Spooky Boundaries at a Distance: Exploring Transversality and Stability with Deep Learning

Mahdi Ebrahimi Kahou* Jesús Fernández-Villaverde† Sebastián Gómez-Cardona*
Jesse Perla* Jan Rosa*

October 4, 2022

Abstract

In the long run we are all dead. Nevertheless, dynamic models require a variety of boundary conditions on long run forward-looking behavior such as transversality, no-bubble, and no-ponzi-scheme conditions. While these are essential to ensure the problems are well-posed and that short-term behavior is disciplined by long run expectations, they represent a computational challenge. Calculating the long run behavior or ergodic sets is typically the most expensive and unstable part of solving a model, and the global nature of spectral solutions makes it the central speed limit on increase dimensionality. We show numerically using a few classic models that with deep learning we can solve under-determined setups that automatically fulfill the appropriate long run boundaries conditions aligned with our models and economic assumptions. With this approach, we cannot efficiently calculate steady states and ergodic behavior. However, it allows us to accurately calculate short-term dynamics from a finite set of initial conditions which are still disciplined by long run boundaries. Not only are these solutions shockingly accurate with very little data and calculated in seconds, but they can even be used to solve transition dynamics with non-stationarity and steady-state multiplicity. While the computer science theory has not yet caught up with the empirical and numerical evidences to prove conditions under which these results would always hold, we provide an intuitive connection to the literature on "double-descent" which shows an implicit bias towards min-norm solutions in over-parameterized deep-learning models.

Keywords: Dynamic programming; dynamic models; deep learning; stationarity.

JEL codes: **something**

*University of British Columbia, Vancouver School of Economics; and †University of Pennsylvania.

‡ We would like to thank Marlon Azinovic, Jess Benhabib, Doga Bilgin, David Childers `Github` repo with code for this paper is at https://github.com.

1 Introduction

1.1 Overparameterization Inductive Bias/etc.

Introduce the idea of min-norm solutions, etc. in general. A few pointers to papers but largely point at the end section for the details.

Introduce "neural networks" and overparameterization.

2 Linear asset pricing

To clearly illustrate the connection between implicit bias of over-parameterized functions and forward-looking boundary conditions, we start by a linear asset pricing model in sequential form.

Setup: Consider an asset that pays a dividend stream of linear form

$$y(t+1) = c + (1+g)y(t) \tag{1}$$

$$y_0 \text{ given.} \tag{2}$$

where $c \geq 0$, $g \geq -1$ and $t = 0, \ldots, \infty$.

Although this problem is discrete in nature, we use a function notation (e.g. $y(t)$ instead of y_t). In order to solve sequential models of this sort we embed the discrete solution within a continuous function. For instance, we use deep neural networks to represent $p : \mathbb{R}_+ \to \mathbb{R}$ instead of $p : \mathbb{N} \to \mathbb{R}$. However, we still evaluate $p(t)$ at integer values. This approach is desirable for couple of reasons. First, from a practical perspective, deep neural networks are defined on a continuous domain. Second, it enables us to use sparse grids, which can be used to assess the interpolation and extrapolation powers of the solution.

The linear asset pricing problem can be written as a linear space model with

$$x(t+1) = Ax(t) \tag{3}$$

$$y(t) = Gx(t), \tag{4}$$

where

$$x(t) \equiv \begin{bmatrix} 1 & y(t) \end{bmatrix}^\top \tag{5}$$

$$A \equiv \begin{bmatrix} 1 & 0 \\ c & 1+g \end{bmatrix} \tag{6}$$

$$G \equiv \begin{bmatrix} 0 & 1 \end{bmatrix}. \tag{7}$$

The price based on the fundamentals, denoted by $p_f(t)$, can be written as the present discounted value of the dividend stream

$$p_f(t) \equiv \sum_{j=0}^{\infty} \beta^j y(t+j) = G\big(\mathbf{I} - \beta A\big)^{-1} x(t), \tag{8}$$

where β is the discount factor, and $\mathbf{I}$ is a two dimensional identity matrix. Here we assume $\beta \in (0,1)$ and $\beta(1+g) < 1$. The second assumption ensures the convergence of the infinite sum in equation (8). This problem can be written in a recursive way

$$p(t) = Gx(t) + \beta p(t+1) \tag{9}$$
$$x(t+1) = Ax(t) \tag{10}$$
$$x_0 \text{ given.} \tag{11}$$

Equation (9) can be interpreted as the price of owning an asset that pays a stream of dividends $\{y(t+j)\}_{j=0}^{\infty}$ is equal to the dividend the agent receives at time t plus the discounted price of owning the asset at time $t+1$. Although for a given initial condition x_0 the price based on the fundamentals $p_f(t)$ satisfies equations (9)-(11), this system of equations do not form a well-posed problem and there are infinitely many solutions of the form

$$p(t) = p_f(t) + \zeta \beta^{-t}. \tag{12}$$

The initial condition x_0, uniquely determines $p_f(t)$. However, ζ is not uniquely determined. In order to have a well-posed problem another condition is required. This is referred to as no-bubble condition and eliminates any solution that grows faster or at the same rate as β^{-t}. Therefore, the well-posed problem can be written as

$$p(t) = Gx(t) + \beta p(t+1) \tag{13}$$
$$x(t+1) = Ax(t) \tag{14}$$
$$0 = \lim_{T\to\infty} \beta^T p(T) \tag{15}$$
$$x_0 \text{ given,} \tag{16}$$

where equation (15) is the no-bubble condition.

2.1 Interpolating solution

As a generalization of the collocation approach we pick a parametric space of functions $\mathcal{H}(\Theta)$, where $\Theta \equiv \{\theta_1, \ldots, \theta_M\}$ represents the set of parameters. For instance, in a standard collocation

method $\mathcal{H}(\Theta)$ can be the space of Chebyshev polynomials of degree M. We pick a grid of points $\hat{X}$. These grid points are chosen from the domain of the function of interest X[1]. For instance, for sequential models including linear asset price model where the domain of the function of interest is time $X \equiv \mathbb{R}_+$. Therefore, the grid is a set of points representing time periods

$$\hat{X} \equiv \{t_1, \ldots, t_N\}. \tag{17}$$

Given a parametric space of functions $\mathcal{H}(\Theta)$, a grid $\hat{X}$, and a very large point in time T, one can find the approximate price function $p(t;\theta) \in \mathcal{H}(\Theta)$ by solving the following optimization problem

$$\min_{\theta\in\Theta} \frac{1}{|\hat{X}|} \sum_{t\in\hat{X}} \Big[p(t;\theta) - Gx(t) - \beta p(t+1;\theta)\Big]^2 + \underbrace{\Big[\beta^T p(T;\theta)\Big]^2}_{\text{Is this necessary?}}, \tag{18}$$

where $x(t)$ is evaluated by the law of motion for x

$$x(t) = A^t x_0 \quad \text{for } t \in \hat{X}. \tag{19}$$

The first term in this minimization represents the residuals of the recursive formulation for the prices (i.e., equation (13)) and the second term is a finite approximation of the no-bubble condition (i.e., equation (15)).

In standard collocation methods where the number of parameters is equal to the number of grid points, i.e., $M = N$, this problem can be solved exactly as a system of equations with a boundary condition at T.

Choice of $\hat{X}$: In high dimensional cases the choice of $\hat{X}$ is very crucial, however in this paper we focus on low dimensional cases. Although this choice is not very crucial for this study, we provide the results for different grids. In this paper we are interested in the generalization power of the solutions. Therefore, we also evaluate the solutions outside of $\hat{X}$. For instance, in sequential models we are interested in the behavior of the solutions for $t > t_N$.

Is the no-bubble condition necessary? As discussed before, without the no-bubble condition the linear asset pricing model has infinitely many solutions of the form expressed in equation (12). Each solution corresponds to a different value for ζ. Therefore, in the absence of the second term (finite approximation of the no-bubble condition), the optimization problem described in (18), has infinitely many solutions that achieve the zero of the objective function on $\hat{X}$.

In the next section we establish how using over-parameterized space of functions (number of parameters is larger than the number of grid points) leads to convergence to the solution based

[1]For instance, in Chebyshev collocation method X are Chebyshev nodes.

on the fundamentals without worrying about the no-bubble condition.

Therefore, using over-parameterized functions we can drop the approximate no-bubble condition and solve

$$\min_{\theta\in\Theta} \frac{1}{|\hat{X}|} \sum_{t\in\hat{X}} \Big[p(t;\theta) - Gx(t) - \beta p(t+1;\theta) \Big]^2 \tag{20}$$

to achieve the solution based on the fundamentals $p_f(t)$.

2.2 Minimum norm interpretation and no-bubble condition

In this paper we use an over-parameterized space functions (i.e., deep neural networks) to solve the optimization problem described in (20). An over-parameterized function $p(t;\theta)$ is characterized by M parameters $\{\theta_1, \ldots, \theta_M\}$ where M is larger then the number of grid points, i.e., $M > N$. In practice the number of parameters is very larger than the grid points. Since $M \gg N$ over-parameterized functions and their corresponding optimization algorithms provides exact interpolating solutions, which are defined as approximate solutions that obtain a zero value for the objective function.

However, since the number of parameters is larger than the number of grid points there are too many degrees of freedom. Therefore, there are many arrangements of the parameters that can achieve exact interpolation. It has been observed and in some cases (mostly linear setups) proved that the optimization algorithms used in over-parameterized interpolation problems have an implicit bias toward a specific class of functions. Recently, it has been proposed that this property can be interpreted as minimizing a new objective function on the parametric space of functions $\mathcal{H}(\Theta)$ subject to exact interpolation at all the points on the grid. Section 7 provides a comprehensive background and discussion on this robust phenomena in over-parameterized functional approximation. We call this minimum norm interpretation of an over-parameterized interpolation.

The minimum norm interpretation of the sequential linear asset pricing model described in equations (18)-(19), can be written as

$$\min_{p(\cdot;\theta)\in\mathcal{H}(\Theta)} \quad \|p(\cdot;\theta)\|_S \tag{21}$$

$$\text{s.t. } p(t;\theta) - Gx(t;\theta) - \beta p(t+1;\theta) = 0 \quad \text{for } t \in \hat{X} \tag{22}$$

$$0 = \lim_{T\to\infty} \beta^T p(T;\theta) \tag{23}$$

where $\mathcal{H}(\Theta)$ is an over-parameterized space of functions, $\hat{X}$ is a set of points in time (i.e., $\hat{X} = \{t_1, \ldots, t_N\}$), and $x(t)$ is evaluated by the exogenous law of motion $x(t+1) = Ax(t)$, given the initial condition x_0.

What is the this new objective function? The new objective function $\|\cdot\|_S : \mathcal{H}(\Theta) \to \mathbb{R}$ is a semi-norm[2] satisfying the following assumption

Assumption 1. *Let $\|\cdot\|_S : \mathcal{H} \to \mathbb{R}_+$ be a semi-norm, Ψ_1 and Ψ_2 be two differentiable functions from a compact space $\mathcal{X} \subset \mathbb{R}$ to $\mathbb{R}$ be such that*

$$\int_{\mathcal{X}} \left|\frac{d\Psi_1}{ds}\right|^2 ds > \int_{\mathcal{X}} \left|\frac{d\Psi_2}{ds}\right|^2 ds \tag{24}$$

then

$$\|\Psi_1\|_S > \|\Psi_2\|_S. \tag{25}$$

Moreover, since $\|\cdot\|_S$ is a semi-norm, it satisfies the triangle inequality

$$\|\Psi_1 + \Psi_2\|_S \leq \|\Psi_1\|_S + \|\Psi_2\|_S. \tag{26}$$

This is an assumption on the implicit bias of the over-parameterized interpolation and their optimization algorithms. Intuitively, this assumption states that among two interpolating solutions on a domain $\mathcal{X}$ the bias is toward the one with smaller derivatives. In other words, the bias is toward the smoother one. Through the rest of the paper we assume Assumption 1 holds. See Section 7 for a discussion of the validity of this assumption and recent advances in the statistics and optimization theory regarding understanding this assumption.

As shown in equation (12) any solution that violates the no-bubble condition is associated with a non-zero ζ. In the following proposition we establish that minimizing $\|\cdot\|_S$ automatically makes the no-bubble condition redundant. Intuitively, due to the explosive term β^{-t} $(0 < \beta < 1)$, all the solutions that contain a bubble term $(\zeta > 0)$ have bigger derivatives than the fundamental price $p_f(t)$. More formally for any compact interval of the form $[0, T]$

$$\int_0^T \left|\frac{dp}{dt}\right|^2 dt > \int_0^T \left|\frac{dp_f}{dt}\right|^2 dt \tag{27}$$

Since the dividends are positive we only focus on prices that are positive for all t, this excludes solutions with $\zeta < 0$.

Proposition 1. *Let $p(t)$ be a solution that satisfies equations (13) characterized by a $\zeta \geq 0$*

$$p(t) = p_f(t) + \zeta\beta^{-t} \tag{28}$$

[2] A semi-norm $\|\cdot\|_S$ is a mapping that for all functions ψ_1 and ψ_2 in $\mathcal{H}$ satisfies

1. Triangle inequality: $\|\psi_1 + \psi_2\|_S \leq \|\psi_1\|_S + \|\psi_2\|_S$, and
2. Absolute homogeneity: $\|\gamma\psi_1\|_S = |\gamma|\|\psi_1\|_S$ for all $\gamma \in \mathbb{R}$.

where $p_f(t)$ is defined by equation (8). Let $\|\cdot\|_S$ be a norm that satisfies Assumption 1, then

$$\underset{p\in\mathcal{H}}{\operatorname{argmin}} \|p\|_S = p_f. \tag{29}$$

In other words the minimum semi-norm solution is attained when $\zeta = 0$.

Proof. Since $\zeta \geq 0$, then by the triangle inequality and Assumption 1

$$\|p_f\|_S \leq \|p\|_S \leq \|p_f\|_S + \zeta\|\beta^{-t}\|_S. \tag{30}$$

□

This proposition establishes that the price based on the fundamentals has the lowest semi-norm.[3] In other words, the interpolating solutions of an over-parameterized functional approximation has a bias toward $p_f(t)$ on $\hat{X}$ and automatically satisfy the no-bubble condition. Therefore, the optimization problem described in (20) is enough to find the price based on the fundamentals without imposing any explicit regularity regarding the long run behavior.

2.3 Results

Figure 1 shows the result of the minimization problem described in equations (20) and (19) for $\beta = 0.9$, $c = 0.01$, $g = -0.1$, and $y_0 = 0.08$. In this experiment $\hat{X} = \{0, 1, 2, \ldots, 29\}$ and $\hat{X}_{\text{test}} = \{0, 1, 2, ..., 49\}$. It is worth mentioning that $\hat{X}$ is the grid and $\hat{X}_{\text{test}}$ is defined to assess the performance of the approximate solutions outside of $\hat{X}$.

We approximate the price $p(t;\theta)$ using a deep neural network with four hidden layers, each with 128 nodes. Each hidden layer uses Tanh and the output layer uses Sofplus as activation functions. The dashed vertical lines separate the interpolation from the extrapolation region. To be more specific, the interpolation region is the defied as $\hat{X}$ and the extrapolation as the points in $\hat{X}_{\text{test}}$ that are not in $\hat{X}$. The left panel shows the price paths. The price based on the fundamentals, denoted by $p_f(t)$, is calculated using equation (8), and $\hat{p}(t)$ shows the approximate solution. The right panel shows the relative error between the approximate and exact solution defined as

$$\varepsilon_p(t) \equiv \frac{\hat{p}(t) - p_f(t)}{p_f(t)} \quad \text{for } t \in \hat{X}_{\text{test}}. \tag{32}$$

[3] What is required in this proof is triangle inequality and an innocuous assumption that if $p(t) > p_f(t)$ for all $t \in \mathcal{X}$ then

$$\|p\|_S > \|p_f\|_S. \tag{31}$$

Therefore, Assumption 1 is strong for this proof and can be relaxed.

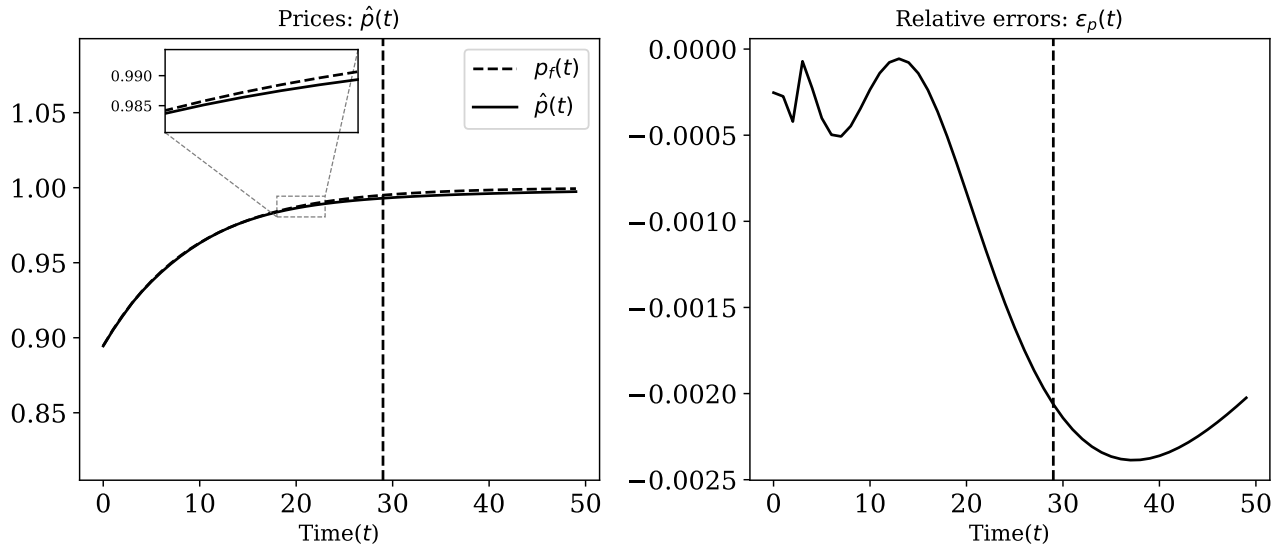

Figure 1: Comparison between the price based on the fundamentals and the price approximated by a deep neural network for the sequential linear asset pricing model. In the left panel the solid curve shows the approximate price and the dashed curve shows the price based on the fundamentals. The right panel shows the relative errors between the approximate solution and the price based on the fundamentals. The dashed vertical lines separate the interpolation from the extrapolation region.

The results of this experiment is multi-fold. First, we can achieve an accurate solution based on the fundamentals and eliminate bubble solutions simply by relying on the implicit bias of deep neural networks. Second, within the grid points $\hat{X}$, the short and medium run solution are accurate and it is not impaired by the long run errors (at most -0.05% relative error). Third, the extrapolation errors are very small which can be explained by the smoothness imposed by deep neural networks and the choice of $\hat{X}$. Here, we used a large time horizon in $\hat{X}$, i.e., $t_N = 29$. The prices in last few points of the grid are very close to its asymptotic value. Therefore, the deep neural network learns that for large values of t the price is almost a constant and stays very close to that value even outside of $\hat{X}$.

Contiguous vs. Sparse Grid There is a common belief that deep neural networks are only suitable for high-dimensional environments with an abundance of data (grid points). In this experiment we show that they can be very useful when there is a little amount of grid points.

Figure 2 shows the results of previous experiment with two different sparse grids for $\hat{X}$. The first grid contains 11 points and is defined as $\hat{X}(\text{Grid 1}) \equiv \{0, 1, 2, 4, 6, 8, 12, 16, 20, 24, 29\}$. The second grid contains 8 points and is defined as $\hat{X}(\text{Grid 2}) \equiv \{0, 1, 4, 8, 12, 16, 20, 24, 29\}$. The contiguous grid is the same as the previous experiment, $\hat{X} = \{0, 1, 2, \ldots, 29\}$ denoted by Contiguous. The top-left panel shows the price paths for these grids. $\hat{p}(t)$: Contiguous shows the approximate price path using $\hat{X}$, $\hat{p}(t)$: $\hat{X}$(Grid 1) shows the approximate price path using $\hat{X}$(Grid 1),

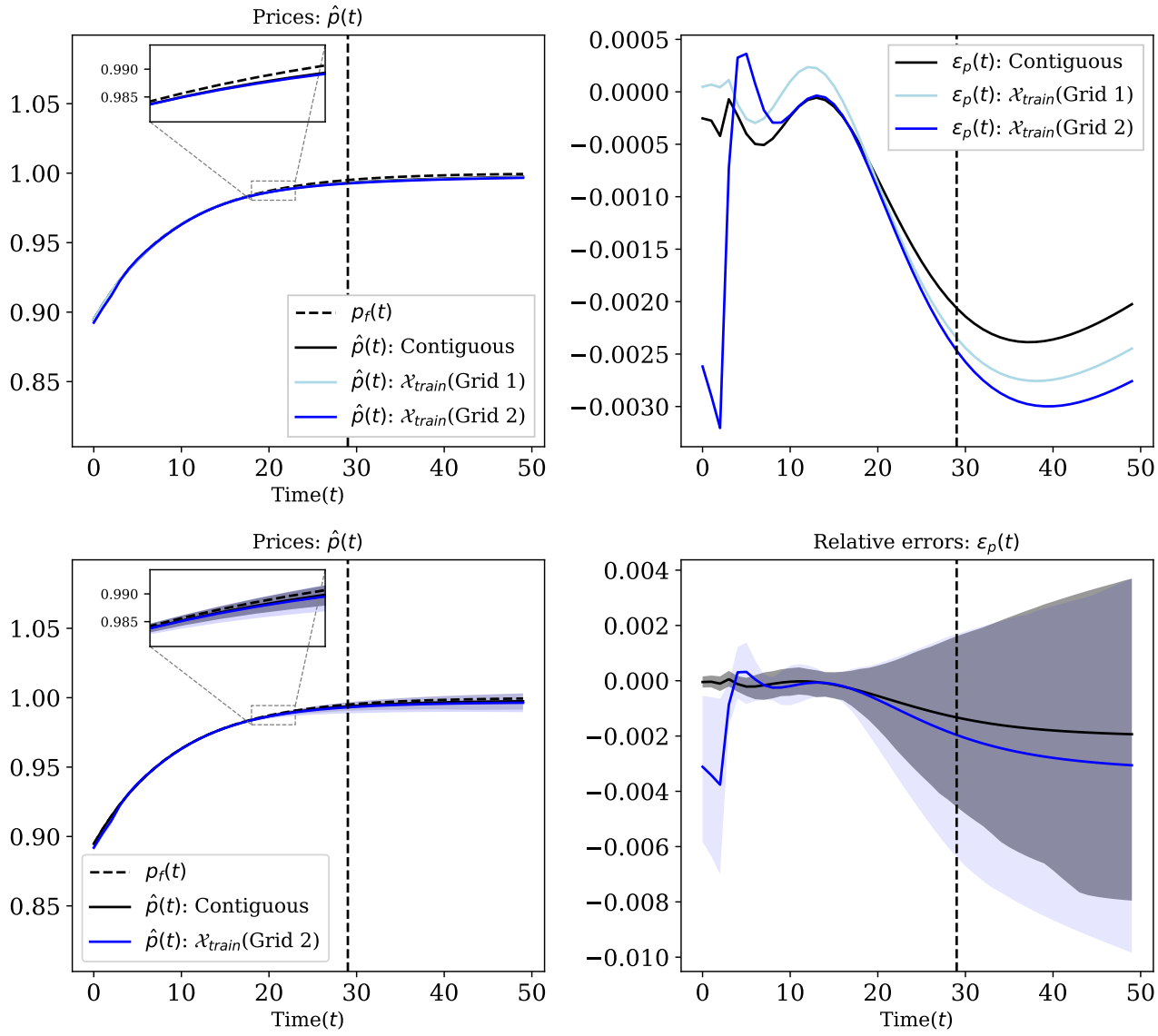

Figure 2: Comparison between the contiguous grid and two sparse grids for the sequential linear asset pricing model. The first sparse grid is defined as $\hat{X}(\text{Grid1}) \equiv \{0, 1, 2, 4, 6, 8, 12, 16, 20, 24, 29\}$ and the second sparse grid is defined as $\hat{X}(\text{Grid2}) \equiv \{0, 1, 4, 8, 12, 16, 20, 24, 29\}$. The top panels show the results for price path for one seed. The bottom panels show the median of the approximate prices and relative errors over 100 seeds for the contiguous grid and $\hat{X}(\text{Grid 2})$. The shaded regions show the 10th and 90th percentiles. The left panels provide a comparison between the approximate prices and the price based on the fundamentals . The right panels show the relative errors between the approximate prices and the price based on the fundamentals. The dashed vertical lines separate the interpolation from the extrapolation region.

and $\hat{p}(t) : \hat{X}(\text{Grid 2})$ shows the approximate price path using $\hat{X}(\text{Grid 2})$. The top-right panel shows the relative errors between the price based on the fundamentals and approximate solutions. $\varepsilon_p(t)$: Contiguous shows the relative errors for prices using contiguous grid, $\varepsilon_p(t) : \hat{X}(\text{Grid 1})$

shows the relative errors for prices using $\hat{X}$(Grid 1), and $\varepsilon_p(t)$: $\hat{X}$(Grid 2) shows the relative errors for prices using $\hat{X}$(Grid 2). In both top panels we use only one random initialization of the deep neural network (one seed) to generate the results. The solid curves in the bottom-left panel shows the median of price paths over 100 seeds for the contiguous grid and $\hat{X}$(Grid 2), the shaded regions show the 10th and 90th percentiles of approximate prices. The solid curves in the bottom-right panel shows the median of relative errors over 100 seeds for the contiguous grid and $\hat{X}$(Grid 2), the shaded regions show the 10th and 90th percentiles of the relative errors. The dashed vertical lines separate the interpolation from the extrapolation region.

These results show that the contiguous grid outperforms both sparse grids. However, the results for both sparse grids are very accurate (at most relative error of -0.35 %). As expected, the most sparse grid has higher relative errors. It is worth mentioning that convergence to the price based on the fundamentals is not sensitive to the gird size. Therefore, it can be a promising avenue for methods of adaptive sample selection in high-dimensional problems. These results also show the robustness to random initialization of the deep neural networks in the optimization process for both contiguous and sparse grid. This confirms the implicit bias in deep neural networks, which in this case the bias is toward the price based on the fundamentals.

Positive growth in dividends ($g > 0$) : The last two experiments assume that the dividends follow a stationary process. Here we show that exploiting a priori economic knowledge combined with the implicit bias, our method can deal with cases where no stationary solution exists. In this setup we know the price based on the fundamentals grows exponentially, this economic knowledge can be flexibly implemented in the design of the function space $\mathcal{H}(\Theta)$. In the case of $g > 0$ and $c = 0$, the fundamental price grows with the rate $1+g$. In this experiment we use this information to construct an approximating function of the form

$$\hat{p}(t;\theta) = e^{\phi t} NN(t;\theta_1) \tag{33}$$

where $\theta \equiv \{\phi, \theta_1\}$, $NN(\cdot;\theta_1)$ is a deep neural network, and ϕ is a single parameter needs to be found in the optimization process.

Figure 3 shows the results for sequential linear asset pricing model with growing dividends (i.e., $c = 0$, $g = 0.02$). We use the same deep neural network we utilize in the first experiment. The only difference is the additional exponential term. In the left panel, the solid curve, denoted by $\hat{p}(t)$, shows the median of approximate price paths over 100 seeds, the dashed curve, denoted by $p_f(t)$ shows the solution based on the fundamentals. The shaded regions show the 10th and 90th percentiles for prices. The right panel shows the median of relative errors between the approximate solutions and the price based on the fundamentals over 100 seeds, and the shaded regions show the 10th and 90th percentiles for relative errors. The dashed vertical lines separate the interpolation from the extrapolation region.

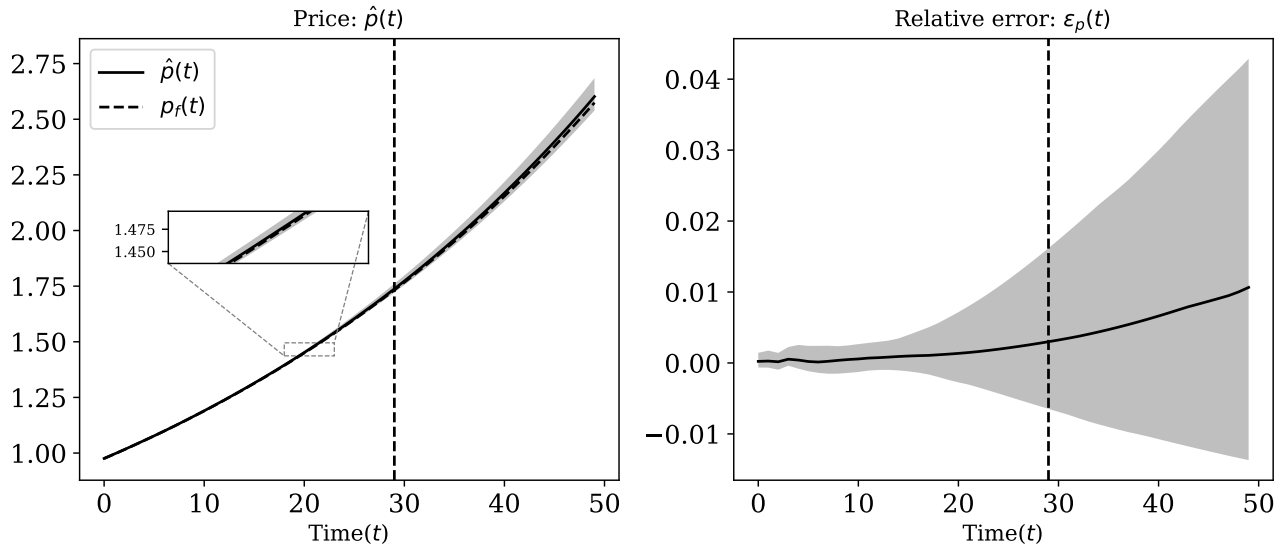

Figure 3: Comparison between the price based on the fundamentals and the price approximated by a deep neural network for the sequential linear asset pricing model with growing dividends ($g = 0.02$). The solid curve in the left panel shows the median of the approximate price paths over 100 seeds, the dashed curve shows the price based on the fundamentals. The solid curve in the right panel shows the median of the relative errors between the approximate price and the price based on the fundamentals over 100 seeds. The shaded regions show 10th and 90th percentiles. The dashed vertical lines separate the interpolation from the extrapolation region.

These results show that the long run errors do not impair the short and medium run accuracy even in the presence of growing dividends. It is worth noting if interpolation is the object of interest (left of the dashed lines) the same result can be obtained without building the exponential term in the approximating function. However, if extrapolation (right of the dashed lines) is the object of interest, using a economic intuition can dramatically enhance the generalization power (less than 0.5% relative error after 20 periods). Moreover, the algorithm learns the growth rate accurately. See Figure 19 and the discussion in Appendix A.1 for a detailed treatment of the approximated growth rate.

Here we used the correct information about the functional form of the growth in the price path. However, in more complicated dividend processes, when the growth in prices is unknown, it is possible to misspecify the functional form of the growth. Figure 20 in Appendix A.2 confirms that even in the presence of functional misspecification long run errors do not impair the accuracy of short and medium run dynamics.

3 Neoclassical growth model: Sequential form

Another form of forward-looking boundary condition at infinity that appears frequently in dynamic economic models is the transversality condition. In these models the transversality con-

dition is a necessary condition for optimality (see Ekeland and Scheinkman (1986), and Kamihigashi (2005)). In this section we focus on the sequential version of the neoclassical growth model.

Setup: Consider an agent maximizing discounted utility of consumption $\sum_{t=0}^{\infty}\beta^t u\big(c(t)\big)$. The agent has access to a production technology $z^{1-\alpha}f(k)$, where z is the total factor productivity and k is the capital. Capital depreciates at rate $\delta\in[0,1]$. The agent's optimization problem can be written as

$$\max_{c(t),k(t+1)} \sum_{t=0}^{\infty}\beta^t u\big(c(t)\big) \tag{34}$$

$$\text{s.t. } k(t+1)=z(t)^{1-\alpha}f\big(k(t)\big)+(1-\delta)k(t)-c(t) \tag{35}$$

$$z(t+1)=(1+g)z(t) \tag{36}$$

$$k(t)\geq 0 \tag{37}$$

$$0=\lim_{T\to\infty}\beta^T u'\big(c(T)\big)k(T+1) \tag{38}$$

$$k_0, z_0 \text{ given }. \tag{39}$$

The problem described above is the standard neoclassical growth problem. Here we focus on the constant relative risk aversion utilities $u(c)=\frac{c^{1-\sigma}}{1-\sigma}$, and production function of the form $f(k)=k^{\alpha}$. The first order conditions for this problem can be written as

$$u'\big(c(t)\big)=\beta u'\big(c(t+1)\big)\big[z(t+1)^{1-\alpha}f'\big(k(t+1)\big)+(1-\delta)\big] \tag{40}$$

$$k(t+1)=f\big(k(t)\big)+(1-\delta)k(t)-c(t) \tag{41}$$

where the first equation is the Euler equation and the second one is the feasibility condition. Equations (40) and (41) do not form a well-posed problem. There are infinitely many solutions $\{c(t),k(t)\}_{t=0}^{\infty}$ that satisfy these equations. The transversality condition, described in (38), uniquely determines the optimal solution.

Saddle path nature of the optimal solution: The optimal solution of this problem is a saddle path and the steady states of the optimal solution is a saddle point. The saddle path nature of the optimal solution makes the optimal path $\{k(t),c(t)\}$ very sensitive to the choice of the initial consumption (i.e. $c(0)$). Therefore, a small error in the choice of consumption at time zero can lead to a solution that violate the transversality condition.

It is also worth noting that if an economy start with zero capital (i.e. $k=0$) this economy

stays at zero level of capital forever, therefore, $k = 0$ is a fixed point of this economy. However, it is a repulsive fixed point. Therefore, for an economy that starts with any non-zero level of capital there is no optimal path that takes the economy to $k = 0$. See the discussion and Figure 21 in Appendix B.1 for analysis of our solution method for very small levels for initial conditions of capital.

3.1 Interpolating solution

As a generalization of the collocation approach we pick a parametric space of function $\mathcal{H}(\Theta)$, where $\Theta \equiv \{\theta_1, \ldots, \theta_M\}$ represent the set of parameters. We pick a grid of points $\hat{X}$. Since this is a sequential model, similar to the linear asset pricing model $\hat{X} \equiv \{t_1, \ldots, t_N\}$. The Euler equation (i.e., equation (40)), the feasibility condition (i.e., equation (41)), the law of motion for total factor productivity (i.e., equation (36)), and the transversality condition (i.e., (38)) combined with the initial conditions for capital and productivity form a system of equations.

For a given capital function $k(t;\theta) \in \mathcal{H}(\Theta)$ we define the consumption function $c(t; k(\cdot;\theta))$ via the feasibility condition

$$c(t; k(\cdot;\theta)) \equiv f\big(k(t;\theta)\big) + (1-\delta)k(t;\theta) - k(t+1;\theta). \tag{42}$$

Given a parametric space of functions $\mathcal{H}(\Theta)$, a grid $\hat{X}$, and a very large point in time T, one can find the approximate capital function $k(t;\theta) \in \mathcal{H}(\Theta)$ by solving the following optimization problem

$$\min_{\theta\in\Theta} \frac{1}{|\hat{X}|} \sum_{t\in\hat{X}} \left[\frac{u'\big(c(t;k(\cdot;\theta))\big)}{u'\big(c(t+1;k(\cdot;\theta))\big)} - \beta\big[z(t+1)^{1-\alpha} f'\big(k(t+1;\theta)\big) + (1-\delta)\big] \right]^2 + \big(k(0;\theta) - k_0\big)^2 + \underbrace{\left[\beta^T u'\Big(c\big(T;k(\cdot;\theta)\big)\Big) k(T+1;\theta)\right]^2}_{\text{Is this necessary?}} \tag{43}$$

where $z(t)$ is evaluated by the law of motion for z

$$z(t) = (1+g)^t z_0 \quad \text{for } t \in \hat{X}. \tag{44}$$

The first term in the optimization problem represents the Euler residuals and the second term is the initial condition residuals. The second term ensures that the capital function at time zero matches the initial level of capital k_0. The third term is a finite approximation of the transversality condition.

In standard collocation methods where the number of parameters is equal to the number of grid points, (i.e, $M = N$) this problem can be solved exactly as a system of equations with a boundary condition at T. It is also possible to approximate both capital and consumption functions simultaneously. See Appendix B.3 for more details.

Choice of $\hat{X}$: Similar to the case of linear asset pricing model, the choice of $\hat{X}$ is not very crucial. However, we provide the results for different grids. We also evaluate the optimal approximate policy function for capital $\hat{k}(\cdot;\theta)$ outside of $\hat{X}$ to study the generalization performance of the solutions.

Where is the transversality condition? Without the transversality condition the neoclassical growth problem has infinitely many solutions. Equations (36),(40), and (41) form a a three dimensional dynamical system with two initial conditions k_0 and z_0[4]. Therefore, in the absence of the third term (finite approximation of the transversality condition) the optimization problem described in (43) has infinitely many solutions that achieve the zero of the objective function on $\hat{X}$.

In Section 3.3 we establish how using over-parameterized space of functions (when $M \gg N$) and their corresponding optimization processes lead to convergence to the solution that does not violate the transversality condition. Intuitively, the solutions that violate the transversality condition have bigger derivatives than the optimal solution. Therefore, using over-parameterized functions we can drop the transversality condition and solve

$$\min_{\theta\in\Theta} \frac{1}{|\hat{X}|}\sum_{t\in\hat{X}} \lambda_1 \left[\frac{u'\big(c(t;k(\cdot;\theta))\big)}{u'\big(c(t+1;k(\cdot;\theta))\big)} - \beta\big[z(t+1)^{1-\alpha} f'\big(k(t+1;\theta)\big) + (1-\delta)\big]\right]^2 + \lambda_2\big(k(0;\theta) - k_0\big)^2 \tag{45}$$

to obtain the optimal policy for capital which does not violate the transversality condition.

The constants λ_1 and λ_2 are a pair of positive parameters which determines the importance of each residuals in the optimization process.[5] The non-negativity of capital, $k(t) \geq 0$, is built into $\mathcal{H}(\Theta)$.

[4]Given the initial level of capital k_0, factor productivity z_0, and an arbitrary c_0 for the initial level of consumption we can generate a set of solutions $\{\tilde{k}(t), \tilde{c}(t), z(t)\}$ by iterating forward the Euler equation, feasibility condition, and the law of motion for total factor productivity. By construction this set of solutions is a solution for this system of equations. Therefore, there are infinitely many solutions each characterized by a different c_0.

[5]One can use the convex combination of the residuals (i.e. $\lambda_1 + \lambda_2 = 1$). In the sequential neoclassical growth model, the results are not sensitive to the choice of these parameters so we can choose equally weighted residuals (i.e., $\lambda_1 = \lambda_2 = \frac{1}{2}$).

3.2 Results

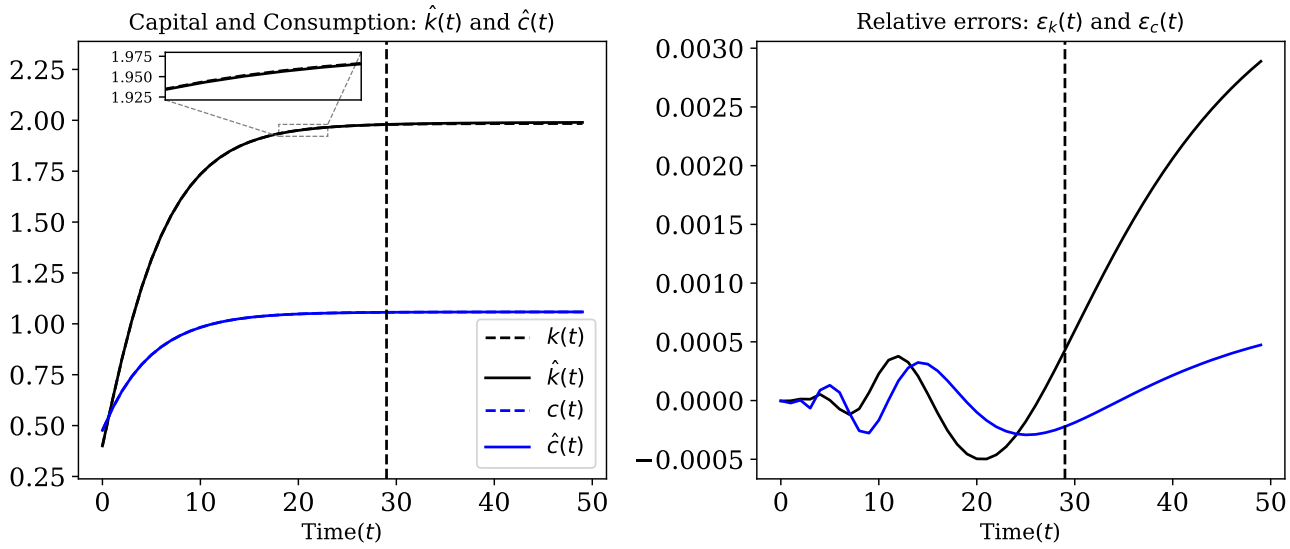

Figure 4: Comparison between the value function iteration and approximate solution using a deep neural network for the sequential neoclassical growth model. In the left panel the black solid line shows the approximate capital path and the blue solid line shows the consumption path. The dashed curves show the capital and consumption path obtained by the value function iteration method. The right panel shows the relative errors for both capital and consumption path. The dashed vertical lines separate the interpolation from the extrapolation region.

Figure 4 shows the result of the minimization problem described in (45) for $\beta = 0.9$, $\alpha = 0.33$, $\delta = 0.1$, $g = 0$, $k_0 = 0.4$, and $z_0 = 1$. In this experiment $\hat{X} = \{0, 1, 2, \ldots, 29\}$ and $\hat{X}_{\text{test}} = \{0, 1, 2, ..., 49\}$. We approximate the capital path $k(t; \theta)$ using a deep neural network with four hidden layers, each with 128 nodes. Each hidden layer uses Tanh and the output layer uses Softplus as activation functions. The dashed vertical lines separate the interpolation from the extrapolation region. The left panel shows the capital and consumption paths. The benchmark solution is calculated using value function iteration method and are denoted by $k(t)$ and $c(t)$.[6] The approximate solutions are denoted by $\hat{k}(t)$, and $\hat{c}(t)$. The right panel shows the relative error between the approximate and the value function iteration solutions defined as

$$\varepsilon_c(t) \equiv \frac{\hat{c}(t) - c(t)}{c(t)} \quad \text{for } t \in \hat{X}_{\text{test}} \tag{46}$$

$$\varepsilon_k(t) \equiv \frac{\hat{k}(t) - k(t)}{k(t)} \quad \text{for } t \in \hat{X}_{\text{test}}. \tag{47}$$

[6]The value function iteration is also an approximation method, here we assume that value function iteration method yields more accurate results than our method.

The results of this experiment is multi-fold. First, we can achieve an accurate solutions that do not violate the transversality condition without imposing any long run constraint. Second, the short and medium run solution are accurate and they are not impaired by the long run errors (at most -0.05% relative error for capital). Third, the extrapolation errors are very small which can be explained by the smoothness imposed by deep neural networks and the choice of $\hat{X}$. Here, we used a large time horizon in $\hat{X}$, i.e., $t_N = 29$. The capital path in last few points of the grid are very close to the steady state. Therefore, the deep neural network learns that for large values of t the capital is almost a constant and stays very close to that value even outside of $\hat{X}$.

Contiguous vs. Sparse Grid Figure 5 shows the result of the previous experiment for two different sparse grid for $\hat{X}$. The first grid contains 11 points and is defined as $\hat{X}(\text{Grid 1}) \equiv \{0, 1, 2, 4, 6, 8, 12, 16, 20, 24, 29\}$. The second grid contains 8 points and is defined as $\hat{X}(\text{Grid 2}) \equiv \{0, 1, 4, 8, 12, 16, 20, 24, 29\}$. The contiguous grid is the same as the previous experiment $\hat{X} = \{0, 1, 2, \ldots, 29\}$, denotey by Contiguous. The dashed vertical lines separate the interpolation from the extrapolation region. The top-left panel shows the approximate capital paths for theses grids versus the value function iteration solution, denoted by $k(t)$. The top-right panel shows the relative errors between the approximate solution and the value function iteration solution for these capital paths. In both top panels we use only one random initialization of the deep neural network (one seed) to generate the results. The bottom-left panel shows the capital paths obtained by value function iteration method, denoted by $k(t)$ and the median of approximate capital path over 100 seeds for the contiguous grid, $\hat{X}(\text{Grid 2})$, the shaded regions show the 10th and 90th percentiles. The bottom-right panel shows the median of relative errors between the approximate and value function errors solution over 100 seeds for the contiguous grid and $\hat{X}(\text{Grid 2})$, the shaded regions show the 10th and 90th percentiles.

This results show that the contiguous grid outperforms both sparse grids. However, the results for both sparse grids are very accurate (at most 0.01 %). As expected the most sparse grid has higher relative errors. It is worth mentioning that convergence to the optimal solution is not sensitive to the gird size. Therefore, it can be a promising avenue for adaptive optimization and sample selection in high-dimensional sequential problems. These results also show the robustness to random initialization of the deep neural networks in the optimization process for both contiguous and sparse grid. This confirms the implicit bias in deep neural networks, which in this case the bias is toward the optimal path for capital.

Far from the steady state: In the previous experiment we established that the we can achieve accurate short and medium run accuracy on sparse grids. However, in both grids we used a large time horizon (i.e., $t_N = 29$). After 29 periods the capital and consumption paths are very close to their steady states. In this experiment we study whether the approximate solution has accurate short run dynamics when using medium time horizon for $\hat{X}$ (e.g., $t_N \approx 10$).

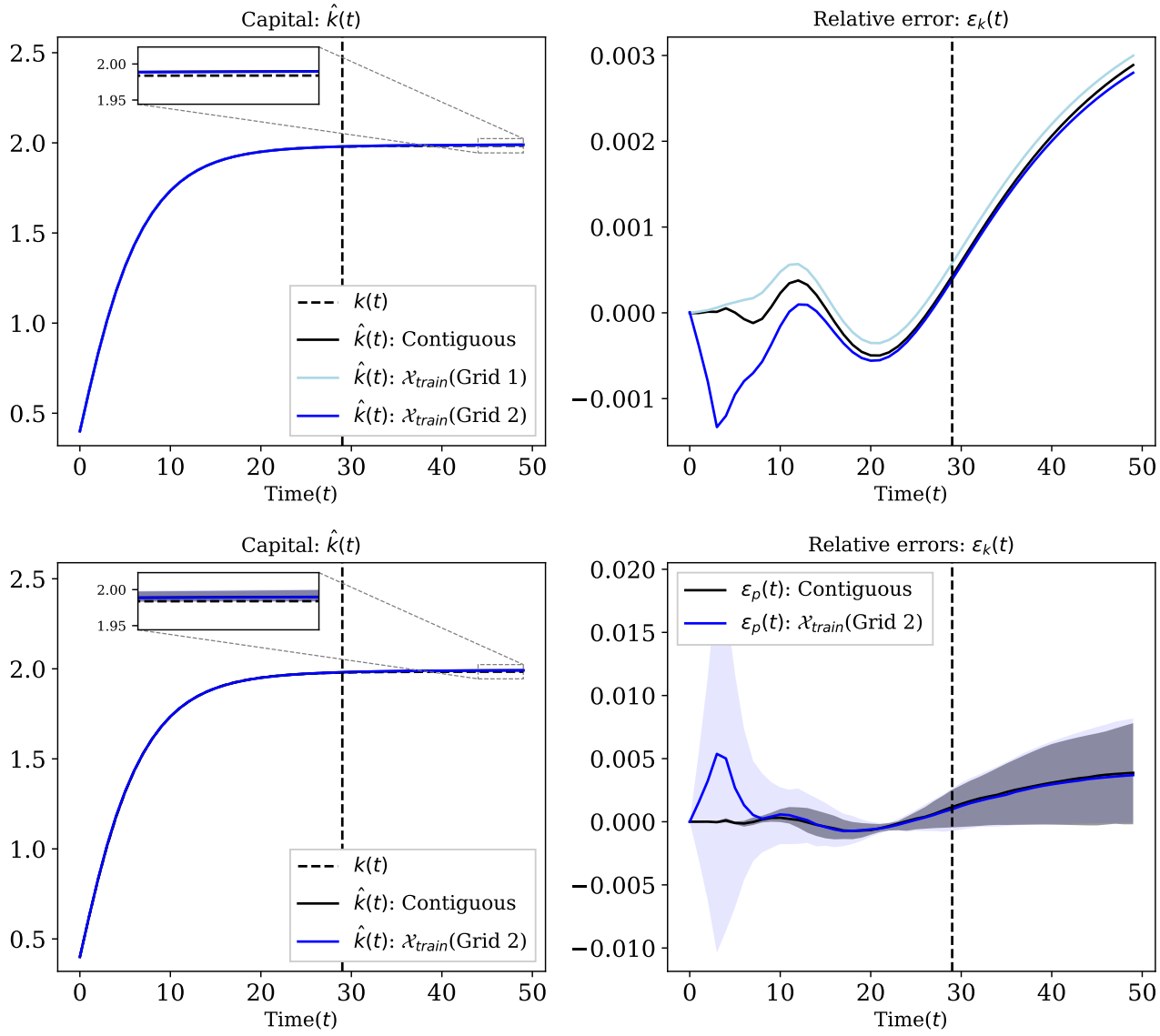

Figure 5: Comparison of between the value function iteration and approximate solution using a deep neural network for the sequential neoclassical growth model. The first sparse grid is defined as $\hat{X}(\text{Grid 1}) \equiv \{0, 1, 2, 4, 6, 8, 12, 16, 20, 24, 29\}$ and the second sparse grid is defined as $\hat{X}(\text{Grid 2}) \equiv \{0, 1, 4, 8, 12, 16, 20, 24, 29\}$. The top panels show the results for capital path and relative errors for $\hat{X}(\text{Grid 1})$, $\hat{X}(\text{Grid 2})$, and the contiguous grid for one seed. The bottom panels show the results for capital paths and relative errors for $\hat{X}(\text{Grid 1})$, $\hat{X}(\text{Grid 2})$, and the contiguous grid for 100 seeds. The shaded regions show the 10th and 90th percentiles and solid curves show the medians over 100 seeds. The dashed curves show the capital and consumption paths obtained by the value function iteration method. The dashed vertical line separates the interpolation from the extrapolation region.

Figure 6 shows the results for the sequential neoclassical growth model with $\hat{X} = \{0, 1, \ldots, 9\}$. We used the same parameters and deep neural network as the previous experiments. The dashed vertical lines separate the interpolation from the extrapolation region. The solid curve in the

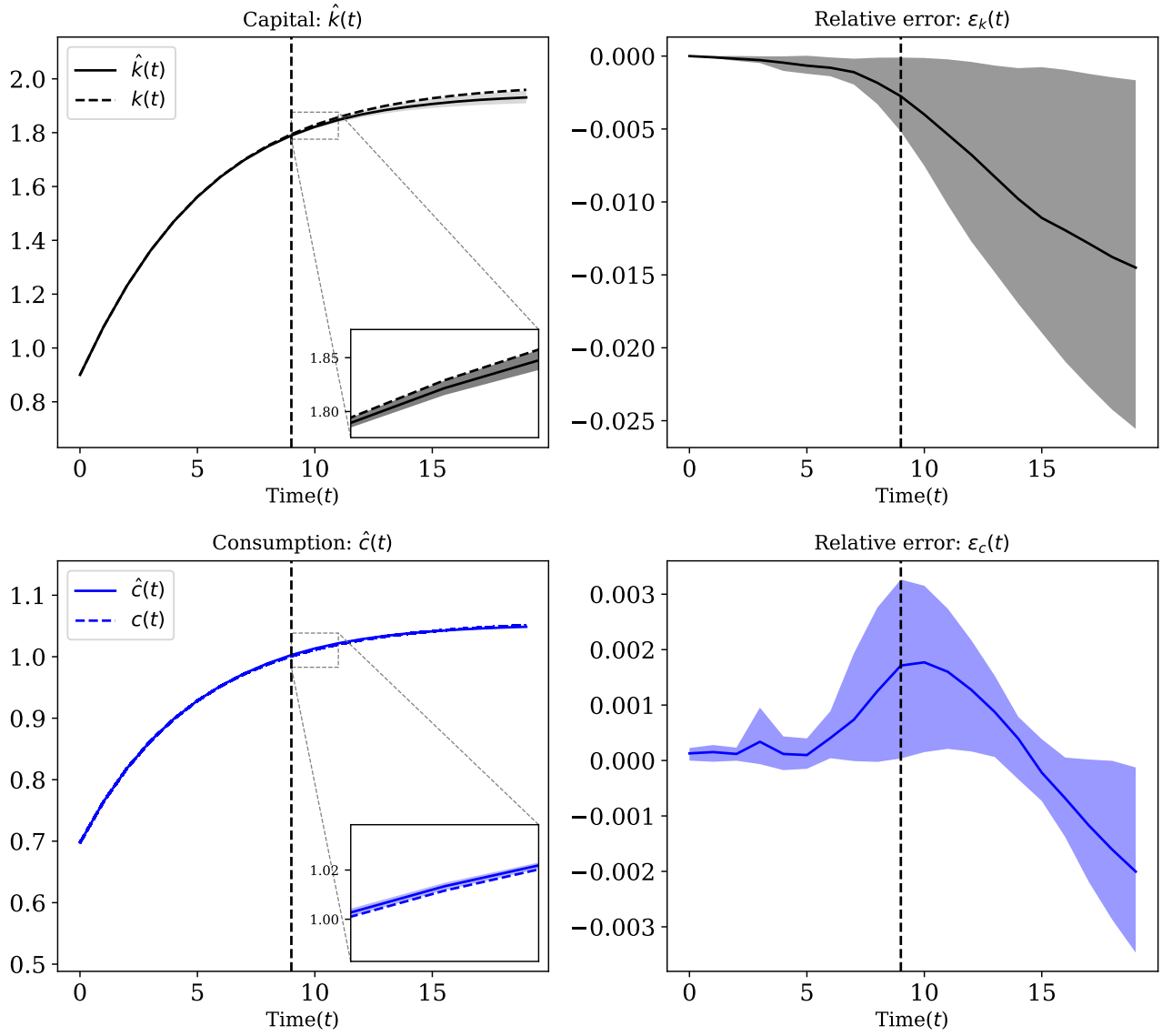

Figure 6: Comparison of between the value function iteration and approximate solution for sequential neoclassical growth model with medium time horizons (i.e., $\hat{X} = \{0, 1, \ldots, 9\}$). The solid curves in the left panels show the median of the approximate capital and consumption paths over 100 seeds and the shaded regions show the 10th and 90th percentiles. The dashed curves show the capital and consumption paths obtained by the value function iteration method. The solid curves in the right panels show the median of the relative errors between approximate solutions and solutions obtained by the value function iteration method for capital and consumption paths. The shaded regions show the 10th and 90th percentiles. The dashed vertical lines separate the interpolation from the extrapolation region.

top-left panel shows the median of capital paths over 100 different random initialization of the parameters of the deep neural network (seeds), the dashed curve shows the capital path obtained by the value function iteration method, and the shaded areas show the 10th and 90th percentiles. The solid curve in top-right panel shows the median of relative errors between the approximate

capital paths and the capital path obtained by the value function iteration method, the shaded areas show the 10th and 90th percentiles. The solid curve in the bottom-left panel shows the median of consumption paths over 100 different random seeds, the dashed curve shows the consumption path obtained by the value function iteration method, and the shaded areas show the 10th and 90th percentiles. The dashed curve in bottom-right panel shows the median of relative errors between the approximate consumption paths and the consumption path obtained by the value function iteration method, the shaded areas show the 10th and 90th percentiles.

These results show that we obtain solutions that do not violate the transversality condition even when we use medium time horizons in $\hat{X}$. Moreover, the long run errors do not impair the accuracy of short run dynamics (less than 0.5% after 5 periods). Therefore, we get accurate solutions when the capital and consumption paths are far from the steady state. This result is robust to using very short time horizon in $\hat{X}$ (i.e., $t_N = 4$). See Figure 23 and the discussion in Appendix B.4 for more details.

Balanced growth path ($g > 0$) In the last experiments we assumed that the total factor productivity is constant and stationary. Here we show that exploiting a priori economic knowledge combined with the implicit bias of deep neural networks, our method can deal with models where no stationary solution exists. Since the production function is homogeneous of degree one, in the long run the capital path has a growth rate of g. This economic knowledge can be implemented flexibly in the design of the function space $\mathcal{H}(\Theta)$. In this experiment we construct an approximating function of the following form

$$\hat{k}(t;\theta) = e^{\phi t} NN(t;\theta_1) \tag{48}$$

where $\theta \equiv \{\phi, \theta_1\}$, $NN(\cdot;\theta_1)$ is a deep neural network, and ϕ is a single parameter needs to be found in the optimization process.

Figure 7 shows the results for sequential neoclassical growth for non-stationary total factor productivity (i.e., $g = 0.02$). We use the same neural network as before for $NN(\cdot;\theta_1)$ and use the original grid, $\hat{X} = \{0, 1, 2, \ldots, 29\}$. The only difference is the additional exponential term. The solid curve in the top-left panel shows the median of approximate capital paths over 100 different initialization of the parameters of the deep neural network (seeds) and the dashed curve shows the capital paths obtained by the value function iteration method, and the shaded area shows the 10th and 90th percentiles. The solid curve in the top-right panel shows the median of the relative errors between the approximate capital paths and the capital paths obtained by the value function iteration method, the shaded area shows the 10th and 90th percentiles. The solid curve in the bottom-left panel shows the median of consumption paths over 100 different random seeds, the dashed curve shows the consumption path obtained by the value function iteration method, and the shaded area shows the 10th and 90th percentiles. The dashed curve in bottom-right

panel shows the median of relative errors between the approximate consumption paths and the consumption path obtained by the value function iteration method, the shaded area shows the 10th and 90th percentiles.

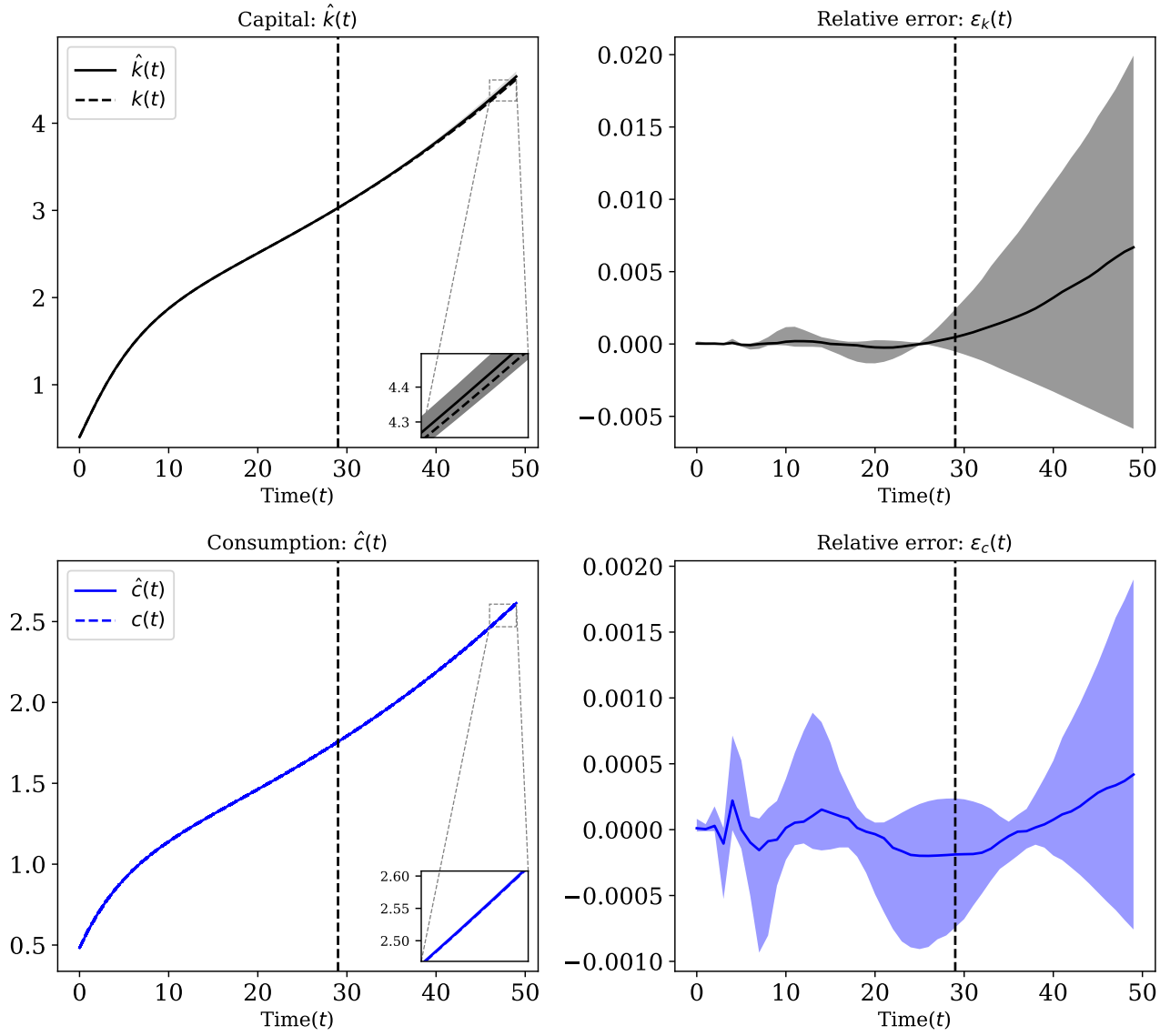

Figure 7: Comparison between the value function iteration and approximate solution using a deep neural network for the sequential neoclassical growth model with growth in total factor productivity (i.e., $g = 0.02$). The solid curves in the left panels show the median of the approximate capital and consumption paths over 100 seeds and the shaded regions show the 10th and 90 percentiles. The dashed curves show the capital and consumption paths obtained by the value function iteration method. The solid curves in the right panels show the median of the relative errors between approximate solutions and solutions obtained by the value function iteration method for capital and consumption paths. The shaded regions show the 10th and 90th percentiles. The dashed vertical lines separate the interpolation from the extrapolation region.

These results show that the long run errors do not impair the short and medium run accuracy

even in the presence of balanced growth path. It is worth noting if interpolation is the object of interest (left of the dashed line) the same result can be obtained without building the exponential term in the approximating function. However, if extrapolation (right of the dashed line) is the object of interest, as shown in this results, using economic intuition can dramatically enhance the generalization power, (less than 0.03% relative error in capital after 49 periods).

Here we used the correct information about the functional form of the growth in capital and consumption paths. However, in cases where the production function is not homogeneous of degree one or the total factor productivity follows a non-linear process, it is possible to misspecify the functional form of the growth. Figure 24 in Appendix B.5 confirms that even in the presence of functional misspecification long run errors do not impair the accuracy of the short and medium run dynamics. Moreover, the results shows that the solutions generalize well in the extrapolation region.

3.3 Minimum norm interpretation and transversality condition

In this section we provide the connection between minimum norm interpretation and the transversality condition. We focus on the case of zero total factor productivity growth (i.e., $g = 0$ and $z = 1$).

The solutions that satisfy the Euler equation and feasibility condition (i.e. equations (40)-(41)) and violate the transversality can be characterized by their asymptotic behavior. Let $\{\tilde{c}(t), \tilde{k}(t)\}_{t=0}^{\infty}$ be a set of solutions that violate the transversality condition, then

$$\lim_{t\to\infty} \tilde{c}(t) = 0 \tag{49}$$

$$\lim_{t\to\infty} \tilde{k}(t) = \tilde{k}_{\max}, \tag{50}$$

where $\tilde{k}_{\max}$ solves

$$\delta\tilde{k}_{\max} = f\left(\tilde{k}_{\max}\right). \tag{51}$$

Since the production function f is increasing and strictly concave $\tilde{k}_{\max}$ exists and is unique. Moreover it is the stable fixed point for a dynamical system described by $k(t+1) = f\big(k(t)\big) + (1-\delta)k(t)$. For $f(k) = k^{\alpha}$

$$\tilde{k}_{\max} = \delta^{\frac{1}{\alpha-1}}. \tag{52}$$

Comparing $\tilde{k}_{\max}$ with the steady state of the capital $k^* \equiv \left(\frac{\beta^{-1}+\delta-1}{\alpha}\right)^{\frac{1}{\alpha-1}}$, one can establish that

$$\tilde{k}_{\max} > k^*. \tag{53}$$

Figure 8 shows a set of solutions (blue curve) for capital, consumption and marginal utility of consumption (shadow prices) denoted by $\tilde{k}(t), \tilde{c}(t)$, and $u'(\tilde{c}(t))$ that satisfy the Euler equation

and feasibility condition, but violate the transversality condition for the case of $k_0 = 0.4$, $\beta = 0.9$, $\alpha = 0.33$, and $\delta = 0.1$. The black curves denoted by $k(t)$, $c(t)$, and $u'(c(t))$ show the optimal paths for capital, consumption and marginal utility of consumption which satisfy the Euler equation, feasibility condition and the transversality condition. The steady states for capital and consumption in the optimal solution are denoted by k^* and c^*. It is important to note that for this set of parameters $\frac{\tilde{k}_{\max}}{k^*} \approx 15$.

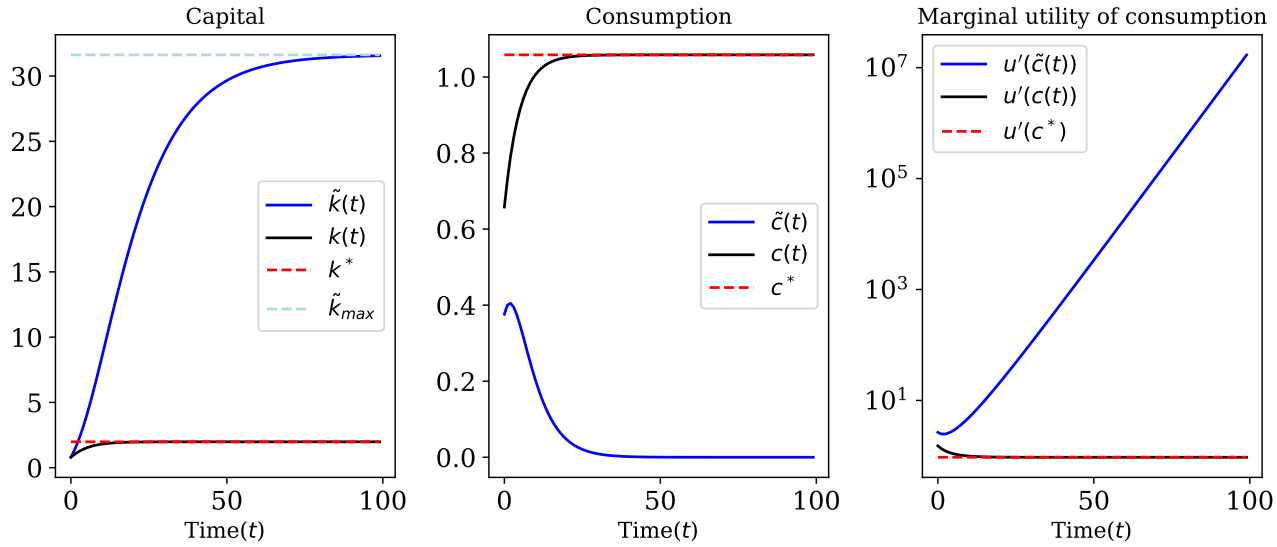

Figure 8: Comparison between the optimal solution and the solutions that violate the transversality condition for $k_0 < k^*$. The blue curves, denoted by $\tilde{k}(t)$, $\tilde{c}(t)$, and $u'(\tilde{c}(t))$, show a set of capital, consumption, and marginal utility of consumption paths that violate the transversality condition. The black curves denoted by $k(t)$, $c(t)$, and $u'(c(t))$ show the capital, consumption, and marginal utility of consumption paths for the optimal solution. The steady states for capital an consumption are denoted by k^* and c^*.

The minimum norm interpretation of the minimization problem described in (43) can be written as

$$\min_{k(\cdot;\theta)\in\mathcal{H}(\Theta)} \quad \|k(\cdot;\theta)\|_S \tag{54}$$

$$\text{s.t.} \quad u'\Big(c\big(t;k(\cdot;\theta)\big)\Big) = \beta u'\Big(c\big(t+1;k(\cdot;\theta)\big)\Big)\Big[f'\big(k(t+1;\theta)\big)+1-\delta\Big] \quad \text{for } t\in\hat{X} \tag{55}$$

$$k(0) = k_0 \tag{56}$$

$$0 = \lim_{T\to\infty} \beta^T u'(c(T;k(\cdot;\theta)))k(T+1), \tag{57}$$

where $\mathcal{H}(\Theta)$ is an over-parameterized space of functions, $\hat{X}$ is a set of points in time (i.e., $\hat{X} = \{t_1,\ldots,t_N\}$), the semi-norm $\|\cdot\|_S$ satisfies Assumption 1, and consumption function $c\big((t;k(\cdot;\theta)\big)$ is defined in equation (42).

As evident in Figure 8, for the case of $k_0 < k^*$, any capital sequence that starts from k_0 and violates the transversality condition converges to $\tilde{k}_{\max}$. Both capital paths $\tilde{k}(t)$ and $k(t)$ increase monotonically. Starting from k_0, $\tilde{k}(t)$ must have bigger derivatives than $k(t)$ to reach $\tilde{k}_{\max}$. Therefore, the solutions that violate the transversality condition have bigger gradients. More formally, let $\tilde{k}(t)$ be a capital path violating the transversality condition and $k(t)$ be the optimal solution, then in a compact space of the form $[0, T]$

$$\int_0^T \left|\frac{d\tilde{k}}{dt}\right|^2 dt > \int_{\mathcal{X}} \left|\frac{dk}{dt}\right|^2 dt. \tag{58}$$

Therefore, by Assumption 1

$$\|k\|_S < \|\tilde{k}\|_S. \tag{59}$$

The solutions that violate the transversality condition have bigger semi-norms. In other words, the interpolating solutions of an over-parameterized space of functions has a bias toward $k(t)$ on $\hat{X}$ and automatically satisfy the transversality condition. Therefore, the optimization described in (45) is enough to find the optimal solution without imposing any explicit regularity regarding the long run behavior.

It is important to note that when the transversality condition is violated the marginal utility of consumption (shadow price) grows boundlessly (order of 10^7 after 100 periods). Alternatively, one can use an over-parameterized function to approximate the shadow price. Since the optimal solution has bounded shadow prices, it is easy to establish that[7]

$$\|u'(c)\|_S < \|u'(\tilde{c})\|_S. \tag{60}$$

Since the difference between the derivatives of marginal utility of consumption for the solutions that violate the transversality condition and the optimal solution is much larger than the difference between the derivatives of their corresponding capital paths, it is easier for the over-parameterized functions to detect the optimal solution. Therefore, in practice approximating the marginal utility can lead to faster convergence to the optimal solution.

In this argument we focus on cases where the initial value of capital is below the steady state. For cases where the initial condition is above the steady state the same argument can be made, see Figure 22 and the discussion in Appendix B.2.

[7]For the case of shadow prices Assumption 1 can be relaxed. Because for all the semi-norms (and norms) we are aware of, an unbounded function has a bigger semi-norm than a bounded function.

4 Neoclassical growth model with multiplicity of equilibria: sequential form

The sequential neoclassical growth model discussed and studied in Section 3 has only one saddle fixed point characterized by (k^*, c^*), while the solutions that violate the transversality condition approach a fixed point $(k, c) = (\tilde{k}_{\max}, 0)$. Given that $\tilde{k}_{\max} > k^*$, and for regular parameters $\tilde{k}_{\max}$ is extremely larger that k^*, the semi-norm of the capital path that converges to k^* is substantially smaller than those that violate the transversality condition. Therefore, it is straightforward for an over-parameterized function and the optimization algorithms to distinguish the difference between these two class of solutions and pick the optimal solution path. However, a natural question that arise is: can the implicit bias of an over-parameterized functions distinguish the optimal path in the presence multiplicity of steady states that are fairly close to each other?

In this section we address this question by studying the sequential neoclassical growth problem with a convex-concave production function of the form

$$f(k) = a \max\{k^\alpha, b_1 k^\alpha - b_2\} \tag{61}$$

where a, b_1, b_2 a are positive constants and $b_1 > 1$. [8] This production function is continuous, however, has a kink at $\left(\frac{b_2}{b_1 - 1}\right)^{\frac{1}{\alpha}}$. We embed this production function in the sequential setup of neoclassical growth as described in Section 3. Specifically, we solve the optimization illustrated in (45) with this convex-concave production function. In this experiment we focus on constant total factor productivity (i.e., $z_0 = 1$ and $g = 0$)[9].

This problem has two steady states for capital

$$k_1^* = \left(\frac{\beta^{-1} + \delta - 1}{a\alpha}\right)^{\frac{1}{\alpha - 1}} \tag{62}$$

$$k_2^* = \left(\frac{\beta^{-1} + \delta - 1}{a b_1 \alpha}\right)^{\frac{1}{\alpha - 1}}. \tag{63}$$

with their corresponding steady state for consumption c_1^* and c_2^*.

Figure 9 shows the results for $\beta = 0.9$, $\alpha = 0.33$, $\delta = 0.1$, $g = 0$, $z_0 = 1$, $a = 0.5$, $b_1 = 3$, $b_2 = 2.5$, and $k_0 = \{0.5, 1.0, 3.0, 4.0\}$. In this experiment $\hat{X} = \{0, 1, 2, \ldots, 29\}$ and $\hat{X}_{\text{test}} = \{0, 1, 2, ..., 49\}$. We approximate the capital path using a deep neural network with four hidden layers, each with 128 nodes. Hidden layers use Tanh and the output layer uses Softplus as

[8] See Skiba (1978) for a comprehensive and detailed treatment of this problem in a continuous time setup.

[9] It is worth noting that here we use Adam optimizer instead of LBFGS. The Adam optimizer is more stable than LBFGS. Due to Adam's longer process of optimization (more steps of optimization), the solution is exposed to more implicit regularization, which is very desirable for this problem.

activation functions. The left panel shows the capital paths and the right panel shows the corresponding consumption paths. The dashed vertical lines separate the interpolation from the extrapolation region.

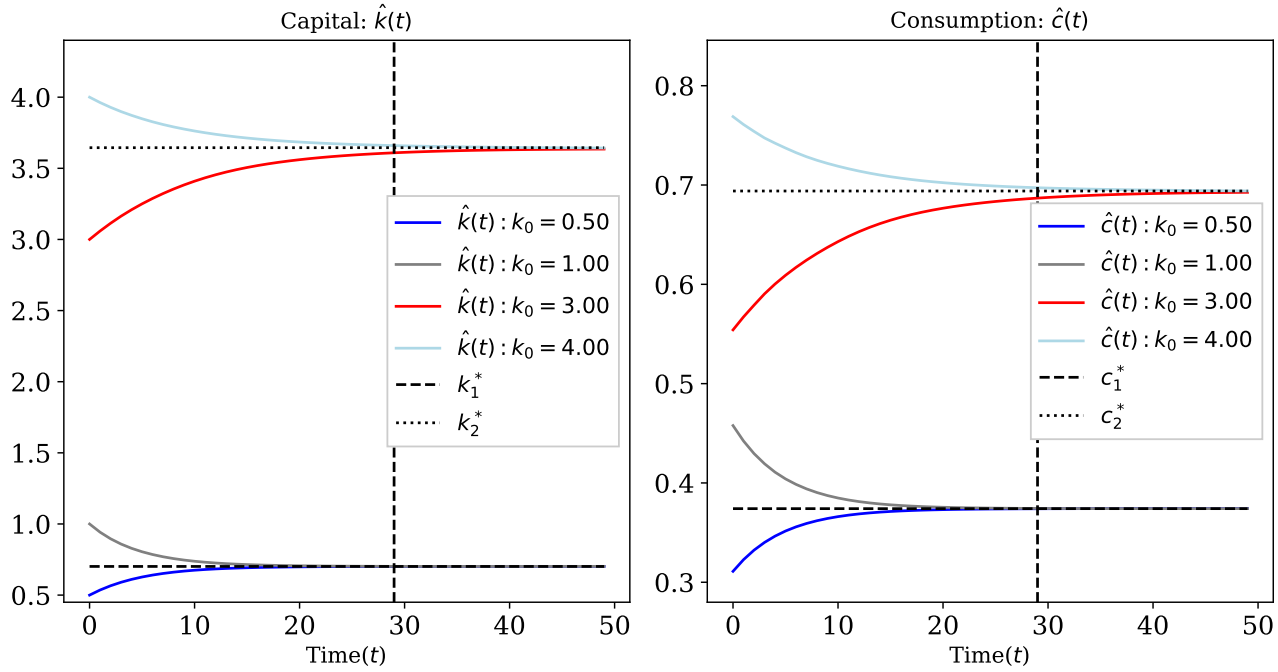

Figure 9: The approximate capital and consumption paths for the sequential neoclassical growth model with a convex-concave production function. The left panel shows the capital paths for four levels of initial capital $k_0 \in \{0.5, 1.0, 3.0, 4.0\}$ and the right panel shows the corresponding consumption paths. The dashed vertical line separates the interpolation from the extrapolation region.

These results show that even in the presence of multiplicity of steady states, the implicit bias picks up the right paths for capital and consumption in the vicinity of the steady states. It is worth noting that when the initial capital is above k_2^* or below k_1^* the optimal solution has the lowest semi-norm and the implicit bias of deep neural networks can accurately represent the optimal solution.

Figure 10 shows the capital and consumption paths for a grid of initial conditions $k_0 \in [0.5, 1.75]$ and $k_0 \in [2.75, 4]$. The top panel shows the capital paths and the bottom panel shows the consumption paths. The dashed horizontal lines show the steady states of capital k_1^* and k_2^* and their corresponding steady states for consumption c_1^* and c_2^*. The dashed vertical lines separate the interpolation from the extrapolation region. This result shows that algorithm is robust to variations in the initial condition for capital in the vicinity of the steady-states.

It is worth noting that the proposed method in this paper is not suitable for solving this problem across the space of initial conditions of capital between the two steady-states. It is

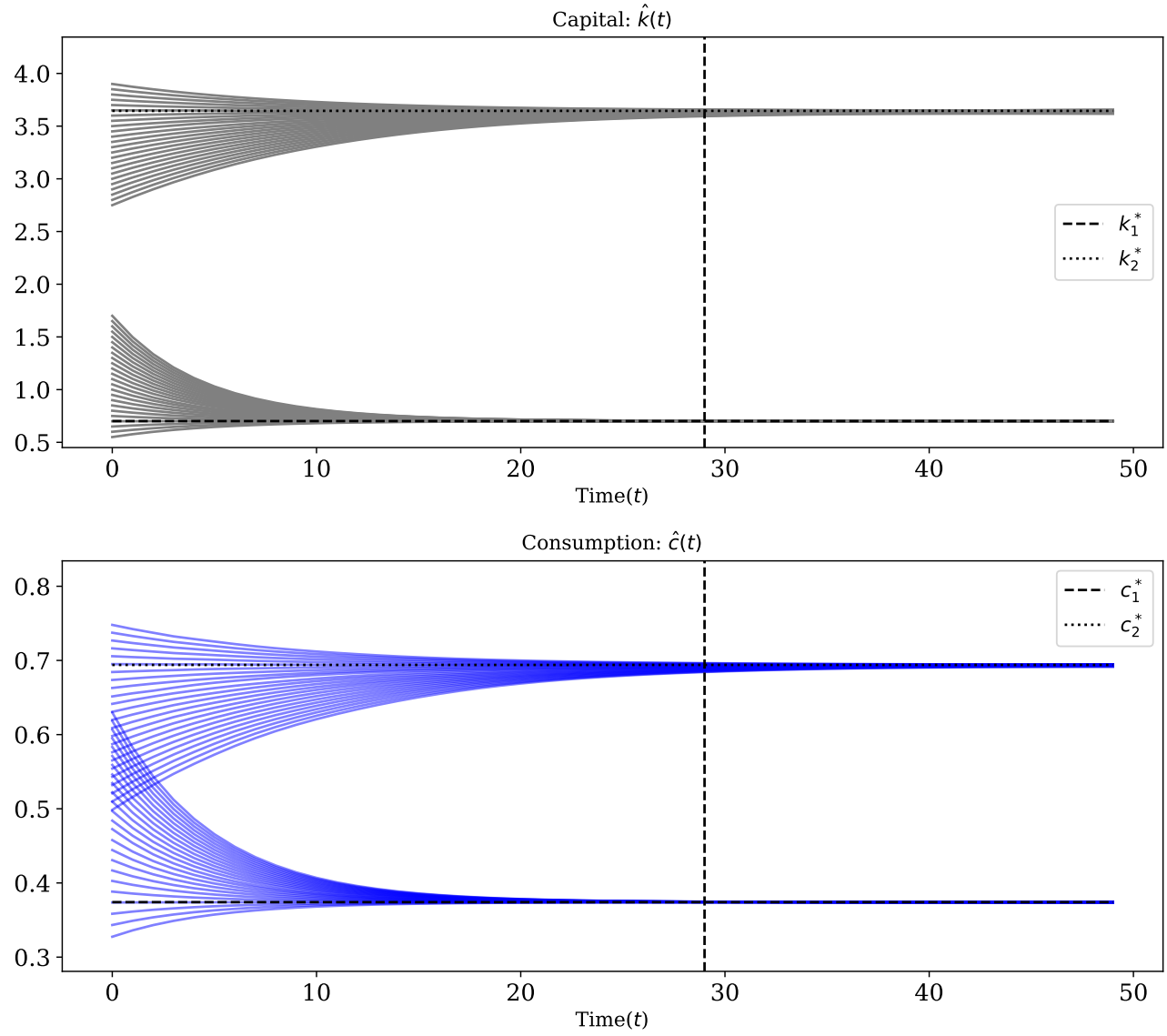

Figure 10: The approximate capital and consumption paths for the sequential neoclassical growth model with convex-concave production function. The gird for the initial condition of capital is from $[0.5, 1.75]$ and $[2.75, 4]$. The top panel shows the the capital paths and the bottom panel shows the consumption paths. The dashed horizontal lines show the steady states of capital k_1^* and k_2^* and their corresponding steady states for consumption c_1^* and c_2^*. The dashed vertical line separates the interpolation from the extrapolation region.

known that this problem has a bifurcation point in the space of initial conditions for capital, where above that point all the capital paths converge to the higher steady-state of capital (i.e. k_2^*) and below that point all the capital paths converge to the lower steady-state of capital (i.e. k_1^*). Due to the discontinuity in the derivative of the production function, we do not except our solution method to find a reliable solution in the vicinity of the bifurcation point.[10] See the

[10]See Benjamin Moll's lecture notes for a global treatment of this problem in a continuous time setup at

discussion in Appendix B.6 and Figure 25 for an analysis of the approximate solutions in the vicinity of that point.

5 Neoclassical growth model: Recursive form

The sequential models outlined in the previous sections can be very useful to study deterministic set ups and MIT shocks. However, they are not suitable for dealing with continuous shocks, such as shock to total factor productivity. In this section we focus on the recursive version of the neoclassical growth model.

Setup: The Bellman equation for the neoclassical growth model can be written as

$$v(k,z) = \max_{c,k'} \{u(c) + \beta v(k',z')\} \tag{64}$$

$$\text{s.t.} \quad k' = z^{1-\alpha} f(k) + (1-\delta)k - c \tag{65}$$

$$z' = (1+g)z \tag{66}$$

$$k' \geq 0, \tag{67}$$

where k is the capital, c is the consumption, and z is the total factor productivity. Capital depreciates with rate $\delta \in [0,1]$, the agent discounts the future with $\beta \in (0,1)$. Here we focus on the constant relative risk aversion utilities $u(c) = \frac{c^{1-\sigma}}{1-\sigma}$, and production function of the form $f(k) = k^{\alpha}$.

The Euler equation for the recursive form of the neoclassical growth model can be written as

$$u'(c) = \beta u(c') \left[z'^{1-\alpha} f'(k') + 1 - \delta \right], \tag{68}$$

where k' and c', z' are the capital, consumption and total factor productivity next period. Since, this is a recursive model k' and c are functions of capital k, and total factor productivity z, therefore, $X \equiv \mathbb{R}^2_+$. We are looking for functions $k'(k,z)$ and $c(k,z)$ such that they satisfy the Euler equation and feasibility condition. Moreover, the optimality of the solution requires all the capital and consumption paths must satisfy the transversality condition for all the initial conditions in X

$$0 = \lim_{t \to \infty} \beta^T u' \left(c^T(k_0, z_0) \right) k'^T(k_0, z_0) \quad \text{for all } (k_0, z_0) \in X, \tag{69}$$

where $k'^T(k_0, z_0)$ and $c^T(k_0, z_0)$ are generated by iterating forward the optimal policy function for

https://benjaminmoll.com/wp-content/uploads/2020/06/skiba.pdf.

capital $k'(k,z)$, consumption function $c(k,z)$ and the law of motion for total factor productivity up to Tth period for a given k_0 and z_0.

5.1 Interpolating solution

As a generalization of the collocation approach we pick a parametric space of functions $\mathcal{H}(\Theta)$, where $\Theta \equiv \{\theta_1, \ldots, \Theta_M\}$ represents the parameters. We pick a grid of points $\hat{X}$ to represent X defined as

$$\hat{X} \equiv \{k_1, \ldots, k_{N_k}\} \times \{z_1, \ldots, z_{N_z}\}, \tag{70}$$

which a Cartesian product of sets of points in the capital and total factor productivity space. The Euler equation (i.e., equation (68)), the feasibility condition (i.e, equation (65)), the law of motion for the total factor productivity (i.e., equation (66)), and the transversality condition (i.e., equation (69)) form a system of equations on $\hat{X}$.

For a given policy function for capital $k'(k,z;\theta) \in \mathcal{H}(\Theta)$ we define a consumption function $c(k,z;k'(\cdot;\theta))$ via the feasibility condition

$$c\big(k,z;k'(\cdot;\theta)\big) \equiv z^{1-\alpha} f(k) + (1-\delta)k - k'(k,z;\theta). \tag{71}$$

.

Given a parametric space of functions $\mathcal{H}(\Theta)$, a grid $\hat{X}$, and very large point in time T, one can find the policy function for capital $k'(k,z;\theta) \in \mathcal{H}(\Theta)$ via the following optimization problem

$$\min_{\theta\in\Theta} \frac{1}{|\hat{X}|} \sum_{(k,z)\in\hat{X}} \left[\frac{u'\Big(c\big(k,z;k'(\cdot;\theta)\big)\Big)}{u'\Big(c\big(k'(k,z;\theta),(1+g)z;k'(\cdot;\theta)\big)\Big)} - \beta\big[\big((1+g)z\big)^{1-\alpha} f'\big(k'(k,z;\theta)\big) + (1-\delta)\big] \right]^2 + \underbrace{\frac{1}{|\hat{X}|} \sum_{(k_0,z_0)\in\hat{X}} \left[\beta^T u'\Big(c^T\big(k_0,z_0;k'(\cdot;\theta)\big)\Big) k'^T\big(k_0,z_0;\theta\big) \right]^2}_{\text{Is this necessary?}}. \tag{72}$$

The first term in the optimization problem are the Euler residuals and the second term is a finite approximation of the transversality condition.

Choice of $\hat{X}$: Similar to the previous cases we provide results for different grids. We also evaluate the optimal approximate policy function for capital $\hat{k}(\cdot,\cdot;\theta)$ outside of $\hat{X}$ to study the generalization performance of the solutions.

Where is the transversality condition? Without the transversality condition the recursive version of the neoclassical growth has more than one solution. Therefore, in the absence of the second term (finite approximation of the transversality condition) the optimization problem described in (72) has more than one solution that achieve the zero of the objective function. As noted by Fernández-Villaverde et al. (2016), without explicitly imposing the transversality condition, it is necessary to verify that solutions satisfy the transversality condition. This verification process is done after solving the minimization problem by generating capital and consumption paths from some initial conditions.

In section Section 5.3 we establish how using over-parameterized functions ($M \gg N_k \times N_z$) and their corresponding optimization processes lead to the solution that does not violate the transversality condition. Intuitively, the policy functions for capital that lead to violation of the transversality condition have bigger derivatives than the optimal policy function for capital. Therefore, we do not need to be worried about long run regularity conditions imposed by the transversality condition. Therefore, using over-parameterized functions we can drop the transversality condition and solve

$$\min_{\theta\in\Theta} \frac{1}{|\hat{X}|} \sum_{(k,z)\in\hat{X}} \left[\frac{u'\Big(c\big(k,z;k'(\cdot;\theta)\big)\Big)}{u'\Big(c\big(k'(k,z;\theta),(1+g)z;k'(\cdot;\theta)\big)\Big)} - \beta\big[\big((1+g)z\big)^{1-\alpha} f'\big(k'(k,z;\theta)\big) + (1-\delta)\big] \right]^2 . \tag{73}$$

The non-negativity of the policy function for capital, $k'(k,z;\theta)$ is built into $\mathcal{H}(\Theta)$.

5.2 Results

Figure 11 shows the result of the minimization problem described in equation (73) for $\beta = 0.9$, $\alpha = 0.33$, $\delta = 0.1$, $g = 0$, $k_0 = 0.4$, and $z_0 = 1$. In this experiment we utilize a grid with 16 points between $k_1 = 0.8$ and $k_{N_k} = 2.5$. We use a deep neural network with four hidden layers, each with 128 nodes. Each hidden layer uses Tanh and the output layer uses Softplus as activation functions. The solid curve in the top-left panel shows the median of capital paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k,1)$ from the initial condition $k_0 = 0.4$ over 100 random initializations of the parameters of the deep neural network (seeds). The dashed curve (denoted by $k(t)$) shows capital path obtained via the value function iteration method. The solid curve is accompanied with 10th and 90th percentiles over 100, however, they are so small that are not visible in the plot. The top-right panel shows the median of the relative errors between the approximate capital paths and the capital path obtained by the value function iteration method. The shaded region shows the 10th and 90th percentiles over

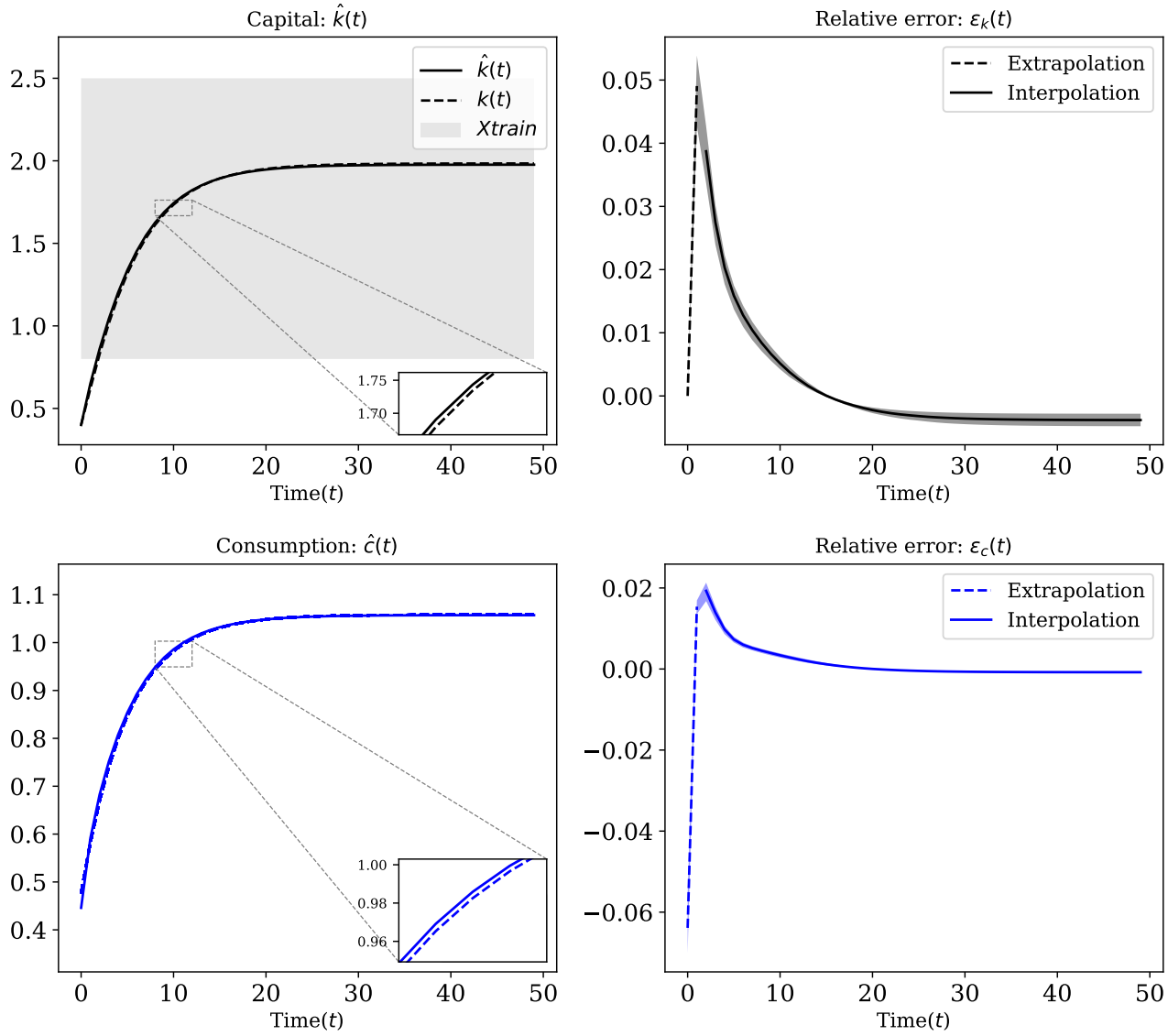

Figure 11: Comparison between the value function iteration and approximate solution using a deep neural network for the recursive neoclassical growth model. The left panels show the median of capital and consumption paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k, 1)$ and consumption function $c(k, 1; \hat{k}')$ over 100 seeds. The dashed curves (denoted by $k(t)$ and $c(t)$) show the capital and consumption paths obtained via the value function iteration method. The right panels show the median relative errors for capital and consumption paths between approximate solution and the solution obtained via the value function iteration method. The shaded regions show the 10th and 90th percentiles. The dashed curves in the right panels show the relative errors for the parts of capital paths that lie outside of the interpolation regions (i.e. $\hat{X}$). The gray region in the top-right panel shows the interpolation region.

100 seeds. The dashed part of the curve shows the median of relative errors in the extrapolation region (the parts of capital path outside of $\hat{X} = [0.8, 2.5]$) and the solid part shows the median

of relative errors in the interpolation region (the parts of capital path inside $\hat{X}$). The solid curve in the bottom-left panel shows the median of consumption paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k,1)$ and consumption function $c(k,1;\hat{k}')$ from the initial condition $k_0 = 0.4$ over 100 seeds. The dashed curve shows the consumption path obtained via the value function iteration method. The bottom-right panel shows the median of the relative errors between the approximate consumption paths and the consumption path obtained by the value function iteration method. The shaded region shows the 10th and 90th percentiles over 100 seeds. The dashed part of the curve (denoted by $k(t)$) shows the median of relative errors for consumption where the corresponding levels of capital are in the extrapolation region and the solid part shows the median of relative errors for consumption where the corresponding levels of capital are in the interpolation region. The gray region in the top-left panel shows the interpolation region (i.e., $\hat{X}$).

To be more specific, the relative error between the approximate and the value function iteration solution for capital and consumption are defined as

$$\varepsilon_c(t) = \frac{\hat{c}(t) - c(t)}{c(t)} \tag{74}$$

$$\varepsilon_k(t) = \frac{\hat{k}(t) - k(t)}{k(t)}, \tag{75}$$

where $\hat{c}(t)$ and $\hat{k}(t)$ are calculated by iterating forward the approximate optimal policy function for capital $\hat{k}(k,1)$ and consumption function $c(k,1;\hat{k}')$

The results of this experiment is multi-fold. First, the implicit generalization in deep neural networks yields a solution that automatically satisfies the transversality condition. Second, the solutions are accurate within $\hat{X}$. Third, given the initial level of capital $k_0 = 0.4$ lies outside of $\hat{X} = [0.8, 2.5]$, the generalization error is small (at most 5% for capital).

As stated in the original problem, the transversality condition should hold for all the initial points of capital, Figure 26 in Appendix C.1 shows the results for different initial conditions of capital both inside $\hat{X}$ and outside $\hat{X}$.

Far from the steady state: In the last experiment we used a grid $\hat{X}$ that contains the steady state for capital k^* (i.e., $k^* \in \hat{X}$). In this experiment we investigate whether we can achieve accurate short run dynamic by using a local grid around the initial condition for capital that does not contain the steady state and is far from it. Specifically, we use a grid $\hat{X} = \{k_{\min}, k_{\max}\}$ where $k_{\max} < k^*$.

Figure 12 shows the results for the recursive neoclassical growth using a local grid around the initial level of capital $k_0 = 0.9$ defined as $\hat{X} = [0.8, 1.5]$. We use the same parameters and deep neural network as the previous experiment. The solid curve (denoted by $k(t)$) in the top-left panel

shows the median of capital paths generated by iterating forward the optimal policy function for capital $\hat{k}'(k,1)$ from the initial condition $k_0 = 0.9$ over 100 seeds. The dashed curve (denoted by $k(t)$) shows capital path obtained via the value function iteration method. The shaded region shows the 10th and 90th percentiles over 100 seeds. The top-right panel shows the median of the relative errors between the approximate capital paths and the capital path obtained by the value function iteration method. The shaded region shows the 10th and 90th percentiles over 100 seeds. The dashed part of the curve shows the median of relative errors in the extrapolation region and the solid part shows the median of relative errors in the interpolation region. The solid curve (denoted by $\hat{c}(t)$) in the bottom-left panel shows the median of consumption paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k,1)$ and consumption function $c(k,1;\hat{k}')$ from the initial condition $k_0 = 0.9$ over 100 seeds. The dashed curve (denoted by $c(t)$) shows the consumption path obtained via the value function iteration method. The bottom-right panel shows the median of the relative errors between the approximate consumption paths and the consumption path obtained by the value function iteration method. The shaded region shows the 10th and 90th percentiles over 100 seeds. The dashed part of the curve shows the median of relative errors for consumption where the corresponding levels of capital are in the extrapolation region and the solid part shows the median of relative errors for consumption where the corresponding levels of capital are in the interpolation region. The gray region in the top-left panel shows the interpolation region (i.e., $\hat{X}$).

These results show that one can achieve accurate short run dynamics when the grid for capital is locally defined around the initial capital and is far from the steady state. For the first few periods the relative error for capital is less then 1%. Moreover, the relative errors stay bounded do not grow excessively even after 20 periods, which can be a very promising avenue for adaptive and simulation based sampling methods in high dimensions.

Balanced growth path ($g > 0$): In this experiment we focus on the positive growth in total factor productivity (i.e. $g > 0$). Here, we partially use an a priori economic intuition that the solution is homogeneous of degree one in z. We design the space of functions $\mathcal{H}$ such that it contains functions of the form

$$\hat{k}'(k,z;\theta) = zNN\left(\frac{k}{z}, z; \theta\right). \tag{76}$$

where $NN(\cdot,\cdot;\theta)$ is a deep neural network. We say partially, because the correctly specified functional form of a homogeneous of degree one is $zNN\left(\frac{k}{z};\theta\right)$.

Figure 13 shows the results for recursive neoclassical growth for non-stationary total factor productivity (i.e., $g = 0.02$). Here we used 16 points in for $[0.8, 3.5]$ for capital, and 8 points in $[0.8, 1.8]$ for total factor productivity. The architecture for the deep neural network $NN(\cdot,\cdot;\theta)$ is the same as the first experiment. The only difference is the linear term z. The solid curve

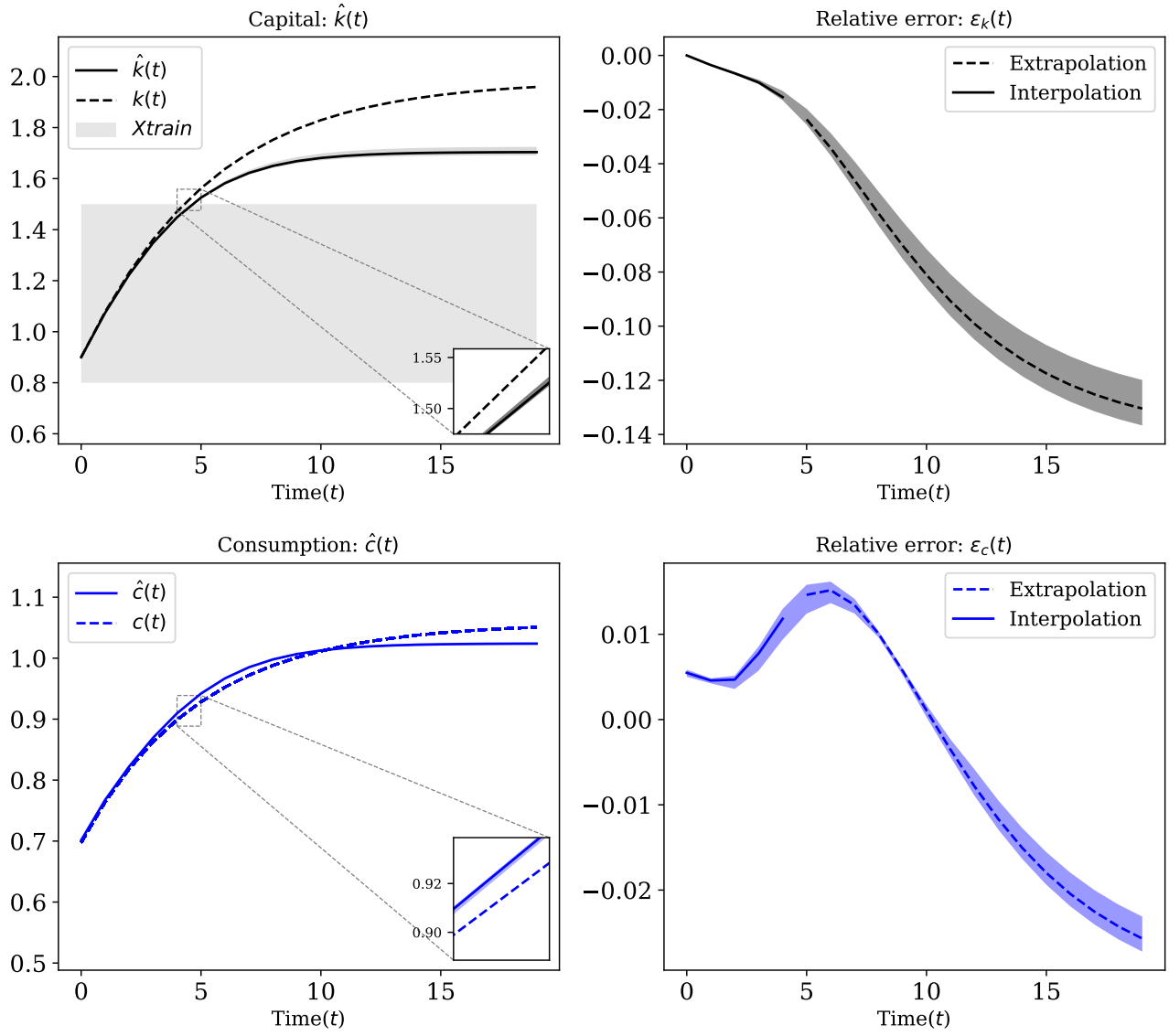

Figure 12: Comparison between the value function iteration and approximate solution using a deep neural network for the recursive neoclassical growth model using a local grid around the initial condition for capital $\hat{X} = [0.8, 1.5]$. The left panels show the median of capital and consumption paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k, 1)$ and $c(k, 1; \hat{k}')$ over 100 seeds. The dashed curves (denoted by $k(t)$ and $c(t)$) show the capital and consumption paths obtained via the value function iteration method. The right panels show the median relative errors for capital and consumption paths between approximate solution and the solution obtained via the value function iteration method. The shaded regions show the 10th and 90th percentiles. The dashed curves in the right panels show the relative errors for the parts of capital paths that lie outside of the interpolation regions (i.e. $\hat{X}$). The gray region in the top-right panel shows the interpolation region.

(denoted by $\hat{k}(t)$) in the top-left panel shows the median of the capital paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k, z)$, and $z' = (1 + g)z$

from the initial conditions $k_0 = 0.4$ and $z_0 = 1$ over 100 random initializations of the parameters of the deep neural network (seeds). The dashed curve (denoted by $k(t)$) shows the capital path obtained via the value function iteration method. The shaded region show the 10th and 90th percentiles over 100 seeds. The top-right panel shows the median of the relative errors between the approximate capital paths and the capital path obtained via the value function iteration method. The shaded region shows the 10th and 90th over 100 seeds. The dashed part of the curve shows the median of relative errors in the extrapolation region. The solid curve (denoted by $\hat{c}(t)$) in the bottom-left panel shows the median of approximate consumption paths generated by iterating forward the approximate policy for capital $\hat{k}(k, z)$, the consumption function $(k, z; \hat{k}')$ and $z' = (1 + g)z$ from the initial conditions for k and z over 100. The dashed curve (denoted by $c(t)$) shows the consumption path obtained via the value function iteration method. The bottom-right panel shows the median of the relative errors between the approximate consumption path and the consumption path obtained via the value function iteration method. The shaded regions show the 10th and 90th percentiles over 100 seeds. The dashed part of the curve shows the median of the relative errors for consumption where the corresponding levels of capital are in the extrapolation region.

The results of this experiment is multi-fold. First, the long run errors do not impair the accuracy of short and medium run dynamics even in the presence non-stationary total factor productivity. Second, the solutions are accurate within $\hat{X}$ (less than 1% relative error for capital in short and medium run). Third, the generalization errors (capital levels outside of $[0.8, 3.5]$) are small.

As stated in the original problem, the transversality condition should hold for all the initial points of capital, Figure 27 in Appendix C.1 shows the results for different initial conditions of capital both inside $\hat{X}$ and outside $\hat{X}$.

5.3 Minimum norm interpretation and transversality condition

In this section we provide the connection between minimum norm interpretation and the transversality condition for the case of recursive neoclassical growth. We focus on the case of zero total factor productivity growth (i.e., $g = 0$ and $z_0 = 1$).

As discussed in Section 3.3 the solutions that violate the transversality condition are associated with a sequence of consumption that approaches zero and a sequence of capital that approaches $\tilde{k}_{\max}$. Therefore, any policy for capital $\tilde{k}'(k)$ that solves the Euler equation and feasibility condition, and violates the transversality condition, must have a fixed point at $\tilde{k}_{\max}$. Formally

$$\tilde{k}'\left(\tilde{k}_{\max}\right) = \tilde{k}_{\max}.$$

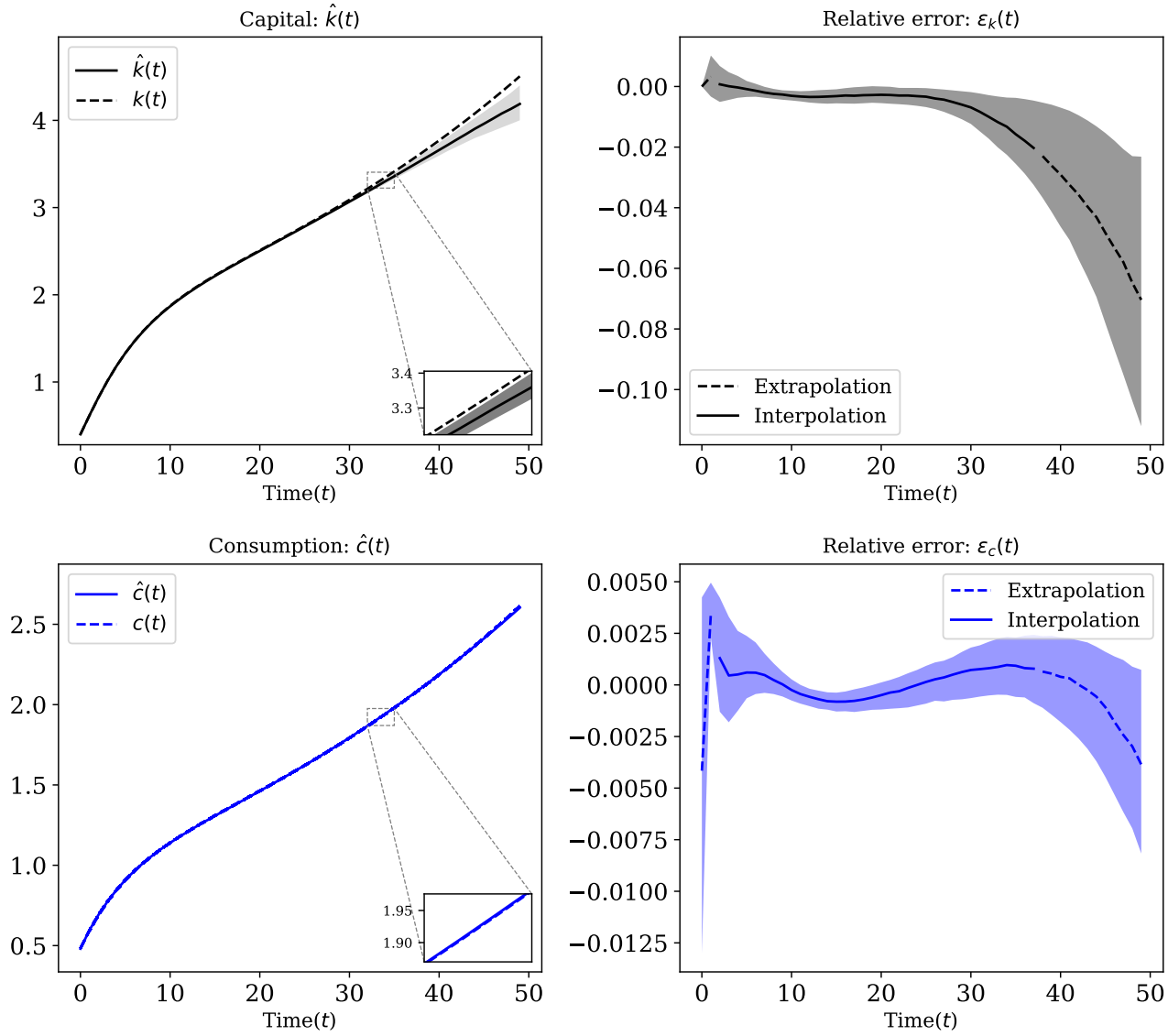

Figure 13: Comparison between the value function iteration and approximate solution using a deep neural network for the recursive neoclassical growth model with growth in total factor productivity (i.e., $g = 0.02$). We use $\hat{X} = [0.8, 3.5] \times [0.8, 1.8]$. The left panels show the median of capital and consumption (denoted by $\hat{k}(t)$ and $\hat{c}(t)$) paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k, z)$, consumption function $c(k, z; \hat{k}')$, and $z' = (1 + g)z$ over 100 seeds. The dashed curves (denoted by $k(t)$ and $c(t)$) show the capital and consumption paths obtained via the value function iteration method. The right panels show the median relative errors for capital and consumption paths between approximate solutions and the solution obtained via the value function iteration method. The shaded regions show the 10th and 90th percentiles. The dashed curves in the right panels show the relative errors for the parts of capital paths that lie outside of the interpolation regions (i.e. $\hat{X}$).

The optimal policy function for capital $k'(k)$ has a fixed point at $k^* = \left(\frac{\beta^{-1}+\delta-1}{\alpha}\right)^{\frac{1}{\alpha-1}}$. Formally

$$k'(k^*) = k^*.$$

Given the fact that $\tilde{k}_{\max} \gg k^*$, $\tilde{k}$ must intersect with 45 degree line way further to the right of k^*.

In Figure 14 the blue curve, denoted by $\tilde{k}'(k)$, shows a policy function for capital that violates the transversality condition and the black curve, denoted by $k'(k)$, shows the optimal policy function for capital. The dashed line shows the 45 degree line. $\tilde{k}'(k)$ intersects with the 45 degree line at $\tilde{k}_{\max} \approx 30$. [11].

The minimum norm interpretation of the minimization problem described in (72) can be written as

$$\min_{k'(\cdot;\theta)\in\mathcal{H}(\Theta)} \quad \|k'(\cdot;\theta)\|_S \tag{77}$$

$$\text{s.t. } u'\Big(c\big(k;k'(\cdot;\theta)\big)\Big) = \beta u'\Big(c\big(k'(k);k'(\cdot;\theta)\big)\Big)\Big[f'\big(k'(k;\theta)\big) + (1-\delta)\Big] \quad \text{for } k \in \hat{X} \tag{78}$$

$$k'(k;\theta) \geq 0 \quad \text{for } k \in \hat{X} \tag{79}$$

$$0 = \lim_{t\to\infty} \beta^T u'\left(c^T(k_0)\right) k'^T(k_0) \quad \text{for all } k_0 \in X. \tag{80}$$

Since the total factor productivity is constant and is always equal to one, we replace $k'(k,z;\theta)$ by $k'(k;\theta)$ and $\hat{X}$ is defined as

$$\hat{X} \equiv \{k_1, \ldots, k_{N_k}\}. \tag{81}$$

$\mathcal{H}(\Theta)$ is an over-parameterized space of functions and $\|\cdot\|_S$ satisfies Assumption 1.

As evident in Figure 14, $\tilde{k}'(k)$ has bigger derivatives than $k'(k)$, i.e., for an arbitrary compact space of the form $[k_{\min}, k_{\max}]$

$$\int_{k_{\min}}^{k_{\max}} \left|\frac{d\tilde{k}'}{dk}\right|^2 dk > \int_{k_{\min}}^{k_{\max}} \left|\frac{dk'}{dk}\right|^2 dk. \tag{82}$$

Therefore, by Assumption 1

$$\|k'\|_S < \|\tilde{k}'\|_S. \tag{83}$$

Since the solutions that violate the transversality condition have bigger norms, the over-parameterized interpolation eliminates those solutions. Therefore, the optimization problem described in (73) is enough to find the optimal policy function for capital $k'(k,z;\theta)$ without imposing any explicit regularity regarding the long run behavior.

[11] In this figure the solutions that violate the transversality condition (i.e. $\tilde{k}'(k)$) are generated via transforming the sequences of capitals that solve the sequential problem and violate the transversality condition.

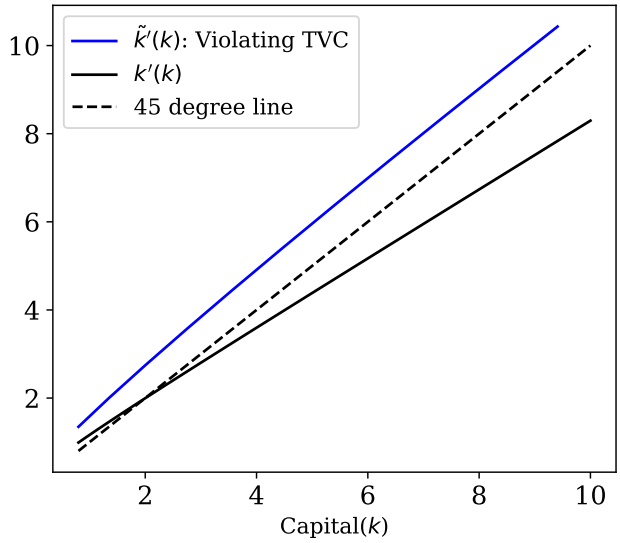

Figure 14: Comparison between the optimal solution and a solution that violate the transversality condition in recursive neoclassical growth model. The solution that violates the transversality equation intersects with the 45 degree line at $\tilde{k}_{\max} \approx 30$.

6 Are Euler and value function residuals enough?

In dynamic models where there is multiplicity of solutions and some boundary conditions at infinity is required to uniquely pin down the optimal solution, finding the root of the functional operators such as Euler or value function residuals is not enough. Due to multiplicity very low levels residuals can be misleading and the solutions might represent non-optimal solutions.

Recursive neoclassical growth model: Minimizing the Euler residuals is not enough For instance in the recursive neoclassical growth model, we want to have a solution that minimizes the Euler residuals and satisfy the transversality condition. Equivalently, the optimal policy function for capital $k'(k)$ has to minimize the Euler residuals and have a fixed point at the steady state k^*. As illustrated in Section 5.3, when we approximate the optimal policy function for capital $k'(k)$ with a deep neural network, the implicit regularization automatically picks the optimal solution. Intuitively, since the optimal policies for capital that violate the transversality condition have bigger norms.

Due to the trade off between the consumption function $c(k)$ and the optimal policy function for capital $k'(k)$ through the feasibility condition, if $k'(k)$ has a bigger norm then $c(k)$ has smaller norm. Therefore, if one decides to approximate the consumption function instead of the optimal policy function for capital they get solutions that violate the transversality condition. Equivalently, the corresponding policy function for capital $k'(k)$ have a fixed point at $\tilde{k}_{\max}$ instead of the steady state k^*. Here we focus on the case with no growth in total factor productivity (i.e.,

$g = 0$ and $z_0 = 1$) and in the notation drop the total factory productivity as an input. The optimization we solve for approximating the consumption function for capital can be written as

$$\min_{\theta \in \Theta} \frac{1}{|\hat{X}|} \sum_{k \in \hat{X}} \left[\frac{u'\big(c(k;\theta)\big)}{u'\Big(c\big(k'(k;c(\cdot;\theta));\theta\big)\Big)} - \beta \left[f'\Big(k'\big(k;c(\cdot;\theta)\big)\Big) + (1-\delta) \right] \right]^2, \tag{84}$$

where for a given consumption function $c(k,\theta) \in \mathcal{H}(\Theta)$, the policy function for capital is defined as

$$k'\big(k;c(\cdot;\theta)\big) \equiv f(k) + (1-\delta)k - c(k;\theta), \tag{85}$$

The policy function for capital is defined exactly and not approximated. The consumption function $c : \mathbb{R} \to \mathbb{R}_+$ is approximated by a deep neural network. Similar to the previous cases the grid for capital is defined as $\hat{X} = \{k_1, \ldots, k_{N_k}\}$.

Figure 15 shows the comparison between approximating the optimal policy function for capital $k'(k)$ versus the approximating the consumption function $c(k)$ with a deep neural network. The economic parameters are the same as before. The solid curve in top-left panel shows the median of the Euler residuals squared over 100 different initializations of the parameters of the deep neural network (seeds) when $k'(k)$ is approximated with a deep neural network. In this case the Euler residual is defined as

$$\varepsilon_E^k(k;\theta) \equiv \frac{u'\Big(c\big(k;k'(\cdot;\theta)\big)\Big)}{u'\Big(c\big(k'(k;\theta);k'(\cdot;\theta)\big)\Big)} - \beta \left[f'\big(k'(k;\theta)\big) + (1-\delta) \right]. \tag{86}$$

Where the superscript k denotes the Euler residuals when approximating the policy function for capital with a deep neural network. The shaded regions around the solid curve show the 10th and 90 percentiles of the Euler residuals squared over 100 seeds. The bottom-left panel shows the median of approximate optimal policy function for capital over 100 seeds when $k'(k)$ is approximated with a deep neural network. The figure also contains the 10th and 90 percentiles, however they are very close together that are not visible in the plot. In this case the approximate optimal policy function for capital has a fixed point at $k^* \approx 2.0$. The solid curve in top-right panel shows the median of the Euler residuals squared over 100 seeds when $c(k)$ is approximated with a deep neural network. In this case the Euler residual is defined as

$$\varepsilon_E^c(k;\theta) \equiv \frac{u'\big(c(k;\theta)\big)}{u'\Big(c\big(k'(k;c(\cdot;\theta));\theta\big)\Big)} - \beta \left[f'\Big(k'\big(k;c(\cdot;\theta)\big)\Big) + (1-\delta) \right] \tag{87}$$

Where the superscript c denotes the Euler residuals when approximating the consumption function with a deep neural network. The bottom-right panel shows the median of approximate optimal policy function for capital over 100 seeds when $c(k)$ is approximated with a deep neural network. The shaded regions show 10th and 90 percentiles of the optimal policy function for capital when $c(k)$ is approximated with a deep neural network. In this case the approximate optimal policy function for capital has a fixed point at $\tilde{k}_{\max} \approx 30$.

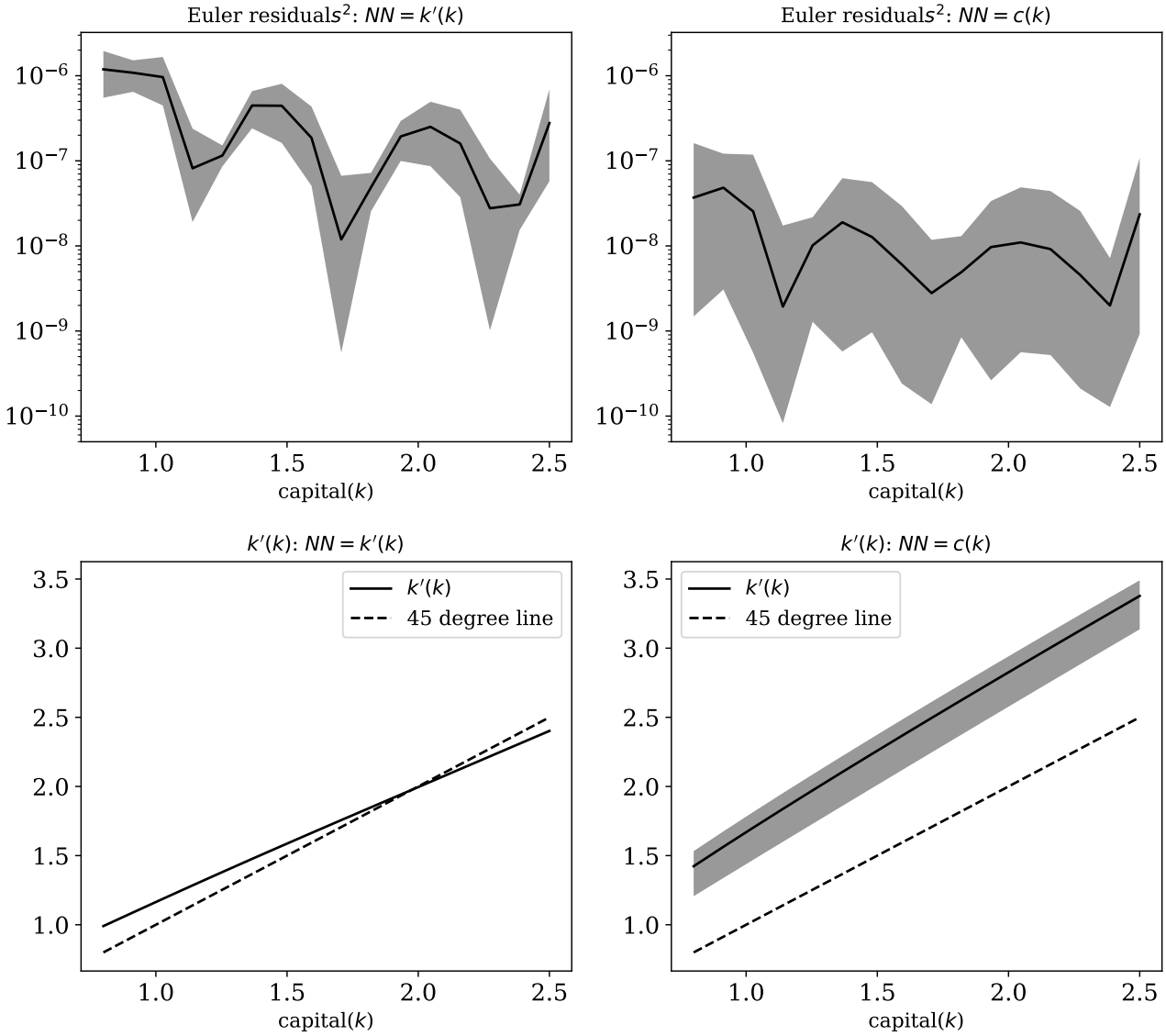

Figure 15: Comparison between approximating the optimal policy function for capital $k'(k)$ versus the consumption function $c(k)$ with a deep neural network. The left panels show the Euler residuals and the optimal policy function for capital when $k'(k)$ is approximated with a deep neural network. The right panels show the Euler residuals squared and the optimal policy function for capital when $c(k)$ is approximated with a deep neural network. The solid curves show the medians and the shaded areas show 10th and 90the percentiles over 100 different seeds.

As Explained these results show that when the optimal policy function for capital is approximated with a deep neural network, $\hat{k}(k)$ has a fixed point at $k^* \approx 2.0$ and it does not violate the transversality condition (bottom-left panel). However, when the the consumption function (i.e., $c(k)$) is approximated with a deep neural network, the optimal policy function for capital has a fixed point at $\tilde{k}_{\max}\tilde{3}0$ and it violates the transversality condition (bottom-right panel). As shown in the bottom-right panel, this result is robust to the different random initializations of the parameters of the deep neural network. Most importantly, the Euler residuals for the solutions that violate the transversality condition (top-right panel) are systematically and substantially lower than the solutions that do not violate the transversality condition. Therefore, having low Euler residuals is not a sufficient condition for convergence to the optimal solution.

Can explicit regularization solve the problem? As shown in Figure 15 approximating the consumption function $c(k)$ instead of the optimal policy function for capital leads to solutions that violate the transversality condition. One natural question that arises is whether explicit regularization of the parameters of the deep neural network can fix this problem. One of the most common methods of explicit regularization is L_2 regularization. This regularization is achieved by penalizing the L_2 norm of the parameters of the neural networks (i.e., $\sum_{\theta_i \in \Theta} \theta_i^2$).[12] Therefore, the minimization takes the following form

$$\min_{\theta \in \Theta} \frac{1}{|\hat{X}|} \sum_{k \in \hat{X}} \left[\frac{u'\big(c(k;\theta)\big)}{u'\Big(c\big(k'(k;c(\cdot;\theta));\theta\big)\Big)} - \beta \left[f'\Big(k'\big(k;c(\cdot;\theta)\big)\Big) + (1-\delta) \right] \right]^2 + \lambda \sum_{\theta_i \in \Theta} \theta_i^2, \tag{88}$$

where λ is the penalization coefficient.

Figure 16 shows the results for approximating the consumption function with and without regularization. The left panels show the results without regularization. The right panels show the results with L_2 regularization with penalization coefficient $\lambda = 1e-9$. The solid curve in the top panels shows the median of the Euler residuals squared defined in equation (87), and the shaded regions show the 10th and 90th percentiles over 100 seeds. The solid curve in the bottom panels shows the median of optimal policy function for capital $k'(k)$ associated with $c(k;\theta)$ and the shaded regions show the 10th and 90th percentiles over 100 seeds.

This result show that explicit regularization cannot lead to solutions that do not violate the transversality condition. However, this regularization reduces the variation of the Euler residuals and the optimal policy function for capital over the random initializations of the parameters of the deep neural network.

[12] This is also called weight decay, and it is very easy to implement in PyTorch and TensorFlow.

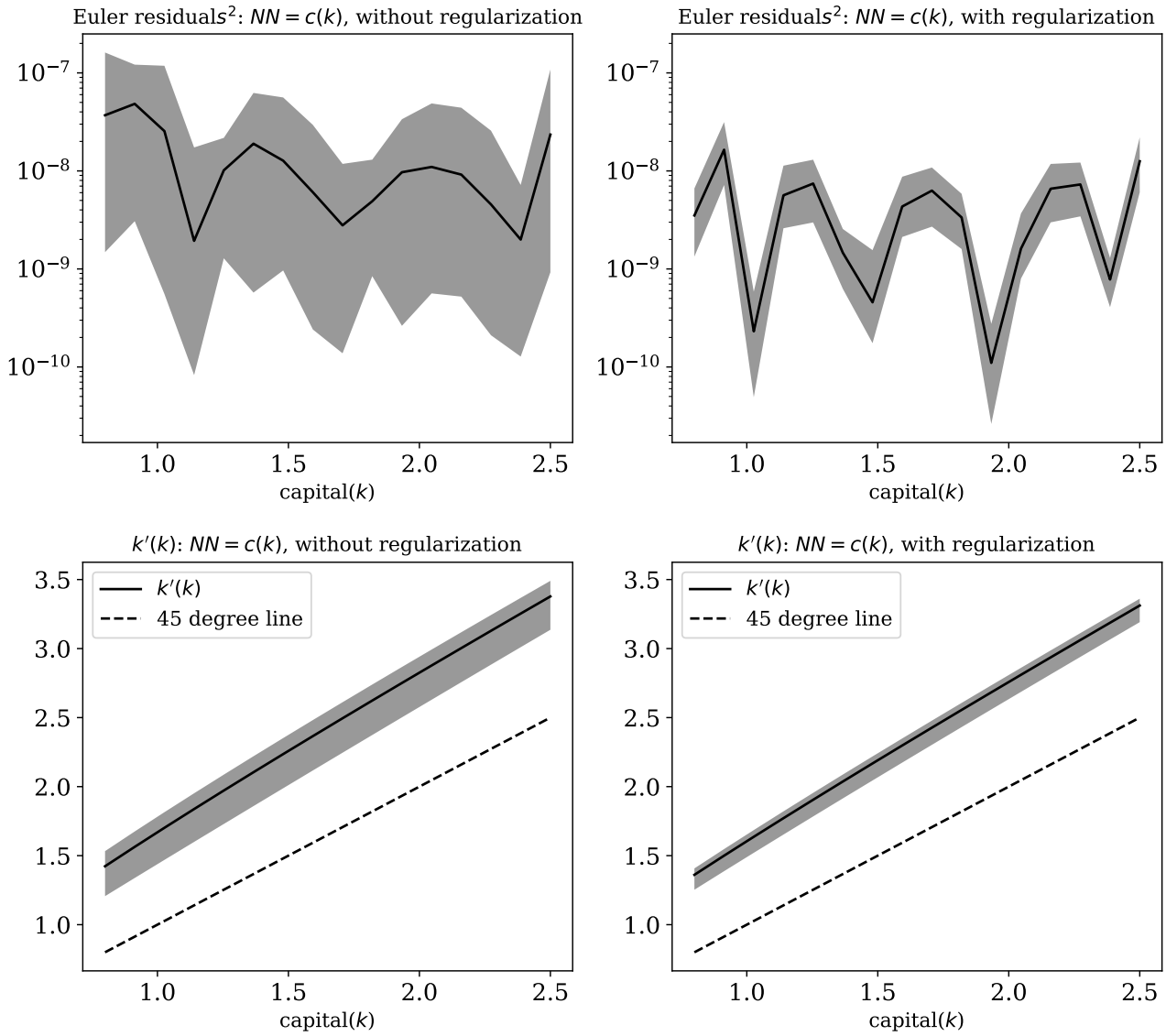

Figure 16: The effect of the L_2 explicit regularization on the solution when the consumption function $c(k)$ is approximated with a deep neural network instead of the optimal policy function for capital. The left panels show the results without regularization. The right panels show the results with L_2 regularization with penalization coefficient $\lambda = 10^{-9}$. The solid curve in the top panels shows the median of the Euler residuals squared and the shaded regions show the 10th and 90th percentiles over 100 seeds. The solid curve in the bottom panels shows the median of optimal policy function for capital $k'(k)$ associated with $\hat{c}(k;\theta)$ and the shaded regions show the 10th and 90th percentiles over 100 seeds.

Sequential asset pricing model: Minimizing the residuals is not enough Similarly, for the sequential models such as linear asset pricing low residuals errors can be misleading. As noted before the solutions can be written as

$$p(t) = p_f(t) + \zeta\beta^{-t}.$$

From the theoretical perspective, given that the dividends are positive, prices should be non-negative for every time period. Therefore, negative values of ζ are not allowed and as showed in Proposition 1 the price based on the fundamentals has the lowest semi-norm for all non-negative values of ζ. However, when $\hat{X} = \{t_1, \ldots, t_N\}$ only contains small values negative values of ζ can leak into the interpolating solutions. It is important to note that they are still interpolating solutions and solve the optimization problem accurately. This is because when t_N is small the solutions with negative ζ have smaller semi-norms.

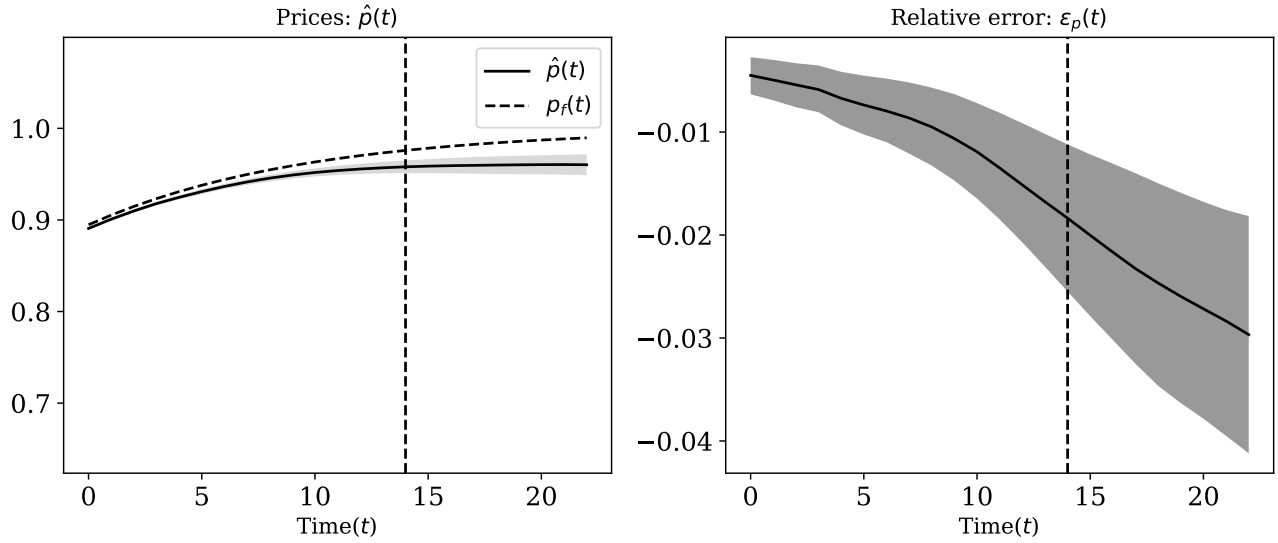

Figure 17: Comparison between the accurate and approximate solution using a deep neural network for the sequential linear asset pricing model for small $t_{\max}$. The dashed vertical lines separate the interpolation from the extrapolation region. In the left panel the solid curve shows the median of approximate prices over 100 seeds. The dashed curve shows the price based on the fundamentals. The solid curve in the right panel shows the median of relative errors between the approximate price and the price based on the fundamentals over seeds. The shaded regions show 10th and 90th percentiles.

Figure 17 shows the result for the sequential asset pricing model for $\hat{X} = \{0, 1, \ldots, 14\}$. The parameters and the deep neural network is the same as before. The dashed curve in the left panel shows the price based on the fundamentals and the solid curve shows the median of approximate prices over 100 seeds, and the shaded regions shows the 10th and 90th percentiles over those 100 seeds. The right panel shows the relative errors, the solid curve is the median of relative errors over 100 seeds and the shaded regions shows the 10th and 90th percentiles.

These results shows the leakage of negative value of ζ into the interpolating solutions. Since ζ is negative and non-negativity of $p(\cdot;\theta)$ is built into $\mathcal{H}(\Theta)$ the approximate solutions $\hat{p}(t)$ are bounded between 0 and $p_f(t)$.

Figure 18 is the result of the same experiment with $\hat{X} = \{0, 1, \ldots, 9\}$. The black solid curve, denoted by $\hat{p}(t)$ shows the median of the approximate price paths over 100 seeds and the shaded

region shows the 10th and 90th percentiles. The dashed curve, denoted by $p_f(t)$ shows the price based on the fundamentals. The solid curve in the right panel shows the median of relative error between the approximate prices and the price based on the fundamentals and the shaded region shows 10th and 90th percentiles. Since the approximate solutions solve the interpolation problem accurately they can be written as

$$\hat{p}(t) = p_f(t) + \hat{\zeta}\beta^{-t}.$$

By setting $t = 0$ we can find the corresponding $\hat{\zeta}$ as

$$\hat{\zeta} = \hat{p}(0) - p_f(0). \tag{89}$$

The solid blue line shows the median of $\hat{p}(t) - \hat{\zeta}\beta^{-t}$ over 100 seeds.The blue shaded regions shows

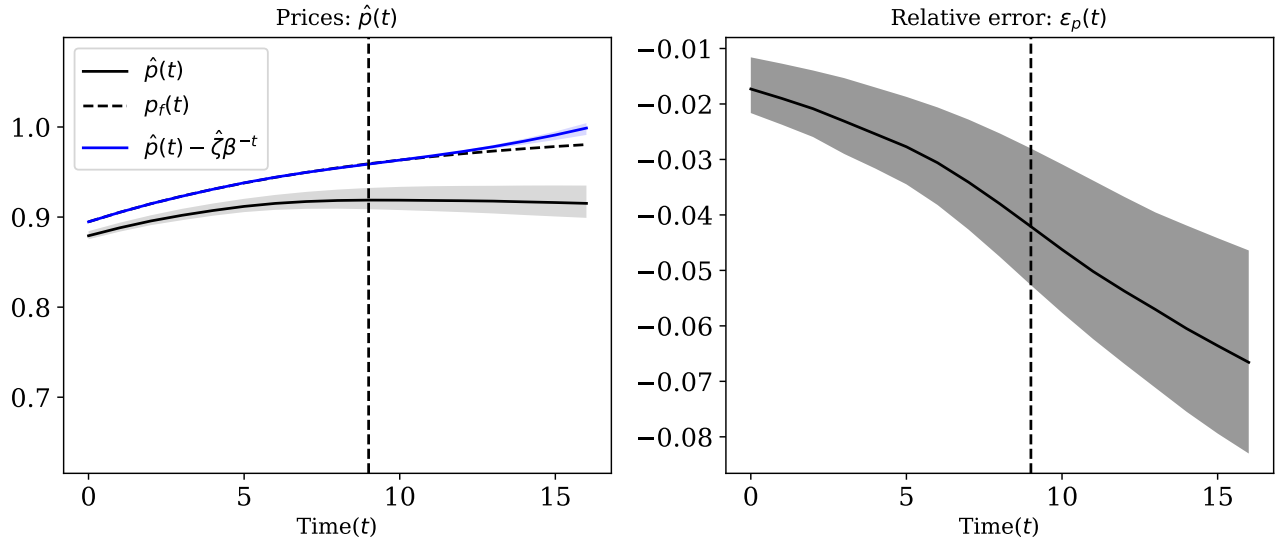

Figure 18: Analyzing the approximate solutions of the sequential linear asset pricing model for small $t_{\max}$. In this experiment $\hat{X} = \{0, 1, \ldots, 9\}$. The dashed vertical lines separate the interpolation from the extrapolation region. In the left panel the black solid curve (denoted by $\hat{p}(t)$) shows the median of approximate prices over 100. The dashed curve (denoted by $p_f(t)$) shows the price based on the fundamentals. The blue solid curve shows the median $\hat{p}(t) - \hat{\zeta}\beta^{-t}$ over seeds. The right panel shows the difference between the approximate prices and the price based on the fundamentals. The shaded regions show the 10th and 90th percentiles.

the 10th and 90th percentiles for these 100 price paths.

These results confirm that when t_N is small (in this example $t_{\max} = 9$) the approximate solution accurately solves the interpolation problem. This is evident from observing that $p_f(t) \approx \hat{p}(t) - \hat{\zeta}\beta^{-t}$ very accurately in $\hat{X}$. The 10th and 90th percentiles for $\hat{p}(t) - \hat{\zeta}\beta^{-t}$ are negligible. This happens because the variation in the approximated prices is due to very small variations in $\hat{\zeta}$. Moreover these results confirm that the leakage of negative values of ζ gets more severe for

smaller values of t_N. However, the approximate solutions are still bounded between 0 and $p_f(t)$.

7 Implicit bias of deep neural networks

For a given over-parameterized space of functions $\mathcal{H}(\Theta)$, an economic model $\ell(\cdot,\cdot)$ (e.g., Euler residuals, Bellman residuals), and a grid $\hat{X}$ the interpolation problem can be written as

$$\min_{\hat{\Psi}\in\mathcal{H}(\Theta)} \sum_{x\in\hat{X}} \ell(\hat{\Psi}(\cdot;\theta),x)^2. \tag{90}$$

Since in over-parameterized interpolation problems, such as interpolation with deep neural networks, the number of parameters is larger than the number of grid points (cardinality of $\hat{X}$) there are many interpolating solutions that can achieve $\sum_{x\in\hat{X}} \ell(\hat{\Psi}(\cdot;\theta),x)^2 = 0$. In this case individual parameters become meaningless and it is more meaningful to think of these interpolating functions in the space of functions (as opposed to the space of parameters Θ). Therefore, we drop the parametric notation Θ and replace $\mathcal{H}(\Theta)$ with $\mathcal{H}$. It has been proposed that one can understand the implicit regularization in the interpolating solutions of minimization described in (90) by looking at the solutions of the following optimization

Problem 1.

$$\min_{\hat{\Psi}\in\mathcal{H}} ||\hat{\Psi}||_S \tag{91}$$

$$s.t.\ \ell(\hat{\Psi},x) = 0, \qquad for\ x\in\hat{X}, \tag{92}$$

where is $\|\cdot\|_S$ is some functional norm. Although, understanding this norm is still an open question in modern machine learning, optimization theory, and statistics, there has been significant advances in understanding this norm. Here we provide an assumption that we use through the paper and provide ample evidence from the literature on the validity of this assumption

Assumption 2. *Let $\|\cdot\|_S$ be the norm in Problem 1, Ψ and Φ be two differentiable solutions from $\mathcal{X}$ to $\mathbb{R}$, i.e., $\ell(\Psi,x) = \ell(\Phi,x) = 0$ for all $x\in\mathcal{X}$ such that*

$$\int_{\mathcal{X}} \left|\frac{d\Psi}{dx}\right|^2 dx > \int_{\mathcal{X}} \left|\frac{d\Phi}{dx}\right|^2 dx \tag{93}$$

then

$$\|\Psi\|_S > \|\Phi\|_S. \tag{94}$$

Discussion regarding the validity of Assumption 1: Deep neural networks and deep learning, as a method of function approximation, have shown an incredible power in predicting

unseen data, i.e., generalization power Zhang et al. (2021). Given that they typically have many more parameters in the optimizations process than data points, they are prone to exact interpolation by achieving a zero of the objective function. One might expect this interpolation leads to over-fitting and hence poor generalization power. However, in many empirical settings it has been shown that is not the case, to name a few see Belkin et al. (2019), Neyshabur et al. (2014).

One approach to avoid over-fitting in over-parameterized problems is to use explicit regularization. The two most common methods are L_1 and L_2 regularization. L_1 regularization corresponds to penalizing the sum of the absolute values of the parameters in the approximating function (similar to Lasso regression). L_2 regularization corresponds to penalizing the sum of the squared of parameters (similar to ridge regression). Surprisingly, it has been shown that deep learning can achieve good generalization without any implicit regularization.

These two observations have led researches to believe that optimization methods used in training deep neural networks possess intrinsic implicit regularization, for example see Neyshabur et al. (2017), Arora et al. (2019), and Smith et al. (2021). The implicit regularization in deep neural networks is still an open question. It is mostly believed that the implicit regularization is caused by first order optimization method such as gradient descent and stochastic gradient descent. It has been established the first order optimization methods favor flat minima in the space of parameters. In this context the flatness/sharpness of the minima is characterized by the trace of the Hessian matrix of the objective function, for more details see Blanc et al. (2020), Damian et al. (2021), Zhu et al. (2019), and Li et al. (2021).

Convergence to a flat minima in the space of parameters does not provide enough information about the properties of the function with respect to its inputs such as smoothness, low derivatives, etc. Due to the multiplicative structure of deep neural networks, Ma and Ying (2021) observe that, there is a tight connection between the flatness of a minima and the norm of the gradient of the function with respect to inputs (Sobolev semi-norm). More specifically, in an interpolation regime, stochastic gradient descent as an optimization algorithm implicitly penalizes the Sobolev norm of the approximating function. In other words deep neural networks along with their optimizers have a tendency to find approximating functions that have small derivatives (in our case not explosive).

It is worth mentioning the Maennel et al. (2018) study the case of a one hidden layer neural network with ReLU activation function ($\max\{0, x\}$) with "simple" low dimensional data points. They show, when the initial parameters are set to have small values, the gradient descent, among many solutions, converges to a data dependent (as opposed to network size dependence) linear interpolation which can be interpreted as interpolating function with low Sobolev semi-norms (small gradients with respect to input). Here we provide the mathematical definition of Sobolev $1-2$ semi-norm for a univariate function

Definition 1 (Sobolev 1-2 seminorm)**.** *Let $h : \Psi \to \mathbb{R}$ be a univariate function, Sobolev* 1-2 *seminorm is defied as*

$$\|\Psi\|_{1,2,\mathcal{X}} \equiv \left(\int_{\mathcal{X}} \left| \frac{d\Psi}{dx} \right|^2 dx \right)^{\frac{1}{2}}. \tag{95}$$

Most of the results in the literature focus on $\|\cdot\|_{1,2,\mathcal{X}}$, which is the L_2 norm of the derivatives (or weak derivatives for multivariate functions). However, this definition can be extended to include higher derivatives.[13]

For sequential models, where time is the input, $\mathcal{X}$ is a bounded interval of time, i.e., $\mathcal{X} = [0, T]$ for a positive T. For recursive models where the state is the input, $\mathcal{X}$ is bounded interval for capital, i.e., $\mathcal{X} \subset \mathbb{R}$.

Remark All the theoretical and empirical results in the literature regarding the implicit regularization in deep neural networks deal with regression and classification (when the outcome variables are binary) problems. The function approximation problems we face in macroeconomics and industrial organizations have a different nature and the objective functions (such as Bellman or Euler residuals) are slightly different from regression problems. However, as verified in the result sections of this paper, we strongly believe the same results can be proved for the sort of problems we face in macroeconomics.

[13] In general a Sobolev norm is defined by including the function and all the derivatives up to degree k. More formally, for a $p \in \mathbb{N}$ and $k \in \mathbb{N}$:

$$\|\Psi\|_{k,p,\mathcal{X}} \equiv \left(\sum_{i=0}^{k} \int_{\mathcal{X}} \left| \frac{d^i \Psi}{dx^i} \right|^p dx \right)^{\frac{1}{p}}, \tag{96}$$

note that the seminorm defined in Definition 1 does not have the positive definite property. That is why it is called a seminorm (for instance any constant function of the form $\Psi(x) = c$ has $\|\Psi\|_{1,2,\mathcal{X}} = 0$). However, definition (96) is a proper norm.

References

Arora, S., N. Cohen, W. Hu, and Y. Luo (2019): "Implicit regularization in deep matrix factorization," *Advances in Neural Information Processing Systems*, 32.

Belkin, M., D. Hsu, S. Ma, and S. Mandal (2019): "Reconciling modern machine-learning practice and the classical bias–variance trade-off," *Proceedings of the National Academy of Sciences of the United States of America*, 116, 15849–15854.

Blanc, G., N. Gupta, G. Valiant, and P. Valiant (2020): "Implicit regularization for deep neural networks driven by an Ornstein-Uhlenbeck like process," *Proceedings of Machine Learning Research vol*, 125, 1–31.

Damian, A., T. Ma, and J. D. Lee (2021): "Label noise sgd provably prefers flat global minimizers," *Advances in Neural Information Processing Systems*, 34, 27449–27461.

Ekeland, I. and J. A. Scheinkman (1986): "Transversality conditions for some infinite horizon discrete time optimization problems," *Mathematics of operations research*, 11, 216–229.

Fernández-Villaverde, J., J. F. Rubio-Ramírez, and F. Schorfheide (2016): "Solution and estimation methods for DSGE models," in *Handbook of macroeconomics*, Elsevier, vol. 2, 527–724.

Kamihigashi, T. (2005): "Necessity of the transversality condition for stochastic models with bounded or CRRA utility," *Journal of Economic Dynamics and Control*, 29, 1313–1329.

Li, Z., T. Wang, and S. Arora (2021): "What Happens after SGD Reaches Zero Loss?–A Mathematical Framework," in *International Conference on Learning Representations*.

Ma, C. and L. Ying (2021): "The Sobolev regularization effect of stochastic gradient descent," *arXiv preprint arXiv:2105.13462*.

Maennel, H., O. Bousquet, and S. Gelly (2018): "Gradient descent quantizes relu network features," *arXiv preprint arXiv:1803.08367*.

Neyshabur, B., R. Tomioka, R. Salakhutdinov, and N. Srebro (2017): "Geometry of optimization and implicit regularization in deep learning," *arXiv preprint arXiv:1705.03071*.

Neyshabur, B., R. Tomioka, and N. Srebro (2014): "In search of the real inductive bias: On the role of implicit regularization in deep learning," *arXiv preprint arXiv:1412.6614*.

Skiba, A. K. (1978): "Optimal growth with a convex-concave production function," *Econometrica: Journal of the Econometric Society*, 527–539.

Smith, S. L., B. Dherin, D. G. Barrett, and S. De (2021): "On the origin of implicit regularization in stochastic gradient descent," *arXiv preprint arXiv:2101.12176*.

Zhang, C., S. Bengio, M. Hardt, B. Recht, and O. Vinyals (2021): "Understanding deep learning (still) requires rethinking generalization," *Communications of the ACM*, 64, 107–115.

Zhu, Z., J. Wu, B. Yu, L. Wu, and J. Ma (2019): "The Anisotropic Noise in Stochastic Gradient Descent: Its Behavior of Escaping from Sharp Minima and Regularization Effects," in *International Conference on Machine Learning*, PMLR, 7654–7663.

Appendix A Sequential linear asset pricing

A.1 Learning the growth rate

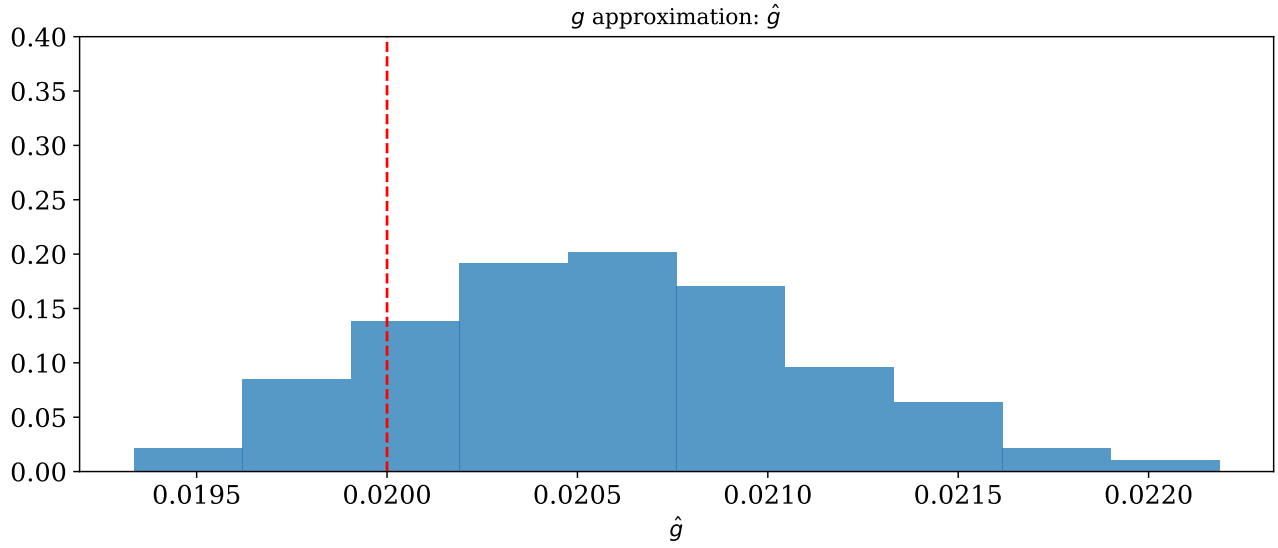

Figure 19: The distribution of the approximated growth rate of the dividends in the sequential linear asset pricing model. The dashed vertical line shows the actual value of g.

Figure 19 shows the distribution for the approximated growth rate for the sequential linear asset pricing model. The approximated growth rate is defined as

$$\hat{g} \equiv e^{\phi} - 1. \tag{A.1}$$

This figure shows the approximated growth rate for 100 seeds, the red vertical line shows the true growth rate, $g = 0.02$. The approximation is slightly biased to the right. However, given the optimization problem is extremely non-convex, the accuracy of approximation is impressive (average relative error around 2.5%). This results shows that the algorithm can separate the growth path from the stationary solution.

A.2 Misspecification of growth in prices

In this experiment we use a linear functional form to approximate the solution of the sequential linear asset pricing model with growing dividends

$$\hat{p}(t;\theta) = tNN(t;\theta_1) + \phi, \tag{A.2}$$

where $\theta \equiv \{\phi, \theta_1\}$, $NN(\cdot; \theta_1)$ is a deep neural network, and ϕ is a parameter needs to be found in the optimization process.

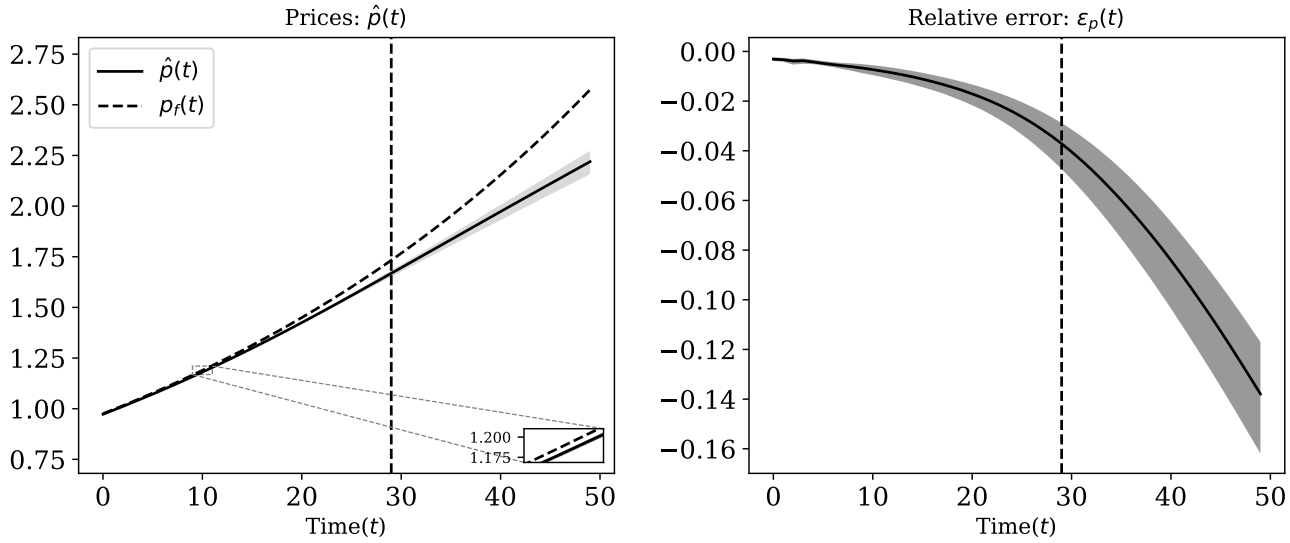

Figure 20: Comparison between the accurate and approximate solution for the sequential linear asset pricing model with growing dividends (i.e., $c = 0$, $g = 0.02$) in the presence of functional misspecification of the growth. The solid curve in the left panel shows the median of the approximate price paths over 100 seeds, the dashed curve shows the price based on the fundamentals. The solid curve in the right panel shows the median of the relative errors between the approximate price and the price based on the fundamentals over 100 seeds. The shaded regions show 10th and 90th percentiles. The dashed vertical lines separate the interpolation from the extrapolation region.

Figure 20 shows the results for sequential linear asset pricing model with growing dividends (i.e., $c = 0$, $g = 0.02$). We use the same neural network we utilize in the correctly specified form. The only difference is the linear term. In the left panel, the solid curve(denoted by $\hat{p}(t)$) shows the median of approximate price path over 100 seeds, the dashed curve (denoted by $p_f(t)$) shows the solution based on the fundamentals. The shaded regions show the 10th and 90th percentiles. The right panel shows the median of relative errors between the exact and approximate solution over 100 seeds, and the shaded regions show the 10th and 90th percentiles. The dashed vertical lines separate the interpolation from the extrapolation region.

This result shows even in the presence of misspecification of the functional form for the growth the long run errors do not impair the accuracy of short and medium run dynamics (at most 2% relative error after 10 periods). However, the misspecification can reduce the generalization power of the solution.

Appendix B Sequential neoclassical growth

B.1 Zero as a fixed point for capital

As discussed before $k = 0$ is a repulsive fixed point for the neoclassical growth model. In experiment we investigate whether the approximate solutions utilizing deep neural networks pick a capital path that converges to the repulsive fixed point due to numerical errors.

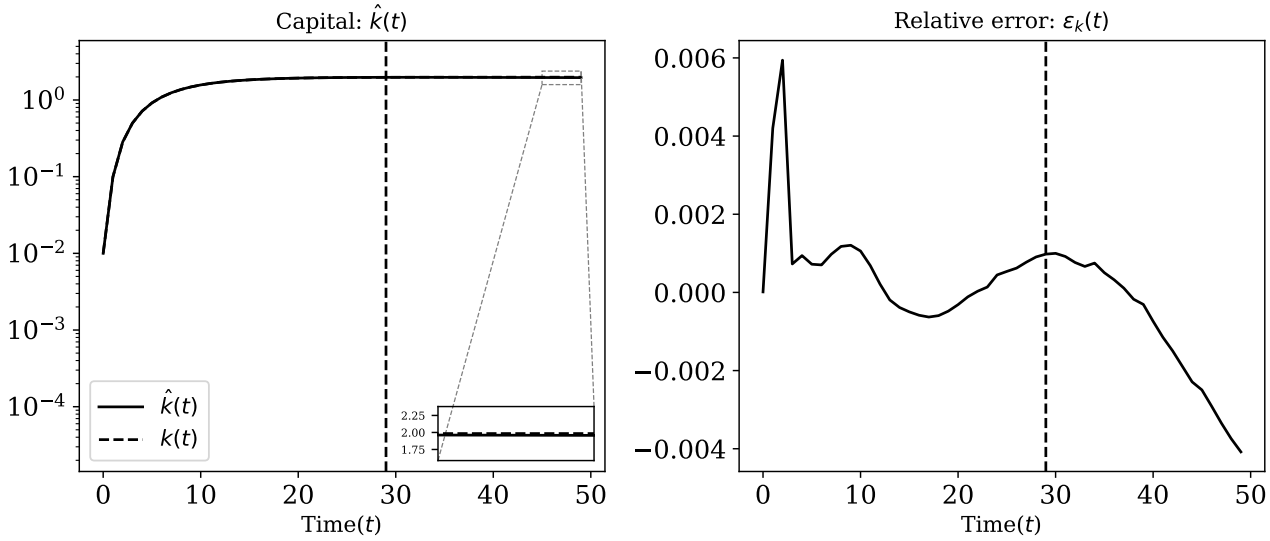

Figure 21: Comparison between the value function iteration and approximate solution using a deep neural network for the sequential neoclassical growth model for a small level initial capital $k_0 = 0.01$. In the left panel the solid curve shows the approximate path and the dashed curve shows the solution based obtained bu the value function iteration method. The right panel shows the relative errors for the capital paths. The dashed vertical lines separate the interpolation from the extrapolation region.

Figure 21 shows the result for the sequential neoclassical growth model for a small level of initial capital. In this experiment we use the same parameters and deep neural network as the sequential neoclassical growth model with no total factor productivity growth, except the initial condition for capital. Here we use $k_0 = 10^{-2}$. In the left panel the solid curve shows the approximate path and the dashed curve shows the solution obtained via the value function iteration method. The right panel shows the relative errors between the approximated capital path and the solution obtained via the value function iteration method. The dashed vertical lines separate the interpolation from the extrapolation region.

This result shows that even for a small levels of initial capital the solutions do not converge to $k = 0$. Therefore, the solutions can detect that k_0 is a repulsive fixed point. Moreover, the short and medium run solutions are accurate and are not impaired by the long run errors (at most 0.6%).

B.2 Solutions violating the transversality condition: Initial capital above the steady state

Figure 22 shows a set of solutions (blue curves) for capital, consumption and marginal utility of consumption denoted by $\tilde{k}(t)$, $\tilde{c}(t)$, and $u'(\tilde{c}(t))$ that satisfy the Euler equation and feasibility condition, but violate the transversality condition for the case of $k_0 = 10$. The black curves denoted by $k(t)$, $c(t)$, and $u'(c(t))$ show the optimal paths for capital, consumption and marginal utility of consumption that satisfy the Euler equation, feasibility condition and the transversality condition. The steady states for capital and consumption in the optimal solution are denoted by k^* and c^*. As evident in Figure 22 the capital path that violate the transversality condition

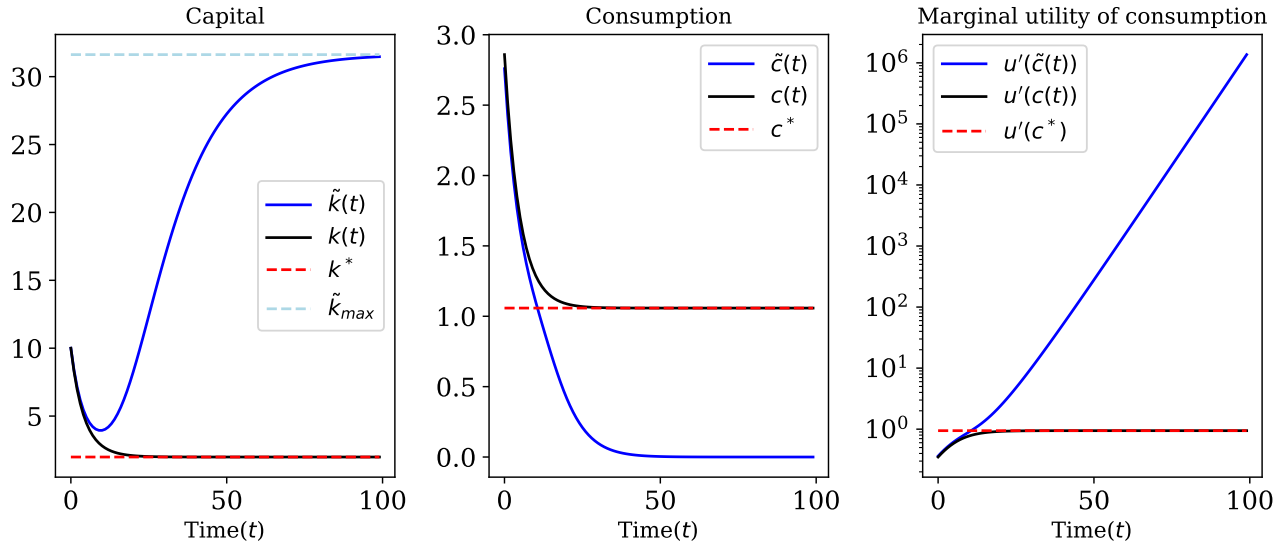

Figure 22: Comparison between the optimal solution and the solutions that violate the transversality condition for $k_0 > k^*$. The blue curves denoted by $\tilde{k}(t)$, $\tilde{c}(t)$, and $u'(\tilde{c}(t))$ show a set of capital, consumption, and marginal utility of consumption paths that violate the transversality condition. The black curves denoted by $k(t)$, $c(t)$, and $u'(c(t))$ show the capital, consumption, and marginal utility of consumption paths for the optimal solution. The steady states for capital an consumption are denoted by k^* and c^*.

has higher derivatives over its domain. More formally, let $\tilde{k}(t)$ be a capital path violating the transversality condition and $k(t)$ be the optimal solution, then in a compact space of the form $[0, T]$

$$\int_0^T \left| \frac{d\tilde{k}}{dt} \right|^2 dt > \int_, ^T \left| \frac{dk}{dt} \right|^2 dt. \tag{B.1}$$

Therefore, by Assumption 1

$$\|k\|_S < \|\tilde{k}\|_S. \tag{B.2}$$

B.3 An alternative way of approximating the optimal solution

Another approach to solve the sequential neoclassical growth model is to simultaneously approximate consumption and capital functions. In this case we pick a parametric space of functions $\mathcal{H}(\Theta)$, and a grid $\hat{X} = \{t_1, \ldots, t_N\}$ and find the the capital and consumption function $[k(t;\theta), c(t;\theta)] \in \mathcal{H}(\Theta)$ via the following optimization problem

$$\min_{\theta\in\Theta} \frac{1}{|\hat{X}|} \sum_{t\in\hat{X}} \left[\beta\big[z(t+1)^{1-\alpha} f'(k(t+1;\theta)) + 1 - \delta\big] - \frac{u'(c(t;\theta))}{u'(c(t+1;\theta))}\right]^2 +$$
$$\left[z(t)^{1-\alpha} f(k(t;\theta)) + (1-\delta)k(t;\theta) - c(t;\theta) - k(t+1;\theta)\right]^2 + \left[k(0;\theta) - k_0\right]^2,$$

where $z(t)$ is evaluated by the law of motion for z

$$z(t) = (1+g)^t z_0 \quad \text{for } t \in \hat{X}.$$

Non-negativity of capital and consumption are built into $\mathcal{H}(\Theta)$. Here we omit the results because they are almost identical to the results illustrated in Section 3.2.

B.4 Being far from the steady state

Figure 23 shows the results for the sequential neoclassical growth model with $\hat{X} = \{0, 1, \ldots, 4\}$. The dashed vertical lines separate the interpolation from the extrapolation region. The solid curve in the top-left panel shows the median of capital paths over 100 different random initialization of the parameters (seeds) of the deep neural network, the dashed curve shows the capital path obtained via the value function iteration method, and the shaded area shows the 10th and 90th percentiles. The solid curve in top-right panel shows the median of relative errors between the approximate capital paths and the capital path obtained via the value function iteration method, the shaded area shows the 10th and 90th percentiles. The solid curve in the bottom-left panel shows the median of consumption paths over 100 different random seeds, the dashed curve shows the consumption path obtained via the value function iteration method, and the shaded area shows the 10th and 90th percentiles. The dashed curve in bottom-right panel shows the median of relative errors between the approximate consumption paths and the consumption path obtained via the value function iteration method, the shaded area shows the 10th and 90th percentiles.

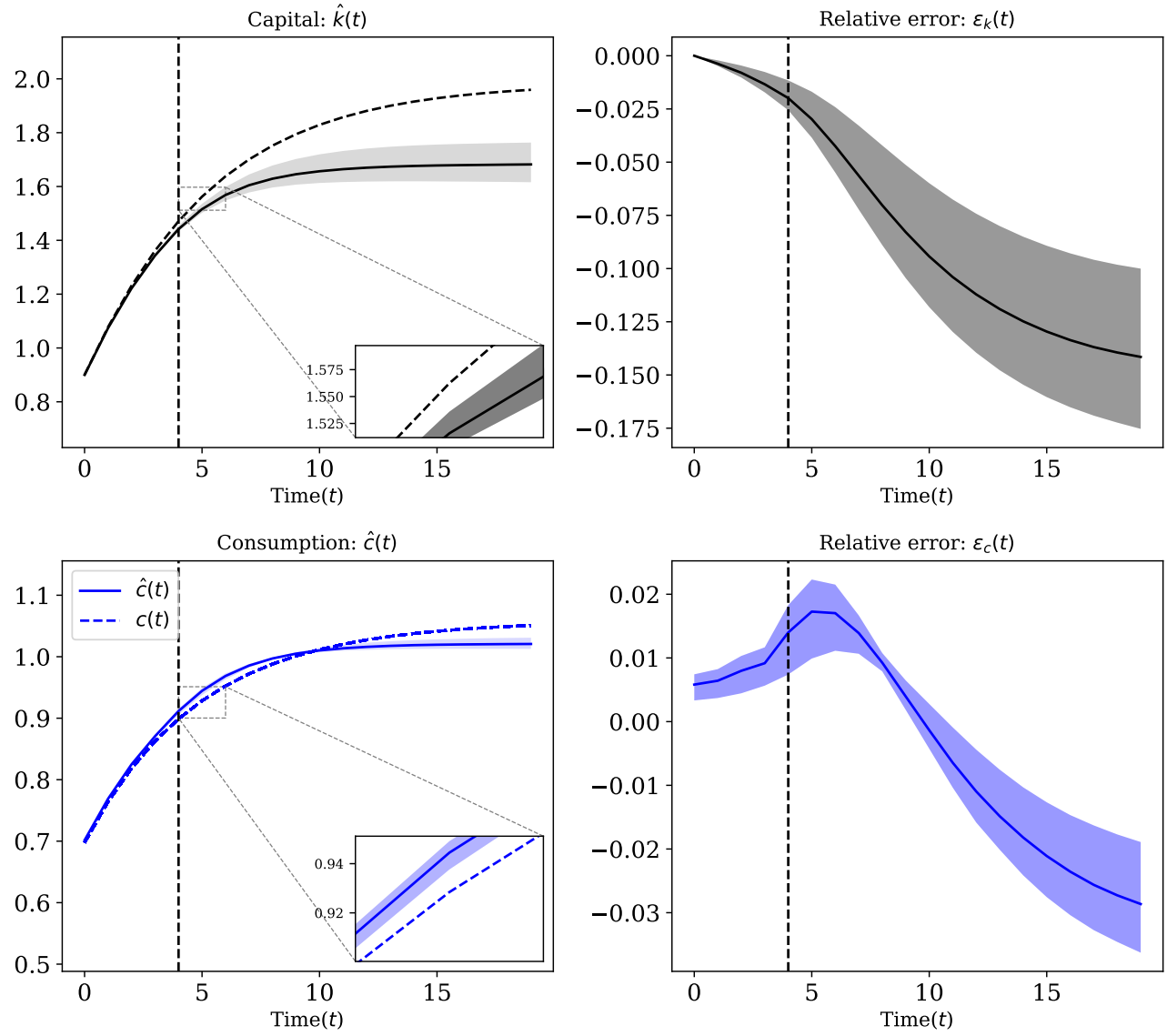

Figure 23: Comparison of between the value function iteration and approximate solution for sequential neoclassical growth model with short time horizons (i.e., $\hat{X} = \{0, 1, \ldots, 4\}$). The solid curves in the left panels show the median of the approximate capital and consumption paths over 100 seeds and the shaded regions show the 10th and 90th percentiles. The dashed curves show the capital and consumption paths obtained by the value function iteration method. The solid curves in the right panels show the median of the relative errors between approximate solutions and solutions obtained via the value function iteration for capital paths ($\varepsilon_k(t)$) and consumption paths ($\varepsilon_c(t)$). The shaded regions show the 10th and 90th percentiles. The dashed vertical lines separate the interpolation from the extrapolation region.

Theses results shows that the approximate solutions are robust to using short time horizons in $\hat{X}$. The long run errors do not impair the accuracy of short run dynamics (less than 1% relative errors in capital in the first three periods).

B.5 Misspecification of growth in capital and consumption paths

In this experiment we use a different functional form to approximate the solution of the sequential neoclassical growth model with growing total factor productivity

$$\hat{k}(t;\theta) = tNN(t;\theta_1) + \phi, \tag{B.3}$$

where $\theta \equiv \{\phi, \theta_1\}$, $NN(\cdot;\theta_1)$ is a deep neural network, and ϕ is a parameter needs to be found in the optimization process.

Figure 24 shows the results for sequential neoclassical growth for non-stationary total factor productivity (i.e., $g = 0.02$) in the presence of functional misspecification of the growth. We use the same neural network as before for $NN(\cdot;\theta_1)$. The only difference is the linear term. The solid curve in the top-left panel shows the median of approximate capital paths over 100 of different initialization (seeds) of the parameters of the neural network and the dashed curve shows the capital paths obtained via the value function iteration method, and the shaded area shows the 10th and 90th percentiles. The solid curve in the top-right panel shows the median of the relative errors between the approximate capital paths and the capital path obtained via the value function iteration method, and the shaded area show the 10th and 90th percentiles. The solid curve in the bottom-left panel shows the median of consumption paths over 100 different random seeds, the dashed curve shows the consumption path obtained via the value function iteration method, and the shaded area shows the 10th and 90th percentiles. The dashed curve in bottom-right panel shows the median of relative errors between the approximate consumption paths and the consumption path obtained via the value function iteration method, and the shaded area shows the 10th and 90th percentiles.

These results show that the long run errors do not impair the accuracy of the short and medium run dynamics even in the presence of functional misspecification of growth in total factor productivity. Moreover, the approximate solutions generalize well in the extrapolation region (less than 5% relative errors after 50 periods).

B.6 Analysis of the approximate solution in the vicinity of the bifurcation point

Figure 25 shows the capital and consumption paths for a grid of initial conditions $k_0 \in [0.5, 4]$ for the neoclassical growth model with the convex-concave production function. The top panel shows the capital paths and the bottom panel shows the consumption paths. The dashed horizontal lines show the steady states of capital k_1^* and k_2^* and their corresponding steady states for consumption c_1^* and c_2^*. The red trajectories show the approximate solutions near the bifurcation point. The non-monotonicity of the consumption paths in red (or equivalently the

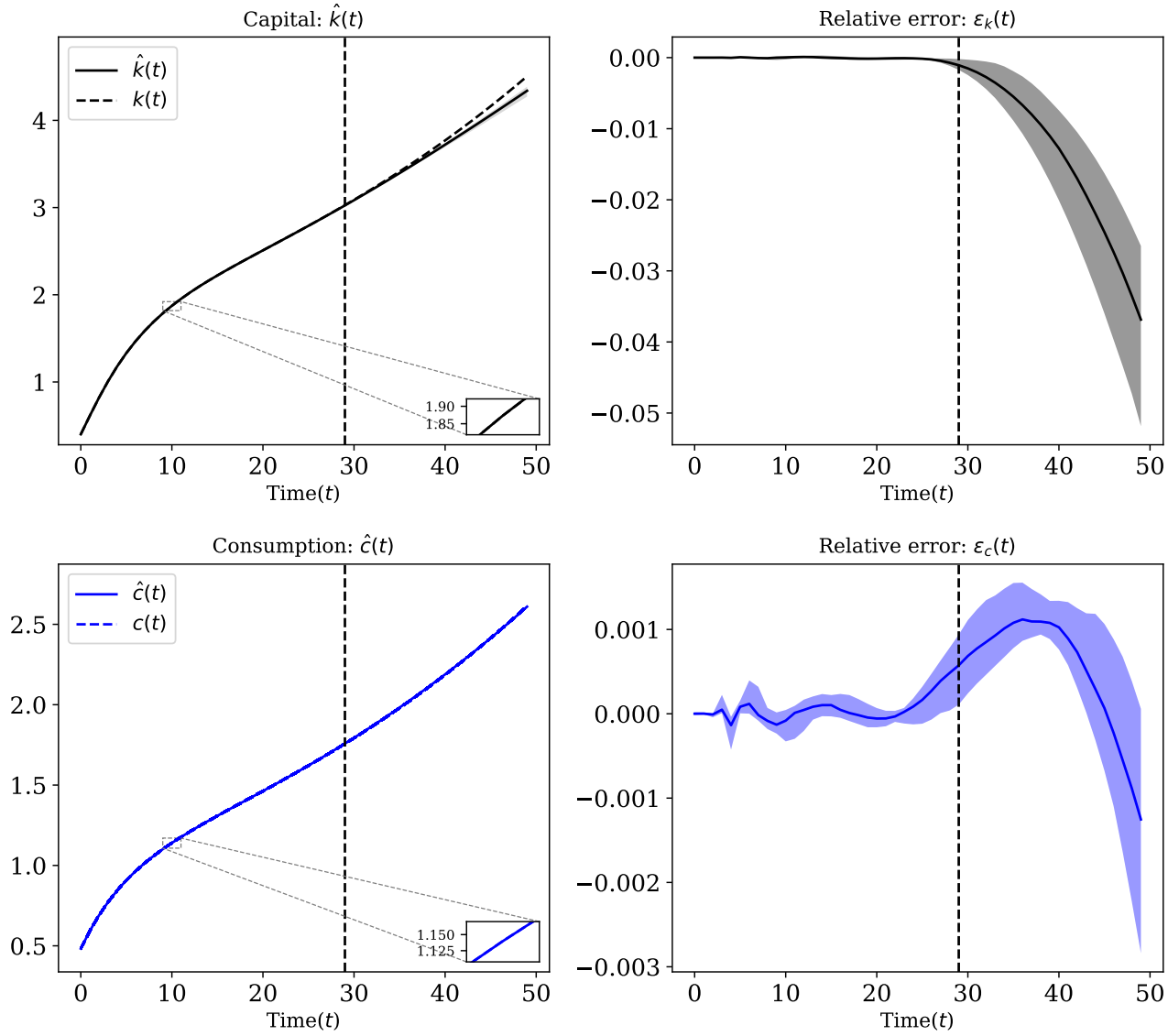

Figure 24: Comparison between the value function iteration and approximate solution using a deep neural network for the sequential neoclassical growth model with growth in total factor productivity in the presence of functional misspecification of the growth. The solid curves in the left panels show the median of the approximate capital and consumption paths over 100 seeds and the shaded regions show the 10th and 90 percentiles. The dashed curves show the capital and consumption paths obtained by the value function iteration method. The solid curves in the right panels show the median of the relative errors between approximate solutions and solutions via the value function iteration method for capital paths ($\varepsilon_k(t)$) and consumption paths ($\varepsilon_c(t)$). The shaded regions show the 10th and 90th percentiles. The dashed vertical lines separate the interpolation from the extrapolation region.

inflection points in capital paths) indicates that the these approximate solutions are not correct. However, the algorithm detects there is a bifurcation point in the space of initial points for capital around $k \approx 2.5$. Due to the discontinuity in the derivative of the production function it

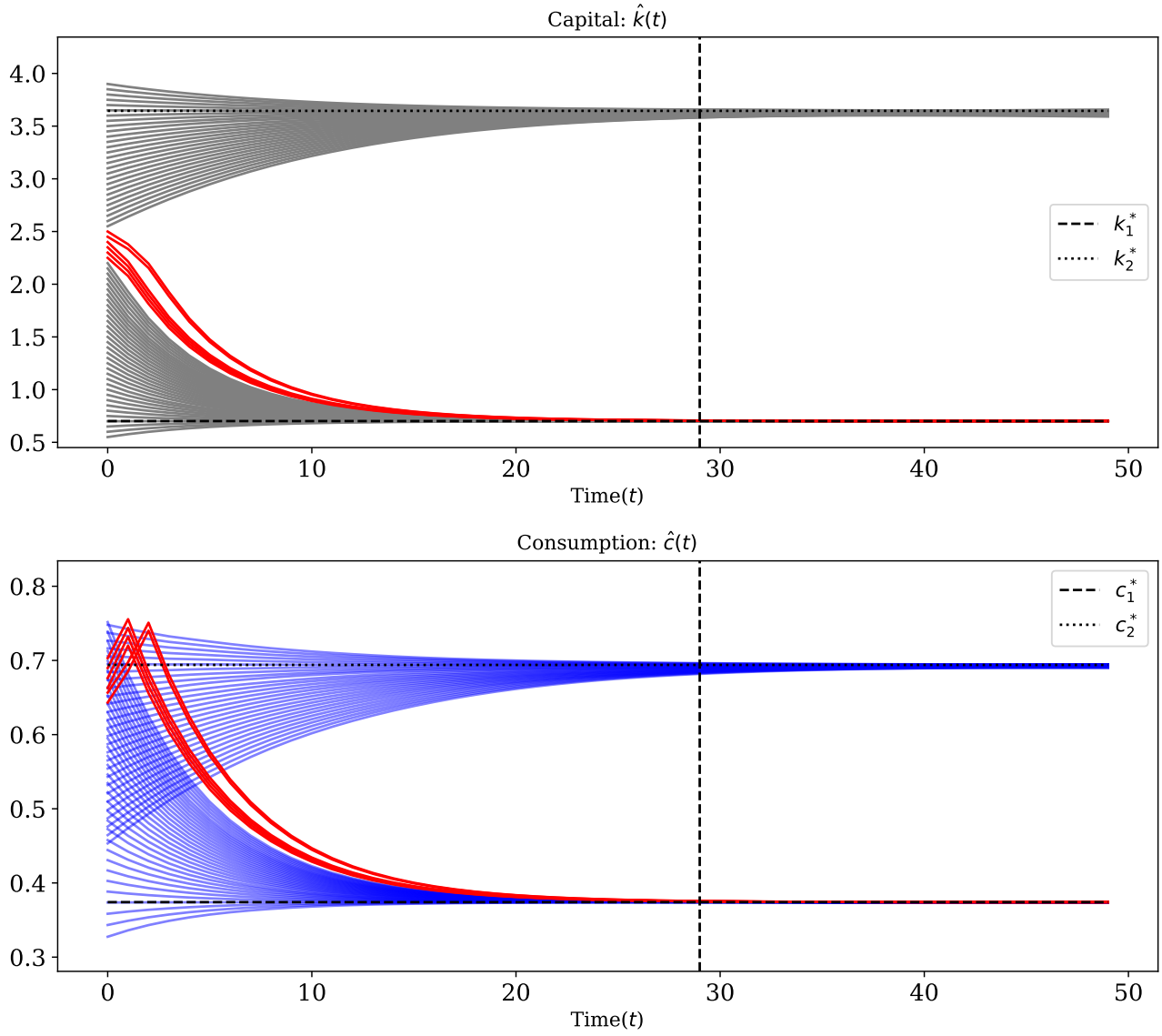

Figure 25: The approximate capital and consumption paths for the sequential neoclassical growth model with convex-concave production function. The gird for the initial condition of capital is from $[0.5, 4]$. The top panel shows the capital paths and the bottom panel shows the consumption paths. The red trajectories show the capital and consumption in the vicinity of the bifurcation point. The dashed horizontal lines show the steady states of capital k_1^* and k_2^* and their corresponding steady states for consumption c_1^* and c_2^*. The dashed vertical line separates the interpolation from the extrapolation region.

is cumbersome to confirm whether algorithm finds the right bifurcation point and it is beyond the scope of this paper.

Appendix C Recursive neoclassical growth

C.1 Robustness of the solution to the initial condition of capital

As stated in the recursive formulation of the neoclassical growth problem, the optimality of the solution requires that all the capital consumption paths generated by the policy function (i.e., $k'(k)$) to satisfy the the transversality condition.

Figure 26 shows the result of the minimization problem described in equation (73) for a range of initial conditions of capital. The solid curves $\hat{k}(t)$, in the top-left panel shows the capital paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k,1)$ from three initial conditions for capital outside of $\hat{X}$, $k_0 \in \{0.5, 3.25, 4.0\}$. The dashed curves (denoted by $k(t)$) show the capital paths obtained via the value function iteration method. The gray region shows the interpolation region $\hat{X} = [0.8, 2.5]$. The top-right panel shows the full range of relative errors between approximate capital paths and the capital paths obtained via the value function iteration method for 70 levels of initial capital, $k_0 \in [0.4, 4]$. The solid curves (denoted by $\hat{c}(t)$) in the bottom-left panel show the consumption paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k,1)$ and consumption function $c(k,1;\hat{k}')$ from three initial conditions for capital $k_0 \in \{0.5, 3.25, 4.0\}$. The dashed curves (denoted by $c(t)$) show the consumption paths obtained via the value function iteration method. The bottom-right panel shows the full range of relative errors between approximate consumption paths and the consumption path obtained via the value function iteration method for 70 levels of initial capital, $k_0 \in [0.4, 4]$.

These results show the approximate optimal policy function for capital satisfies the transversality condition in $\hat{X}$ and outside of $\hat{X}$.

Figure 27 shows the result of the recursive neoclassical growth problem for a range of initial conditions of capital in the presence of non-stationary total factor productivity (i.e., $g = 0.02$). The solid curves (denoted by $\hat{k}(t)$) in the top-left panel show the capital paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k,z)$ and $z' = (1+g)z$ from three initial conditions for capital $k_0 \in \{0.5, 3.25, 4.0\}$ and $z_0 = 1$. The dashed curves (denoted by $k(t)$) show the capital paths generated via the value function iteration method. The top-right panel shows the full range of relative errors between approximate capital paths and the capital paths obtained via the value function iteration method for 70 levels of initial capital $k_0 \in [0.4, 4]$. The solid curves (denoted by $\hat{c}(t)$) in the bottom-left panel show the consumption paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k,z)$, consumption function $c(k,z;\hat{k}')$ and $z' = (1+g)z$ from three initial conditions for capital $k_0 \in \{0.5, 3.25, 4.0\}$ and $z_0 = 1$. The bottom-right panel shows the full range of relative errors between approximate consumption paths and consumption path obtained via the value function iteration method for 70 levels of initial capital $k_0 \in [0.4, 4]$.

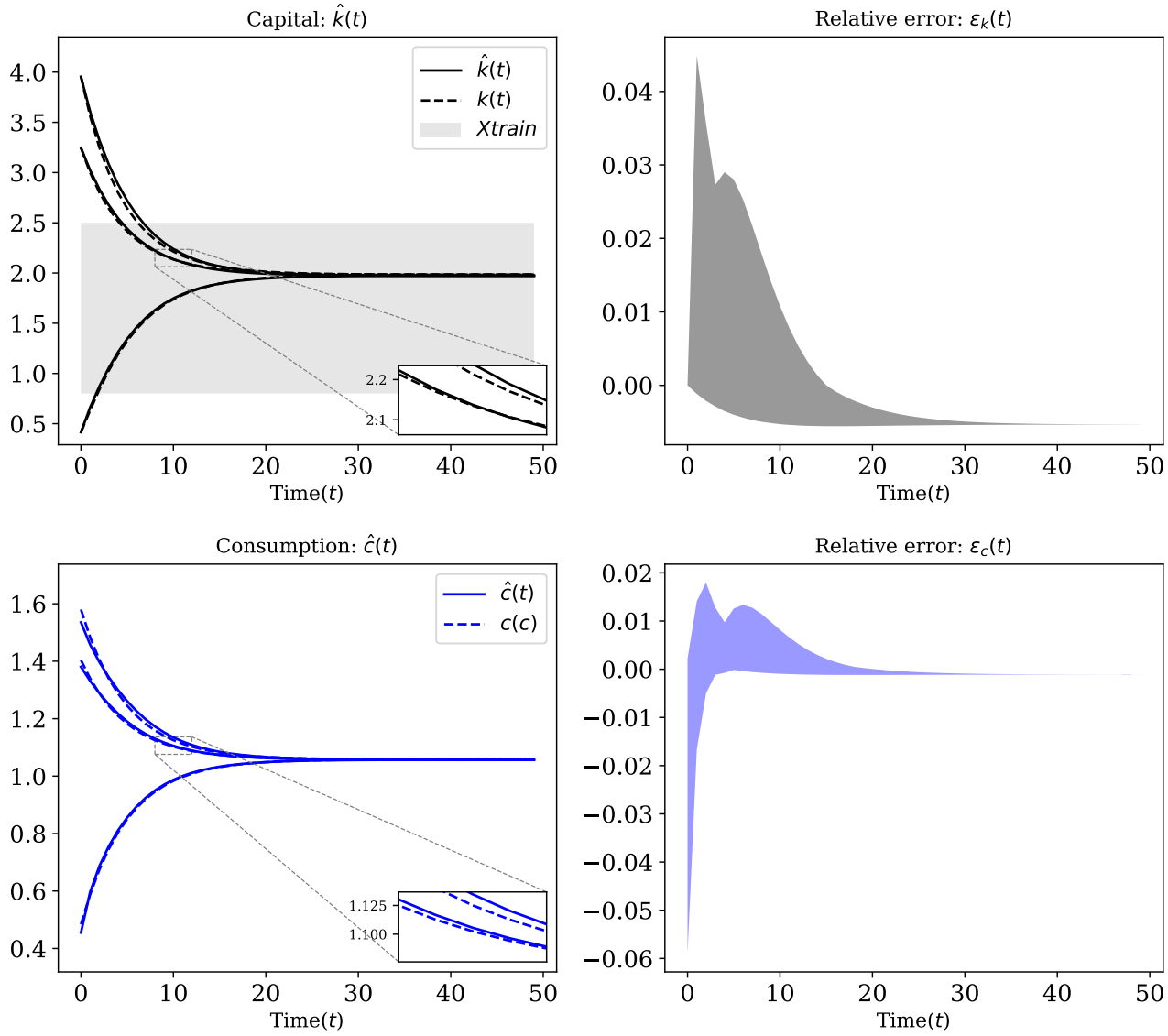

Figure 26: Comparison between the value function iteration and approximate solution using a deep neural network for the recursive neoclassical growth model for a wide range of initial levels of capital. The left panels show capital and consumption paths generated by iterating forward the approximate optimal policy function for capital $\hat{k}'(k, 1)$ and consumption function $c(k, 1; \hat{k}')$ from three initial conditions outside of $\hat{X}$, $k_0 \in \{0.5, 3.25, 4.0\}$. The dashed curves (denoted by $k(t)$ and $c(t)$) show the capital and consumption paths obtained via the value function iteration method. The right panels shows the full range of relative errors for capital and consumption paths for 70 initial conditions of capital in $[0.4, 4]$.

These results show that even in the presence of non-stationary, the approximate optimal policy function for capital satisfies the transversality condition in $\hat{X}$ and outside of $\hat{X}$.

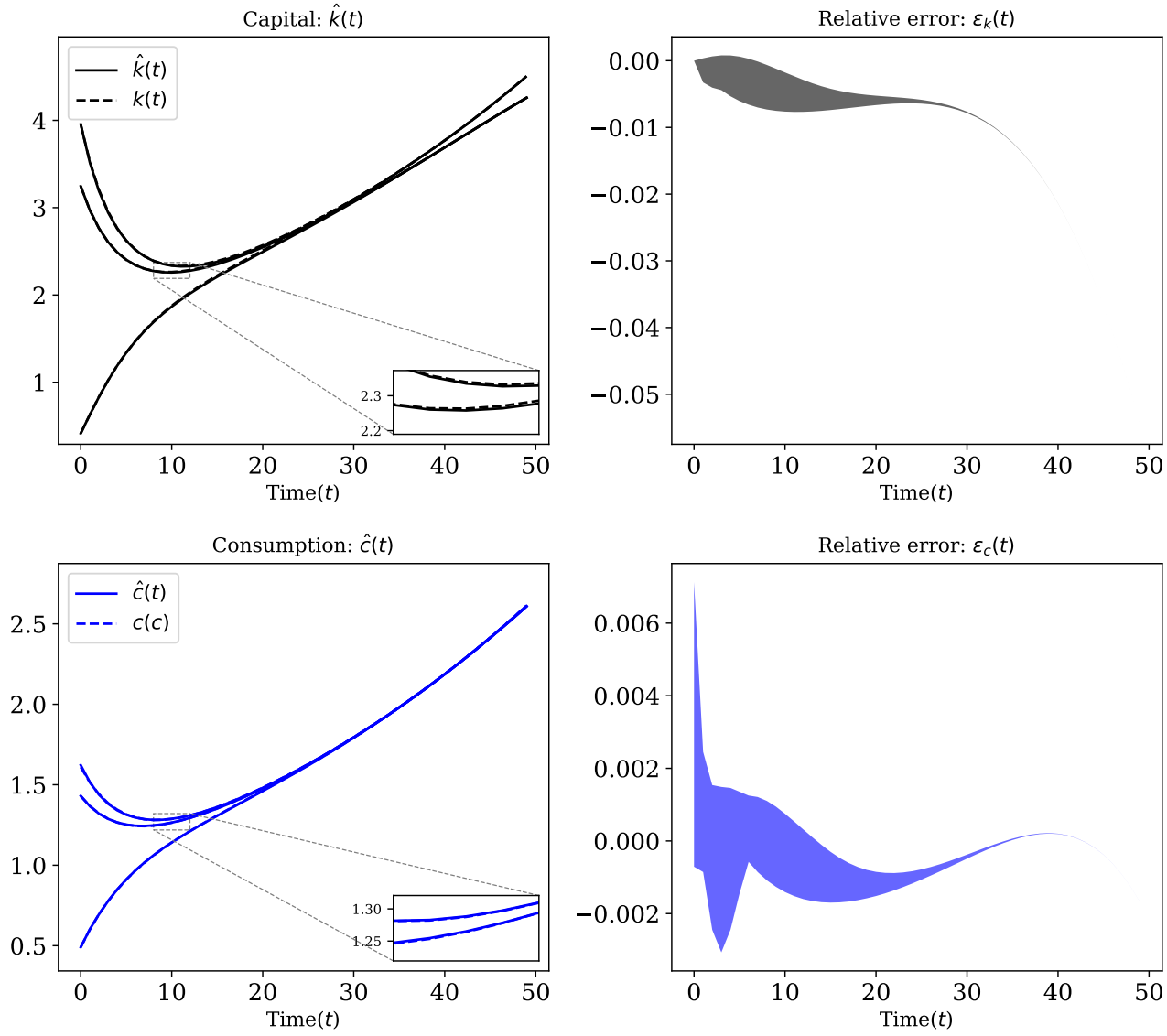

Figure 27: Comparison between the value function iteration and approximate solution using a deep neural network for the recursive neoclassical growth model with growth in total factor productivity (i.e., $g = 0.02$) for a wide range of initial levels of capital. The left panels show capital and consumption paths (denoted by $\hat{k}(t)$ and $\hat{c}(t)$) generated by iterating forward the approximate optimal policy function for $\hat{k}'(k, z)$, consumption function $c(k, z; \hat{k}')$, and $z' = (1+g)z$ from three initial conditions $k_0 \in \{0.5, 3.25, 4.0\}$ and $z_0 = 1$. The dashed curves (denoted by k(t) and $c(t)$) show the solutions obtained via the value function iteration method. The right panels shows the full range of relative errors for capital and consumption paths for 70 initial conditions of capital in $[0.4, 4]$.