{(1)}(k^i) = u([k^i]^\alpha + (1-\delta)k^i - k^j) + \beta V^{(0)}(k^j),$$

　　を計算する。右辺第 2 項の $V^{(0)}(k^j)$ は、初期値ではステップ 1 で与えた値を使う。
5. **価値関数を最大にする貯蓄を探す**　各 k^i について、$V^{(1)}(k^i)$ を最大にする k^j を探す。
6. **収束しているか確認**　古い価値関数 $V^{(0)}$ と新しい価値関数 $V^{(1)}$ の距離を測る。全ての k^i について $||V^{(0)}(k^i) - V^{(1)}(k^i)|| < \varepsilon$ であればストップ。そうでなければ、$V^{(1)}$ を $V^{(0)}$ に代入して、ステップ 4 とステップ 5 を繰り返す。縮小写像の性質があるため、いずれ収束する。

それでは、実際の数値計算結果を確認していこう。図 3(a) は、近似した価値関数と解析的解から計算した真の価値関数である。高い精度で真の関数を近似出来ている事が見て取れる。図 3(b) は同様に近似した政策関数と真の政策関数を比較している。目では見分けが付かないほど近いことから、高い精度で近似計算がうまくいっていることが確認できる。政策関数は 0.2 近辺で 45 度線と交わる事から、0.2 を上回る資本ストック k からスタートするとどんどん資本は減っていく事になる。そのため、極端に大きな値をとる必要はないが、グリッドの最大値は 0.2 を十分に上回る値に

設定している。



(a) 価値関数　　(b) 政策関数

図 3: 操作変数が離散的な場合の価値関数と政策関数

MATLAB Code

- `main_ddp.m`

5.2.2　操作変数が連続な場合

上記の方法は、状態変数と操作変数の両方を離散化しているため、今回のベンチマークモデルのようなシンプルなモデルであれば問題ないが、モデルを拡張する場合には計算速度と精度、さらにはコンピュータのメモリサイズの点からより洗練された手法が必要になる。そこで、状態変数のみ離散近似をして、操作変数については連続的な値を取る事を許容し、状態変数の間の値については内挿法で埋める事を考えよう。

以下は、操作変数が連続な場合の動的計画法の解法である[11]。

■アルゴリズム

1. **グリッド生成**　状態空間の評価点を有限個のグリッドに区切る。今回は 21 個とする。上と同様に、有限個のグリッド上における価値関数の値 $V^{(0)}(k^i)$ の初期値を当て推量する。価値関数は収束するので、やはり始めは 0 からスタートしても大丈夫であるが、例えば、離散近似でラフに計算した価値関数を初期値として採用するなど工夫をすると、収束が速くなる。

[11] 例えば、Cai and Judd (2014) を参照せよ。

2. **収束の基準**　収束の基準になるパラメータ ε を与える。

3. **近似・評価**　近似点 k^i 上にない価値関数の値については、線形近似や多項式近似、スプライン補間などを使って近似する必要があるため、その係数を計算する。$V(k;\mathbf{b})$ をパラメータ $\mathbf{b}$(例えば、線形近似の場合はグリッド間の傾き) を使って近似した時の、k 上での価値関数の値とする。

4. **最適化:**　各 k^i について、

$$V^{(1)}(k^i) = \max\{u([k^i]^\alpha + (1-\delta)k^i - k') + \beta V^{(0)}(k';\mathbf{b})\},$$

を計算する。価値関数を最大にする k' を探すためには、各言語に備わっている最適化関数を利用する。このステップで新しい価値関数 $\{V^{(1)}(k^i)\}$ 及び政策関数 $g(k^i)$ を得る。

5. **収束しているか確認**　古い価値関数 $V^{(0)}$ と新しい価値関数 $V^{(1)}$ の距離を測る。あらゆる k^i について $||V^{(0)}(k^i) - V^{(1)}(k^i)|| < \varepsilon$ であればストップ。そうでなければ、$V^{(1)}$ を $V^{(0)}$ に代入して、ステップ 3 とステップ 4 を繰り返す。十分に価値関数の**繰り返し誤差 (iteration error)** が小さくなったらそこでストップする。

操作変数が連続的な場合に VFI によって得られた価値関数及び政策関数は、図 3(a)、(b) と見た目はほぼ一緒になるため割愛する。図 4(a) は、収束しているか否かを確認する際に必要になる繰り返し計算誤差の値をプロットしたものである。およそ 200 回で繰り返し計算誤差は、$\varepsilon = 10^{-5}$ 以下になる。注目して欲しいのは、価値関数の繰り返し計算誤差より政策関数の繰り返し計算誤差の方が早く 0 に近づいている点である。これは偶然ではない。

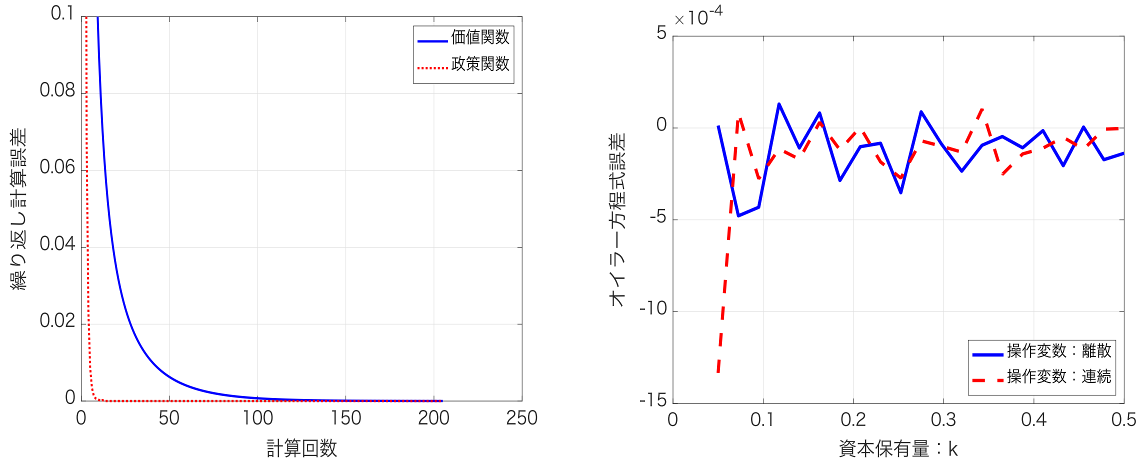
価値関数は収束するというありがたい性質があるが、その収束速度は遅いことが知られている。我々が知りたいのは価値関数の値ではなく、政策関数である事が多いので、政策関数が十分に収束していることを基準としてもよい。また、VFI ではなく政策関数を用いた繰り返し計算法である**政策関数反復法 (policy function iteration)** というアプローチも存在しており、一定の精度を保ったまま計算速度を速める工夫が数多く存在している。

5.2.3　計算誤差

図 4(b) は得られた政策関数 $k' = g(k^i)$ をオイラー方程式 $u'(c) = \beta u'(f(k') - g(k'))f'(k')$ に代入して、誤差

$$\frac{\beta u'([g(k^i)]^\alpha - g(g(k^i)))f'(g(k^i))}{u'([k^i]^\alpha - g(k^i))} \tag{7}$$

を計算したものである[12]。真の政策関数ではオイラー方程式が等式で成立するため、(7) 式は 0 になるはずであるが、近似によって誤差が生じている。また、操作変数を離散化した場合はグリッドを 10001 個に区切ったのに対して、操作変数が連続の場合にはわずか 21 個で同程度の誤差となる。計算時間も、筆者のノートパソコンでは離散化の場合が 60 秒程度の計算時間を要するのに対して、操作変数を連続にした場合には 17 秒程度で計算が完了する。このように、数値計算手法の選択によって計算時間と精度が変わってくる。



(a) 繰り返し計算誤差　　(b) オイラー方程式に代入した際の誤差

図 4: 操作変数が連続的な場合の価値関数の繰り返し誤差とオイラー方程式の誤差

MATLAB Code

- `main_ndp.m`、`BellmanEq.m`

6　進んだトピック

動的計画法は様々な方面で高い応用性を持っている。例えば、De Nardi et al. (2016) は米国の医療費支出問題について、高齢者の意思決定をベルマン方程式の形で定式化して、構造推計の中に組み込んで豊かな政策的含意を導き出している。また、Gabaix (2016) は行動経済学に基づいたマクロ経済モデルを動的計画法で解く方法を提案するなど、現在も動的計画法を用いた研究は様々な方面に広がりをみせている。動的計画法の解法に関する進展については、Cai and Judd (2014) や Maliar and Maliar (2014) が詳しい。

[12] 離散化の誤差については、連続の場合と同じ 21 個のグリッド上で誤差を計算している。

参考文献

Bertsekas, Dimitri P. (2012) *Dynamic Programming and Optimal Control*, Vol. 2: Athena Scientific.

——— (2017) *Dynamic Programming and Optimal Control*, Vol. 1: Athena Scientific.

Cai, Yongyang and Kenneth L. Judd (2014) "Advances in Numerical Dynamic Programming and New Applications," in Schmedders, Karl and Kenneth L. Judd eds. *Handbook of Computational Economics*, Vol. 3: Elsevier, Chap. 8, pp. 470-516.

De Nardi, Mariacristina, Eric French, and John Bailey Jones (2016) "Medicaid Insurance in Old Age," *American Economic Review*, Vol. 106, No. 11, pp. 3480-3520.

Gabaix, Xavier (2016) "Behavioral Macroeconomics via Sparse Dynamic Programming," NBER Working Paper No. 21848.

Maliar, Lilia and Serguei Maliar (2014) "Numerical Methods for Large Scale Dynamic Economic Models," in Schmedders, Karl and Kenneth L. Judd eds. *Handbook of Computational Economics*, Vol. 3: Elsevier, Chap. 3, pp. 325-477.

Rust, John (1997) "Using Randomization to Break the Curse of Dimensionality," *Econometrica*, Vol. 65, pp. 487-516.

Stokey, Nancy L., Robert E. Lucas Jr., and Edward C. Prescott (1989) *Recursive Methods in Economic Dynamics*: Harvard University Press.

Wickens, Michael (2012) *Macroeconomic Theory*: Princeton University Press, 2nd edition.

加藤涼 (2006) 『現代マクロ経済学講義動学的一般均衡モデル入門』, 東洋経済新報社.

齊藤誠 (2006) 『新しいマクロ経済学 [新版]』, 有斐閣.

廣瀬康生 (2012) 『DSGE モデルによるマクロ実証分析の方法』, 三菱経済研究所.