DANMARKS TEKNINSKE UNIVERSITET

ADVANCED TIME SERIES ANALYSIS
Course number: 02427

Computer Exercise 2

Authors:
Mads Esben Hansen, s174434
Karl Emil Takeuchi-Storm, s130377

November 3, 2020

Introduction

This is answer to "Computer Exercise 2" in the course in advanced times series analysis at DTU. The exercise is divided into 4 parts that will be answered independently. The subject of the exercise is parameter estimation for various non-linear models. Whenever there is a referral to *the lecture notes*, the notes in mention is *Modelling Non-Linear and Non-Stationary Time Series*, by Henrik Madsen and Jan Holst, December 2006.

Part 1

In this exercise the goal is to estimate the parameters of a simulated model. For this purpose the SETAR(2,1,1) model we used in computer exercise 1 is chosen, and 1000 points are simulated. In order to estimate the parameters the SETAR model is implemented using variables for the parameters, such the error magin between the model and the data can be minimized.

In order to find the local minimum, with five free variables, the r-function optim() is used. As five free parameters can render a lot of local minima, the optimization is done in two steps. An initial step using "simulated annealing" algorithm, followed by the BGFS algorithm, to get a more precise result.

As can be seen in below figure 1, the estimate for the conditional mean is very good.

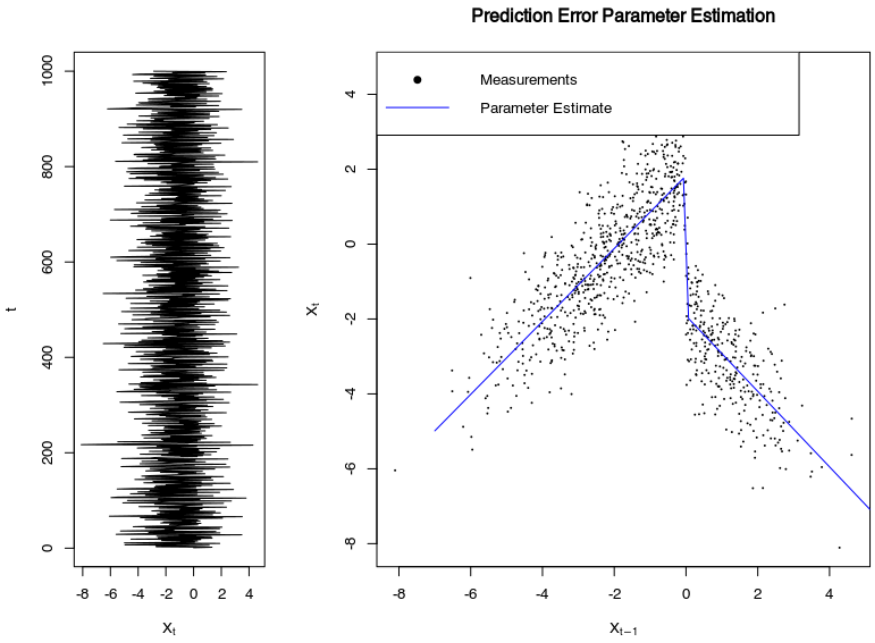


Figure 1: Parameter estimation of all five parameters of a SETAR(2,1,1) model.

Using modern optimization algorithms, makes it quite easy to estimate parameters of simple model like the SETAR(2,1,1), given by:

$$X_t = \begin{cases} \theta_1 + \theta_2 \cdot X_{t-1} + \epsilon_t^1 & X_{t-1} \leq \theta_5 \\ -\theta_3 - \theta_4 \cdot X_{t-1} + \epsilon_t^2 & X_{t-1} > \theta_5 \end{cases} \tag{1}$$

As shown in below table 1, the results are very promising, as the simulation results might actually have slightly different values for the parameters, due to the introduced noise.

Table 1: Numeric results of parameter estimation for 1000 simulated points

	θ_1	θ_2	θ_3	θ_4	θ_5
Model	2	1	-2	-1	0
Estimate	1.93	1.01	-1.98	-1.03	0.00

In order to check if the method is as good as above results indicate, the same procedure is carried out, but on 100.000 simulated data points. The resulting parameter estimates are given in below table 2, which is computationally very heavy. Price comes with a big gain, as the parameters are estimated to a precision of one thousand of the original value.

Table 2: Numeric results of parameter estimation for 100.000 simulated points

	θ_1	θ_2	θ_3	θ_4	θ_5
Model	2	1	-2	-1	0
Estimate	2.00	1.00	-2.00	-1.00	0.00

Parameter estimation using the prediction error method is quite feasible, however as the number of parameters increase so does the need for computation time. The advantage of using simulated annealing, is that the process parallelize extremely well, though being demanding in itself.

Part 2

In this part we will look at the relationship between the squared residuals, Q_N, and two arbitrary chosen parameter estimates, θ_i and θ_j. For simplicity we will limit ourselves to the parameters for the first regime, i.e. θ_1 and θ_2. Hence we look at the function given by $Q_N(\theta_1, \theta_2)$.

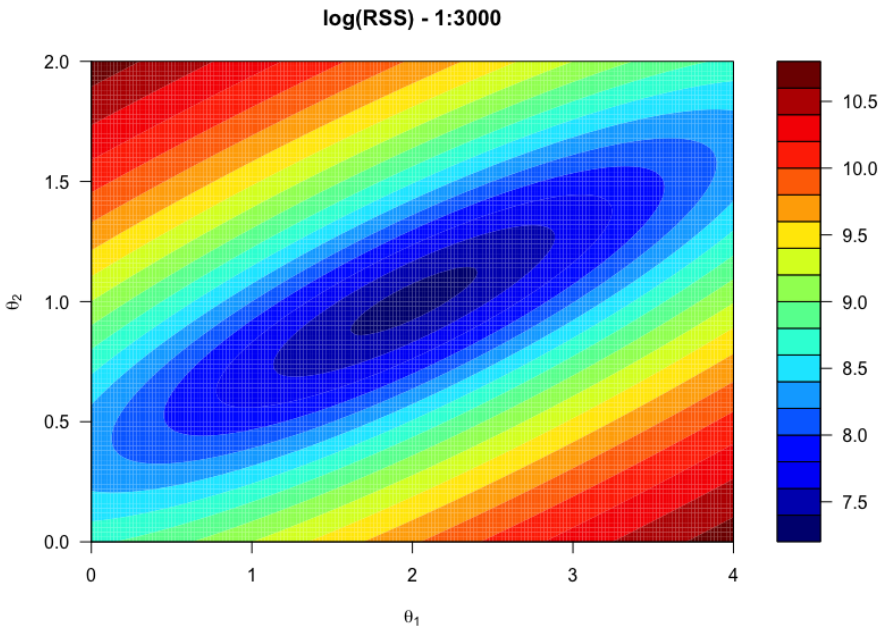


Figure 2: $Q_N(\theta_1, \theta_2)$ based on simulation points 1:3000.

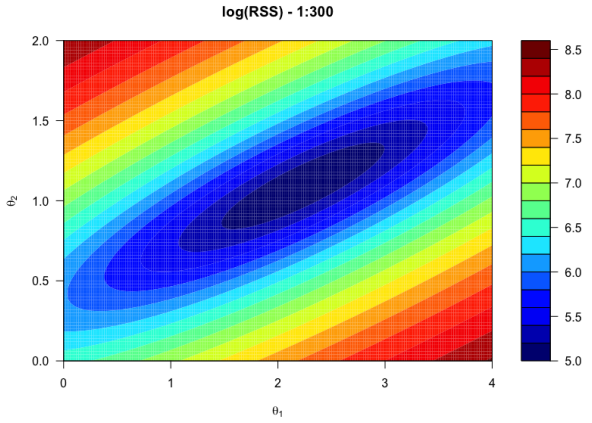


Figure 3: $Q_N(\theta_1, \theta_2)$ based on simulation points 1:300.

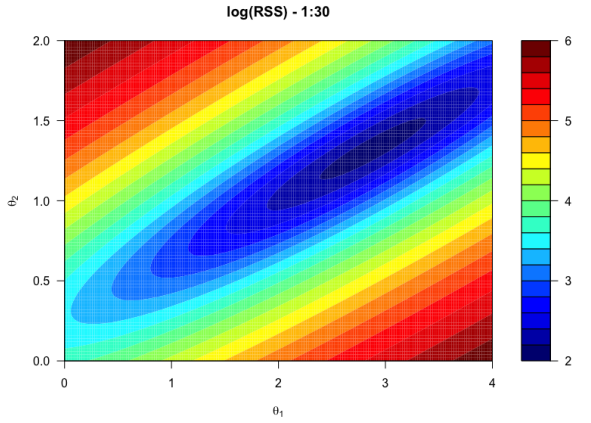


Figure 4: $Q_N(\theta_1, \theta_2)$ based on simulation points 1:30.

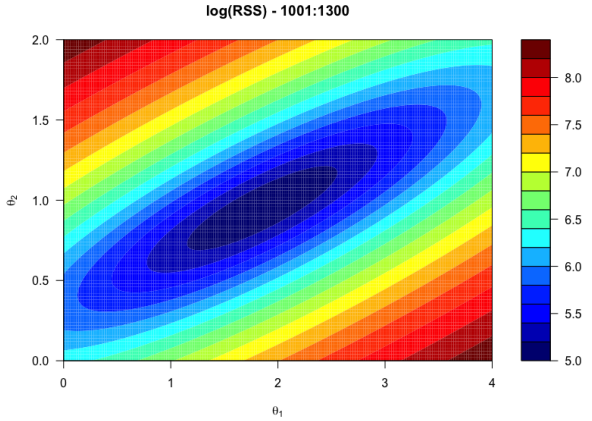


Figure 5: $Q_N(\theta_1, \theta_2)$ based on simulation points 1001:1300.

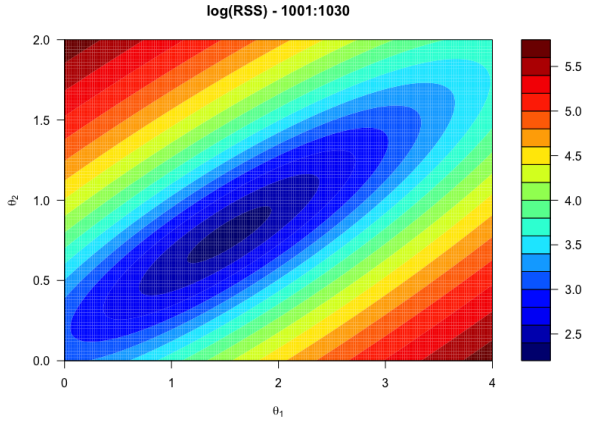


Figure 6: $Q_N(\theta_1, \theta_2)$ based on simulation points 1001:1030.

From these plots it is quite evident that the minimum is in the neighborhood of $(\theta_1, \theta_2) = (2, 1)$ in all cases, which is as expected. What is more interesting is the fact that there seem to be some correlation between θ_1 and θ_2 indicating, that is if θ_1 is high, a high value of θ_2 will result is a lower Q_N than if θ_2 had been low. We can further see that the number of points used in the estimating has an effect on the center of the ellipse. It seems that a larger number of points used results in an ellipse with a center closer to the expected of $(2, 1)$, i.e. a larger simulation set results in parameter estimates closer to the true values.

Part 3

In this section an ARMA(2,1) model is implemented as an doubly stochastic state space model. The model is given by:

$$X_t^1 = -\phi_1 \cdot X_{t-1}^1 + \cdot X_{t-1}^2 + r_t \tag{2}$$

$$X_t^2 = -\phi_2 \cdot X_{t-1}^1 + \theta_t \cdot r_t \tag{3}$$

$$Y_t = X_t \tag{4}$$

where:

$$\phi_1 \sim N(0.3, 0.5), \phi_2 \sim N(0.7, 0.25), \theta_1 \sim N(0.2, 1), r_t \sim N(0, 1) \tag{5}$$

Simulating 1000 points from the model gives

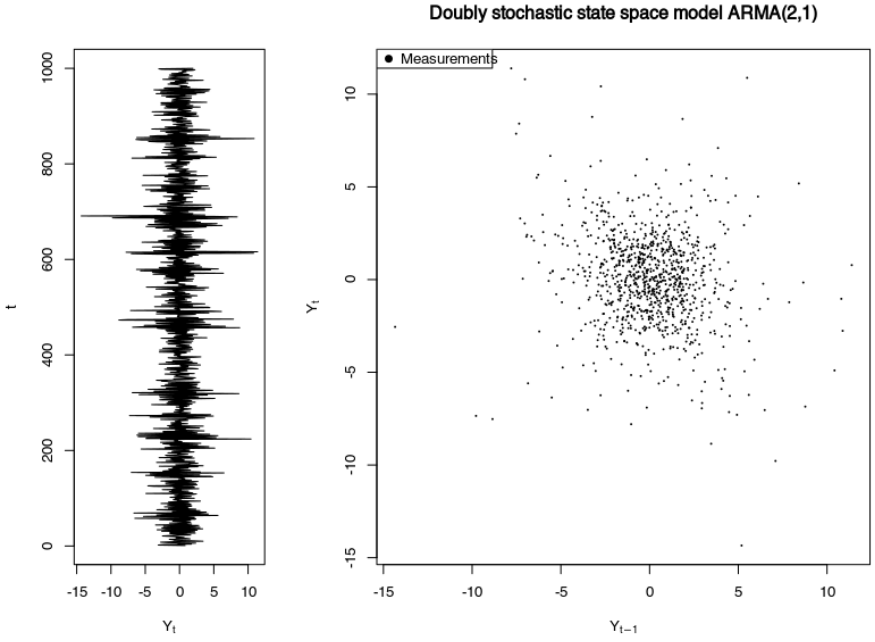


Figure 7: Simulation of 1000 outcomes from the doubly stochastic model

I order to keep the process mostly stationary, the mean and variance is kept rather low. The corresponding AR-model would be stationary, and our model is chosen such it does not display the huge instability periods seen with higher variance on the ϕ-parameters.

Part 4

Let consider the state space model given by:

$$x_{t+1} = a \cdot x_t + v_t \tag{6}$$
$$y_t = x_t + e_t \tag{7}$$

where a is an unknown parameter, and v_t and e_t are mutually uncorrelated white noise processes with variances σ_v^2 and σ_e^2 respectively.

4A

Initially we can simulate this model to see what it looks like.

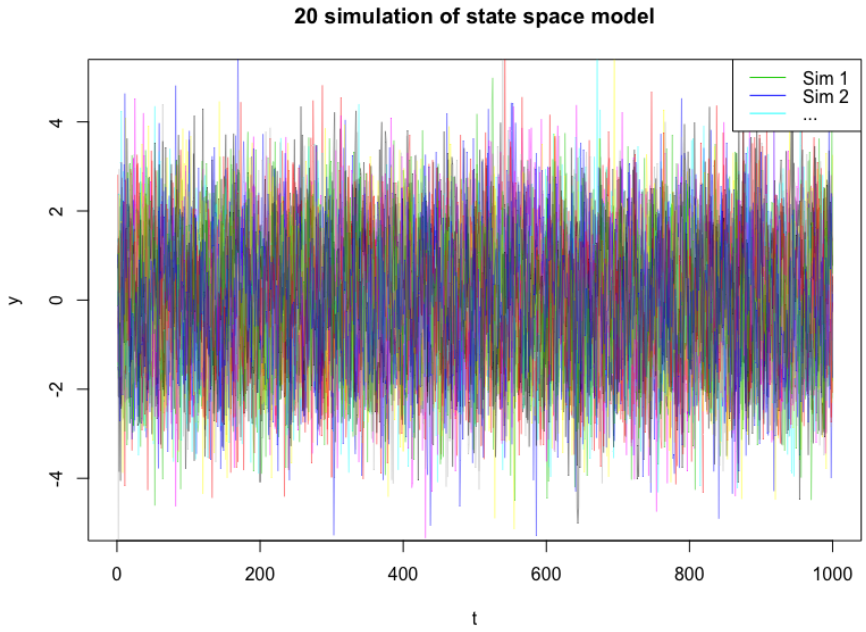


Figure 8: 20 simulation of state space model with n=1000.

In figure 8 we can see that the process seems to be stable. Apart from this it is difficult to say anything else based on this figure. We can rewrite the state space model such that the parameter a is included in the state space by:

$$\begin{bmatrix} x_t \\ a_t \end{bmatrix} = \begin{bmatrix} a_{t-1} & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{t-1} \\ a_{t-1} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \cdot v_t \tag{8}$$

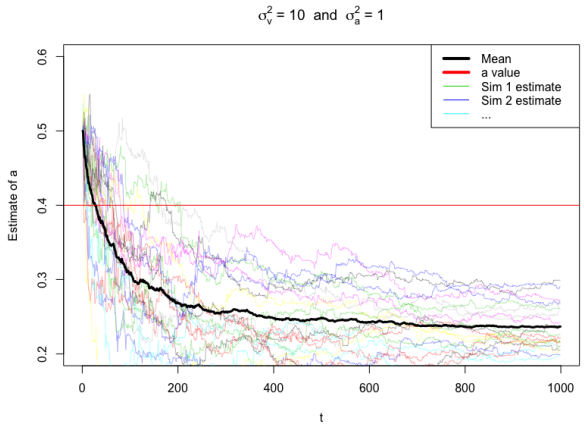
$$y_t = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_t \\ a_t \end{bmatrix} + e_t \tag{9}$$

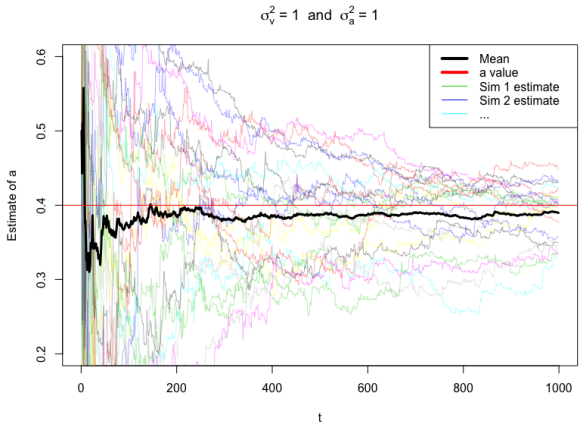
This reformulation allows us to estimate the parameters in a state space model using an Extended Kalman Filter (EKF) which is described in section 7.4 in *the lecture notes*.

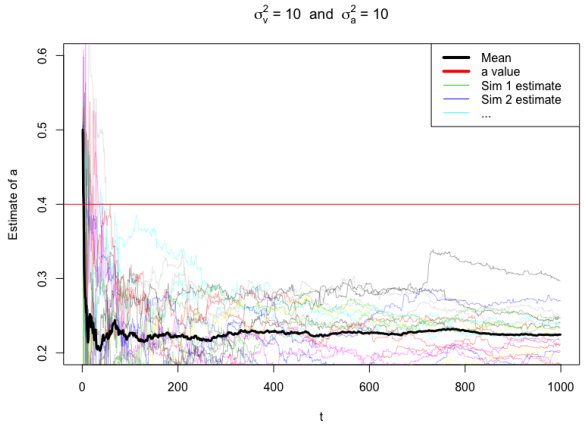
4B

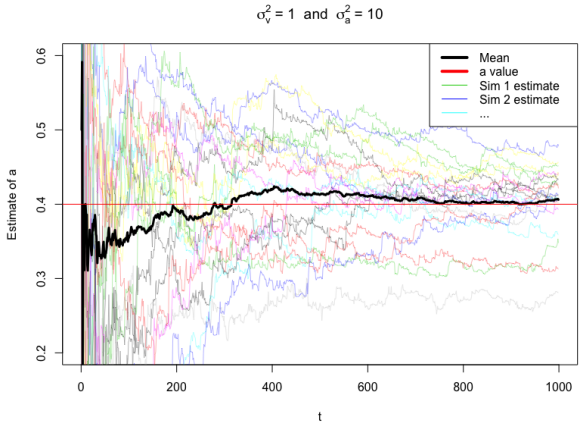
In this part we will dig further into the usage of the EKF. We will estimate a_t using EKF using
$a_{init} \in \{-0.5, 0.5\}, \quad \sigma_v^2 \in \{1, 10\}, \quad \sigma_a^2 \in \{1, 10\},$
i.e. 8 permutations, and investigate the asymptotic behaviour of a_t.

Initial a=0.5

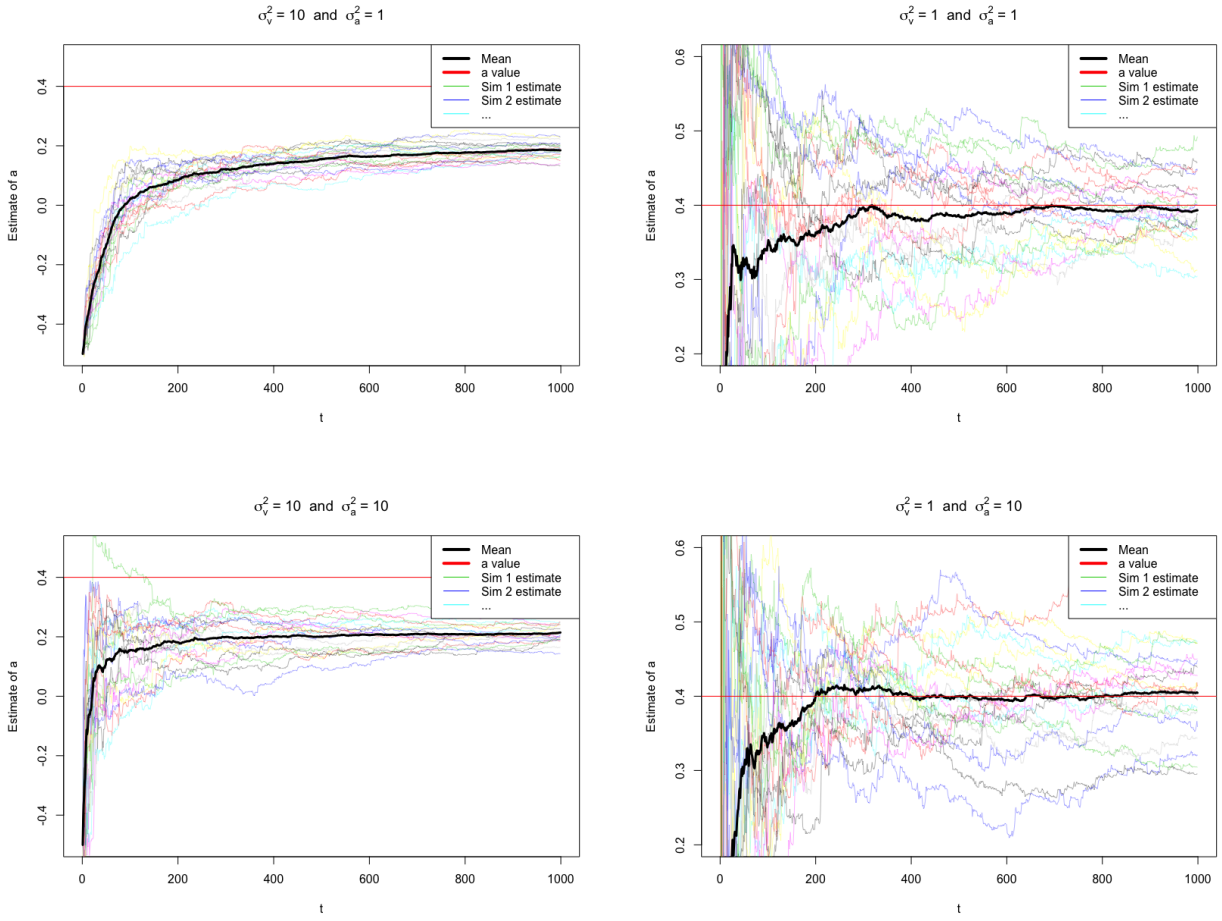

$\sigma_v^2 = 10$ and $\sigma_a^2 = 1$
Mean
a value
Sim 1 estimate
Sim 2 estimate
...
Estimate of a
t


$\sigma_v^2 = 1$ and $\sigma_a^2 = 1$
Mean
a value
Sim 1 estimate
Sim 2 estimate
...
Estimate of a
t


$\sigma_v^2 = 10$ and $\sigma_a^2 = 10$
Mean
a value
Sim 1 estimate
Sim 2 estimate
...
Estimate of a
t


$\sigma_v^2 = 1$ and $\sigma_a^2 = 10$
Mean
a value
Sim 1 estimate
Sim 2 estimate
...
Estimate of a
t

Initial a=-0.5



From these figure we can draw the following conclusions:

- The initial a value affects the parameter estimates in the sense that it take slightly longer to converge, but it does not affect the asymptotic behavior of a_t.
- σ_v^2 seems to control the value to which a_t converges. If σ_v^2 is large, a lot of the 'signal' is lost in a very high system variance. This is less likely to happen in a real setting, as the system variance is estimated by real measurements, and a good estimate of the variance should be available.
- σ_a^2 seems to affect the rate of convergence s.t. a higher σ_a^2 means faster convergence and vice versa. This makes sense, since it quantifies how much we 'trust' a_{t-1} for each time step, manifested by the Kalman gain.