Report

This is the report of this Project: Machine Learning for Economics.

1 Results

In this section, we will replicate some results of the Childers(22).

Since the limitation of time and computation resource, I only replicate a part of result of the first case, RBC model:

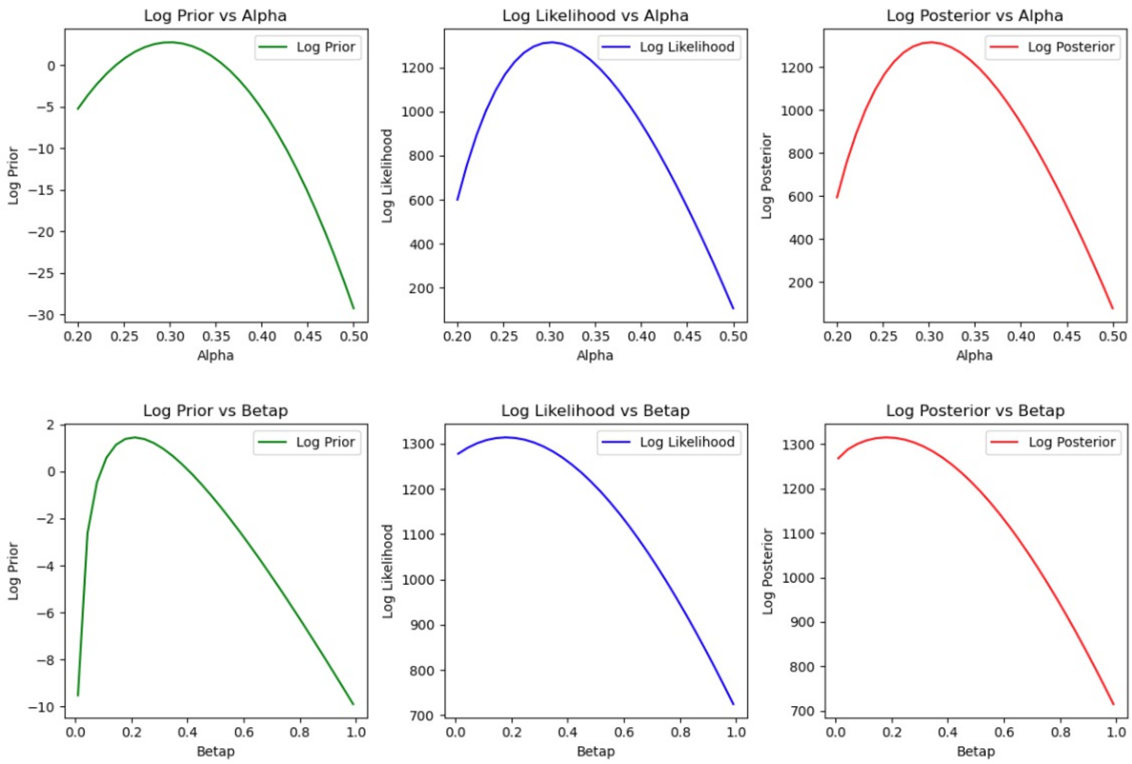
$$
\begin{aligned}
\frac{1}{c_t} - \beta \frac{\alpha e^{z_{t+1}} k_{t+1}^{\alpha-1} + (1-\delta)}{c_{t+1}} &= 0 \\
c_t + k_{t+1} - (1-\delta)k_t - y_t &= 0 \\
y_t - e^{z_t} k_t^{\alpha} &= 0 \\
z_{t+1} - \rho z_t - \sigma \varepsilon_{t+1} &= 0 \\
100\left(\frac{1}{\beta} - 1\right) &= \beta_{\text{draw}}
\end{aligned}
$$

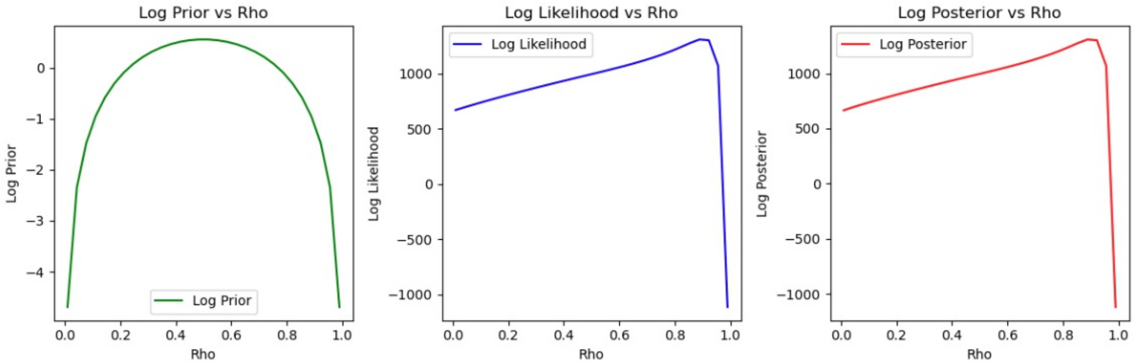
The pseudotrue parameter values are

$$\alpha = 0.3, \beta_{\text{draw}} = 0.2, \rho = 0.9, \delta = 0.025, \sigma = 0.01$$

and choose α, β, ρ as the unknown parameters, the observables are $c_{\text{obs}} = c_t - \bar{c}$ and $i_{\text{obs}} = i_t - \bar{i}$, which are the consumption and investment deviation from the deterministic steady state.

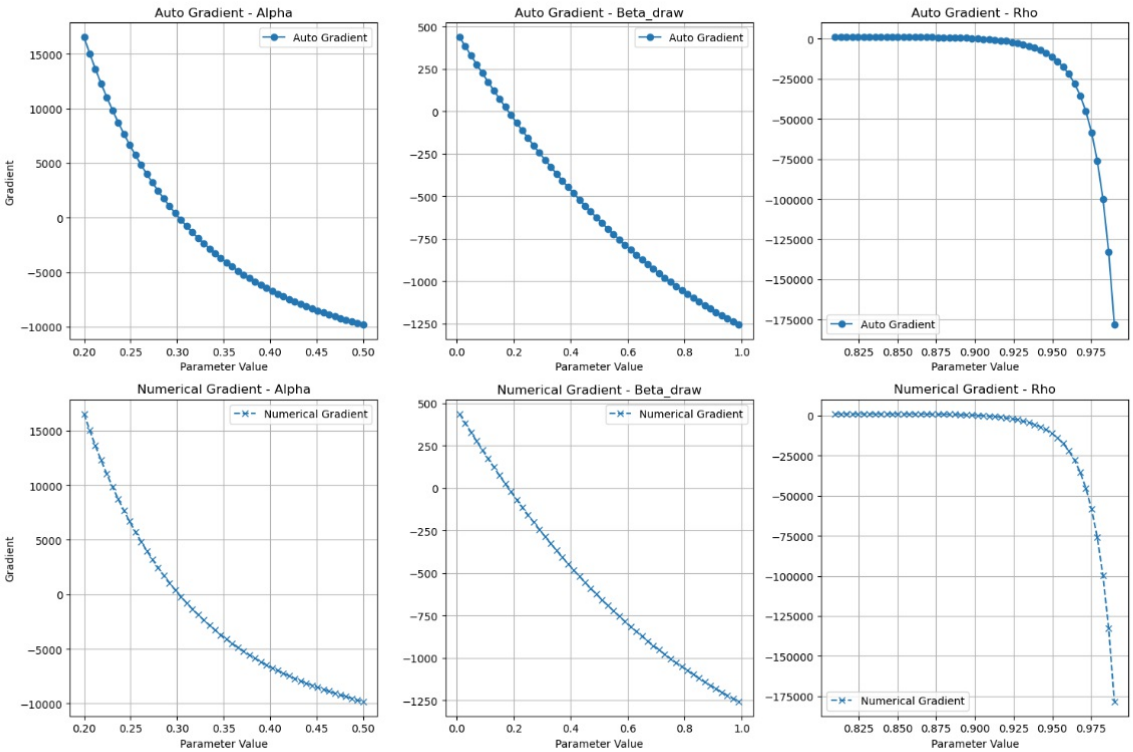
Firstly, the marginal likelihood for the three parameter are





There are three figures for each parameter: log prior, log likelihood, and log posterior. We can see that the mode of the three log posterior is the true value: $\alpha = 0.3$, $\beta_{\text{draw}} = 0.2$, $\rho = 0.9$.

The following is the gradient plots for three parameters, to exam our auto gradient by Tensorflow, we also calculate the numerical gradient using central difference. We can see that the two gradients are matched.



1. Sampling 110000 samples from the RWMH with marginal posterior of the first-order RBC model and discard the first 11000 samples.

Parameters	Pseudotrue	Post. Mean	Post. Std	ESS	Time
α	0.3	0.30596623	0.02703978	510.3	122min
β_{draw}	0.2	0.21592883	0.1799922	306.2	122min
ρ	0.9	0.90466733	0.01238041	4854.2	122min

Table 1.

2. Sampling 2500 samples from the NUTS with marginal posterior of the first-order RBC model and discard the first 500 samples.

Parameters	Pseudotrue	Post. Mean	Post. Std	ESS	Time
α	0.3	0.30281183	0.00724254	332.3	7400min
β_{draw}	0.2	0.19466405	0.0430717	279.7	7400min
ρ	0.9	0.90932818	0.00480159	1504.4	7400min

Table 2.

Secondly, sample results from joint likelihood for the three parameter are

1. Sampling from the NUTS with joint posterior of the first-order RBC model

Failed, the ESS is near zero, for 203-dimensional sampling, RWMH can not sample the effective samples in reasonable time steps.

2. Sampling 2500 samples from the NUTS with joint posterior of the first-order RBC model and discard the first 500 samples.

Parameters	Pseudotrue	Post. Mean	Post. Std	ESS	Time
α	0.3	0.30772509	0.009914077	15.3	2989min
β_{draw}	0.2	0.18970959	0.072409782	24.1	2989min
ρ	0.9	0.89054141	0.00854782	134.5	2989min

Table 3.

2 Answers to questions

1. Are there any typos and errors in the paper?

   a) In 6.1, page20: $\sigma = 0.1$ may be a typo, since I used this value cannot produce the convergent RWMH sequence. The true $\sigma = 0.01$ which is a reasonable value, then the variance is 10^{-4} can be compared with the measurement variance 10^{-5}.

   b) In appendix B.2, page53: The first term in the sylvester equation shoule be

$$\begin{bmatrix} \frac{d\mathcal{H}_{y'}}{d\theta_i} & \frac{d\mathcal{H}_{y}}{d\theta_i} & \frac{d\mathcal{H}_{x'}}{d\theta_i} & \frac{d\mathcal{H}_{y}}{d\theta_i} \end{bmatrix} = \begin{bmatrix} \frac{d\mathcal{H}_{y'}^T}{d\theta_i} \\ \frac{d\mathcal{H}_{y}^T}{d\theta_i} \\ \frac{d\mathcal{H}_{x'}^T}{d\theta_i} \\ \frac{d\mathcal{H}_{y}^T}{d\theta_i} \end{bmatrix}^T$$

2. Are there any parts that are unclear?

   Many parts are unclear, such as the perturbation, HMC, and the formula for Kalman filter.

3. Other than replicating their results, can you write a short note to explain some of the difficult points that may be unclear to readers unfamiliar with the subject?

   Refer to DSGE in a Nutshell.

4. Can we apply the method in Childers(22)? If yes, can you write a program to show the results? If no, what can we do?

   Yes, we can apply the method to the recursive utility.

5. Is the Caldara, Fernandez-Villaverde, Rubio-Ramirez and Yao (2012) paper above useful?

   Yes, we can use the method in the paper to compute the perturbation solution.

6. Do you think the neural network method in Maliar's paper is helpful? Can we train a neural network solution which takes the parameters of the DSGE model as input so that we do not have to retrain the neural network during estimation?

   Yes, we can train a neural network to take the parameters of a DSGE model as input along with current state variables and shocks. This approach allows the network to learn the mapping from both the parameters and states to the optimal decisions, encapsulating the entire policy function within the network. Consequently, this precludes the need to retrain the network when performing estimation tasks for different parameter values within the trained range.

   However, incorporating model parameters as inputs increases the complexity of the neural network and may complicate the training process. Moreover, it requires a sufficient coverage of the parameter space during training to ensure accurate predictions for new parameter values. Advanced generative models like Conditional GANs or Conditional Diffusion Models can effectively handle conditional probability distributions and may be suitable alternatives for capturing the relationships between model parameters and policy functions. Nevertheless, these sophisticated models come with their own set of challenges related to training difficulty, computational cost, and model evaluation.