Continuous Autoregressive Model via Kalman Filter

COMP777 Final Report

Jun. S. Han (42420601), Department of Mathematics and Statistics, Macquarie University

This report contains the overall summary of the project which has been replicated to recognise open science and reproducible research. It is submitted as a final report for COMP777 (Computing Methods for Research) course in November 2019.

1. Introduction

1.1. Brief Description

The autoregressive (AR) model has been commonly used in various data analysis to fit a time-series data and make appropriate forecasts based on certain assumptions. It has a statistical feature, that the output variable is regressed on its own previous values. The model is widely applied in many areas, including biology, econometrics, epidemiology, finance, and medical science. However, this model has a major drawback, as it assumes that each observation is recorded at an equal time gap. That is, the observation is only seen at a discrete time scale.

A more realistic approach is first proposed by Jones (1981), which considers fitting time-series data that is sampled at an irregular time gap using the continuous AR model. Belcher et al. (1994) modifies the work by reparametrising the continuous AR model which takes into account the time scale factor. In addition, the Kalman filter is implemented in the estimation of unknown scale factor and in the forecasts. The modified work is then written as an R code by Wang (2013) in the **cts** package, which includes coding for model calibration, model diagnostics, Kalman smoothing, and forecasts, along with identical examples provided in the original work. It is claimed that this approach is useful with unequally sampled time-series data, especially in the area of geophysics and medical science.

1.2. Evaluation of the Work

The significance of this project is that the methodology proposed by Belcher et al. (1994) has been coded for use in R, which is opened to anyone intending to use this methodology for fitting a continuous time-series data, observed at an irregular time space. As this paper focuses on introducing a new methodology, and how it is implemented in R, there is no such objective evaluation metric or framework provided that evaluates the methodology in comparison to others. However, within the methodology proposed in the original paper, some measures to compare models with different orders are used, namely *Akaike Information Criterion* and *Bayesian Information Criterion*. They are used to compare the goodness of fit among different types of statistical models, where the lower the value is, the better the model fit is. Belcher et al. (1994) claims that this approach provides a better fit in medical application, but its comparison with other methods is not shown.

1.3. Justification for the Choice of this Project

The aim of this project is to replicate Wang's project, using the code available in science. The article was submitted and published on *the Journal of Statistical Software* in 2013, where the Journal Impact Factor is recorded as 11.655 in 2018. This article has been cited 19 times since 2013, which, depending on people's views, may be considered as appropriate, given the subject area is quite narrow. The code used for the article is also shared with the submission of the paper.[1]

[1] Wang, Z., 2013, cts: An R Package for Continuous Time Autoregressive Models via Kalman Filter, *Journal of Statistical Software*, Vol. 53, No.5. The article and the code is available from the following website: https://www.jstatsoft.org/article/view/v053i05

Another reason for the choice of this project is that, as a research student from the department of mathematics and statistics, it is believed that the topic introduced in this article is a fundamental concept which needs to be understood for personal future research. A personal area of interest is undertaking stochastic analysis and modelling using financial commodity price data. As price data are considered as continuous due to some irregularities in observations coming from weekends or public holidays, this prompted us to examine whether any financial data, or simulated data with properties from the finance literature, can be applied to the original project.

1.4. Outline of the Report

The remainder of this report is organised as follows. Section 2 explains the basic concepts, the proposed methodology and what is included in the provided code. Section 3 shows the result of the application using the simulated dataset and a brief conclusion of the simulated study. Finally, Section 4 concludes with personal reflections on the replication procedure.

2. Replication of Original Work

2.1. Basic Concepts

In this section, a brief overview of the methodology is outlined, through explaining some relevant basic concepts.

2.1.1. The Continuous Autoregressive (CAR(p)) Model

Define x_t to be the observed variable at time t_k, for $k = 1, 2, ...n$. Then, we assume that this variable satisfies the equation $x_{t_k} = y(t_k) + \eta_k$, where $y(t_k)$ follows a p-th order CAR, such that:

$$Y^{(p)}(t) + \alpha_1 Y^{(p-1)}(t) + ... + \alpha_{p-1} Y^{(1)}(t) + \alpha_p Y(t) = \epsilon(t)$$

where $Y^{(i)}(t)$ is the i-th derivative of $Y(t)$, and $\epsilon(t)$ follows a Brownian process $B(t)$ with its variance measured as $\sigma^2 = \text{var}(B(t+1) - B(t))$.

This model is reparameterised in Belcher et al. (1994), which is expressed in terms of ϕ_i, where $\phi_i = -\frac{(1+\alpha_i/\kappa)}{(1-\alpha_i/\kappa)}$. This reparameterisation ensures that the model is numerically stable, and it accounts for the time scale factor κ in the model to deal with measurement irregularities. However, the estimated ϕ_i's do not have any interpretational meaning which could be a disadvantage compared to an ordinary AR(p) model.

2.1.2. The Kalman Filter

The Kalman filter is a statistical technique used in time-series analysis, for which the unobservable factor of the variable of interest is used in the model calibration, estimation, and forecast. The observed variable has a linear relationship with the unobservable factor, such that $x_k = G\psi(t_k) + \eta_k$, where $\psi(t_k)$ represents the unobservable factor at time t_k, and it is linked with the transition matrix G, and some error term η_k. In the forecasting process, the Kalman filter is implemented, and the recursion procedure is described in Figure 1.

In order to implement the Kalman filter in the forecast process, the residuals, or the error terms must be normally distributed with constant variance, and serially independent. This assumption can be checked visually by inspecting four diagnostic plots:

- The residual vs fitted plot: the standardised residuals are expected to be distributed evenly above and below zero if the residual has constant variance.
- The autocorrelation function (ACF) plot: any autocorrelation beyond, and including lag 1, must be below the significance level.
- The cumulative periodogram: if the line follows $y = 2x$, where x is frequency, then it is symptomatic for white noise (normally distributed process).
- P-values for Ljung-Box test statistic: as Ljung-Box test examines whether residuals are independently distributed or not, this plot is expected to show all p-values above 5% level of significance.

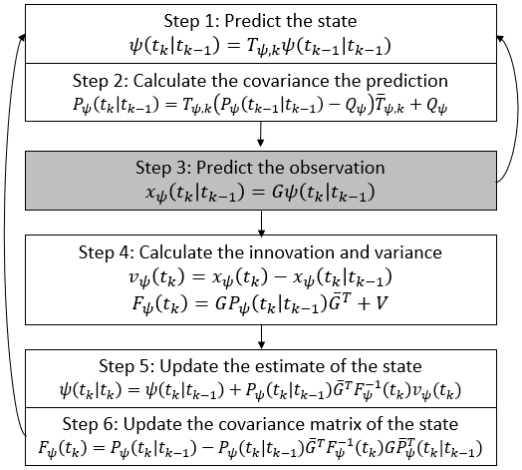

Figure 1: A flowchart of the Kalman filter process

2.1.3. Kalman Smoothing

The Kalman smoothing technique is used to estimate the unobservable components at each time point in the data. The smoothed plot provides visual information about the unobservable factors $\psi(t_k)$ as described in Section 2.1.2. A model with relatively low order can be used to provide interpretational meaning to each unobservable factor, which includes trend, diurnal or randomness component. One thing to highlight is that the model with order 1 cannot be decomposed into different components in **cts** package, as not enough information is provided by the model for the Kalman smoothing procedure.

2.2. The Original Dataset

Wang uses two datasets[2] to illustrate how the codes perform to fit the time-series data. Two datasets are described below, and those datasets are attached within **cts** package.

2.2.1. Geophysical Application

In the first example, the oxygen isotope data is used. A series of 164 measurements of relative abundance of an oxygen isotope in an ocean core is recorded, where each measurement is taken at regular depths at intervals of 10cm. However, due to variation in rates of sedimentation, it corresponds to unequally spaced time points, with an average separation of 2,000 years. The dataset is available in RData file, which contains the time of observation (in kiloyears) and the value at each time point. The dataset was provided to the original authors by Dr. N.J. Shackleton.

Table 1: First five observations from oxygen isotope data

Time	Measure
6.1290	0.92
8.3871	0.74
10.6450	0.59
12.9030	0.14
15.1610	-0.22

2.2.2. Medical Application

The second example illustrated in the original work uses medical data. 209 measurements of the lung function of an asthma patient are recorded, which are made by the patient mostly at 2 hour time interval, but with irregular gaps between each observation. This dataset is also available in RData file, which only consists of the time in hours and the observed measure. The dataset was originally provided to the authors by the Occupational Asthmatic Systems Group.

[2]To download the dataset, please refer to https://github.com/cran/cts/tree/master/data.

Table 2: First five observations from lung function data

Time	Measure
8	520
10	500
12	540
14	520
16	500

2.3. Replication of the Work

Replicating the original work using the code available in *the Journal of Statistical Software* is relatively easy, as each step has a description of what the code is doing. The program used to run the code for this project is the statistical software called R. As the datasets used in examples are attached to the **cts** package, the procedure is fairly straightforward. The code file provided includes all concepts explained in Section 2.1, and if interested in comparing the result to the original work, please refer to this github repository.[3]

Two datasets can easily be loaded from the package, and plots of the data, model calibration, Kalman smoothing and forecasts are replicated quite easily. All plots and numerical values, including the estimated coefficients, values of unobservable factors and predicted values can be obtained as shown in the paper. There is no problem with getting identical results from the paper, as the datasets are not randomly simulated, and the mathematics behind those codes provide the same result. However, potential differences are expected when the data is randomly simulated using any stochastic process.

3. Simulation Study

3.1. Simulated Dataset

A new dataset is simulated using a commonly used stochastic process in the finance literature, known as *Ornstein-Uhlenbeck Process*, or *Cox-Ingersoll-Ross Model*. Main reasons for the choice of this process are the followings:

- A mean-reverting property: Observed values will tend to move back to the average over time. This property is important, as in the finance literature, a lot of analysts assume that a stock's price will move back to its expected price over time.
- Positive values: This process guarantees that the observed values will be positive, which is consistent with typical financial data.

Consider the following stochastic process:

$$dX_t = \lambda(\mu - X_t)dt + \sigma\sqrt{X_t}dW_t$$

where

$$\begin{aligned} X_t &= \text{The value of the variable at time } t \\ \lambda &= \text{The speed of the mean-reversion} \\ \mu &= \text{The mean of } X_t \\ \sigma &= \text{The volatility of } X_t \\ dt &= \text{Time difference between two close observations} \\ dW_t &= \text{The standard Brownian Motion} \end{aligned}$$

For the purpose of simulation, the discretised version of this process is used, which is defined as:

$$X_{t+\Delta t} = X_t + \lambda(\mu - X_t)\Delta t + \sigma\sqrt{X_t}\varepsilon\sqrt{\Delta t}$$

where

$$\Delta t = \text{Time difference between observations, } \varepsilon \sim N(0,1).$$

[3]The codes for replicating original data, simulating new data and its application are included in the github. Available at: https://github.com/Junee1992/COMP777-Final-Report.

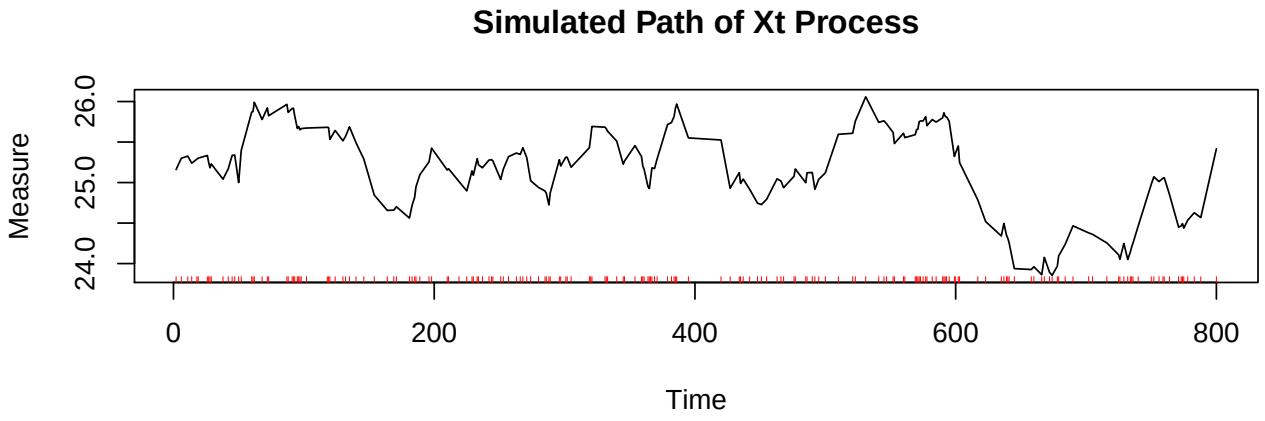

Figure 2: Plot of simulated data

The following specification is used for simulating the dataset.

$$X_0 = 25, T = 1, n = 900, \Delta t = \frac{1}{900}, \mu = 25, \sigma = 3$$

where

T = The whole time period considered, n = The number of observations to be simulated

To remove its initial effect, the first 100 observations are discarded as "burn-in", and to adjust for measurement irregularity, 600 random observations are removed, only leaving with 200 observations recorded at an irregular time space. Hence, within a time space of 800, only 200 observations are recorded at a random time. The plot of the simulated dataset is illustrated in Figure 2, and the first five observations of this dataset is briefly shown in Table 3.

Table 3: First five observations of the simulated data

Time	Measure
2	25.16045
6	25.30007
11	25.32625
14	25.23889
18	25.28914

3.2. Results

We have first fitted the simulated data up to order of 13, and compared each model using AIC and BIC, as explained in Section 1.2. We only show the result up to order of 5.

Table 4: Order selection statistics using AIC and BIC

order	AIC	BIC
1	286.1305	292.7271
2	286.0767	295.9716
3	287.2117	300.4050
4	289.1992	305.6908
5	290.6467	310.4366

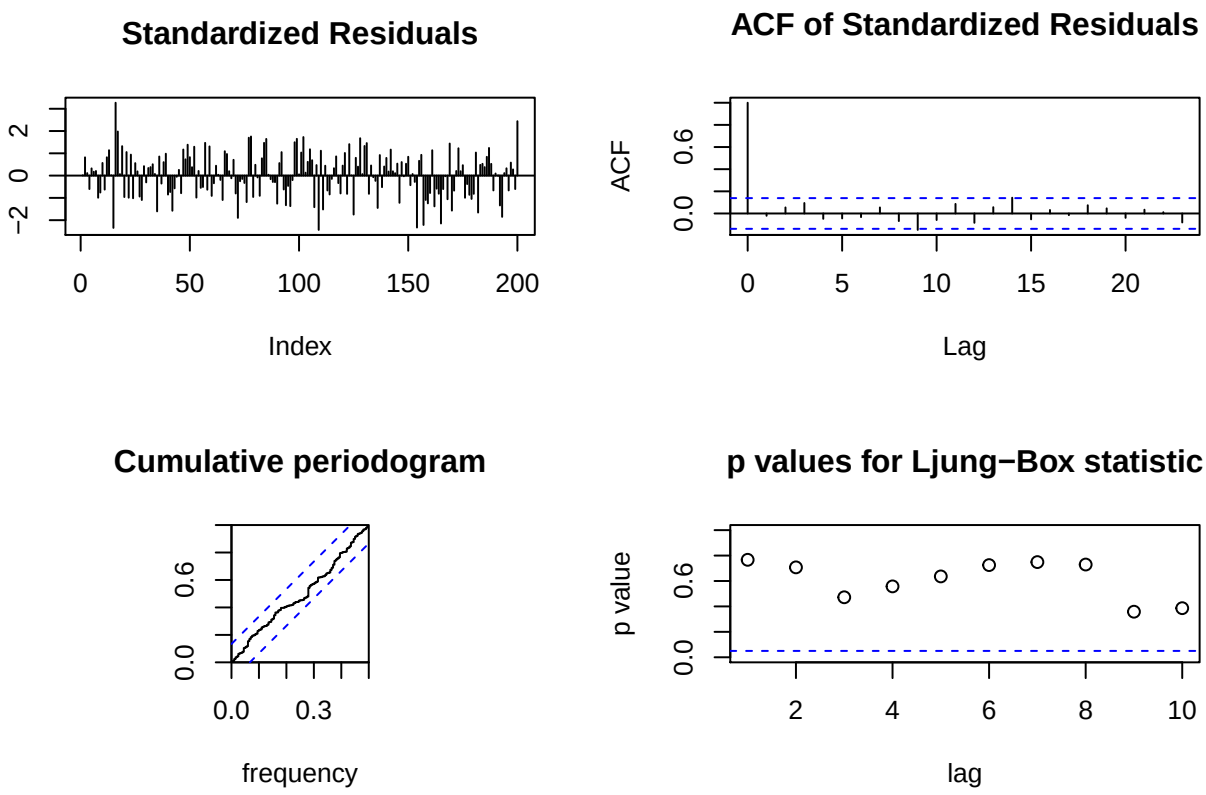

Figure 3: Model diagnostics for simulated data

The ideal model for this dataset is the model with order 1, as it has both AIC and BIC at the lower end. However, as explained in Section 2.1.3., a model with order 1 cannot be used to show the Kalman smoothing. To be consistent with examples provided by Wang (2013), the model with order 4 is used, and the summary of the model is provided below. In addition, as explained in Section 2.1.1, these estimated coefficients do not have any interpretational meaning. Subsequently, to implement the Kalman filter, the assumption on the residual must be checked, as seen in Figure 3. As all the plots in the model diagnostic are satisfactory, the Kalman filter and the Kalman smoothing can be applied.

```
Call:
car(x = data1, scale = 0.25, order = 4)

Order of model = 4, sigma^2 = 1.87e-06

Estimated coefficients (standard errors):
      phi_1 phi_2 phi_3  phi_4
coef -0.996 0.054 0.089 -0.011
S.E.  0.089 0.116 0.128  0.101

Estimated mean (standard error):
[1] 25.152
[1] 0.169
```

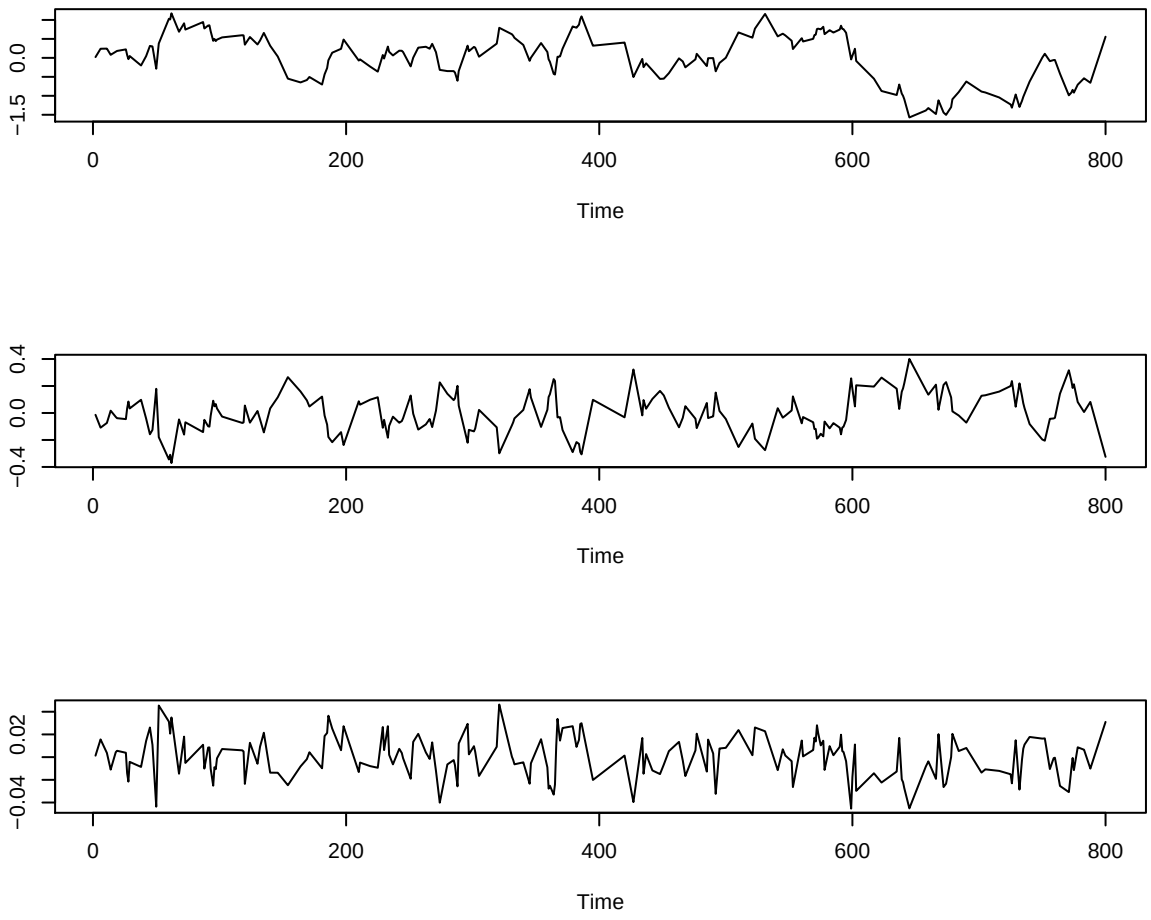

Figure 4: Components of the simulated data. From top to bottom: trend, diurnal and randomness component.

Now, the estimated unobservable components can be summarised as a plot, as shown in Figure 4. Following the interpretation from the original work, the top plot represents the trend component, the middle represents the diurnal component, and the bottom represents the randomness component. All three components are represented in terms of vector ψ_{t_k} which is linearly related to the observed values x_{t_k}. These components can then be used in the forecast procedure, and Figure 5 shows 10 points ahead of the point that is last observed.

```
Call:
car(x = data1, scale = 0.25, order = 4)

             1       2       3       4       5       6       7      8
Time    801.000 802.000 803.000 804.000 805.000 806.000 807.000 808.00
Predict  25.427  25.438  25.448  25.458  25.465  25.472  25.477  25.48
             9      10
Time    809.000 810.000
Predict  25.482  25.483
```

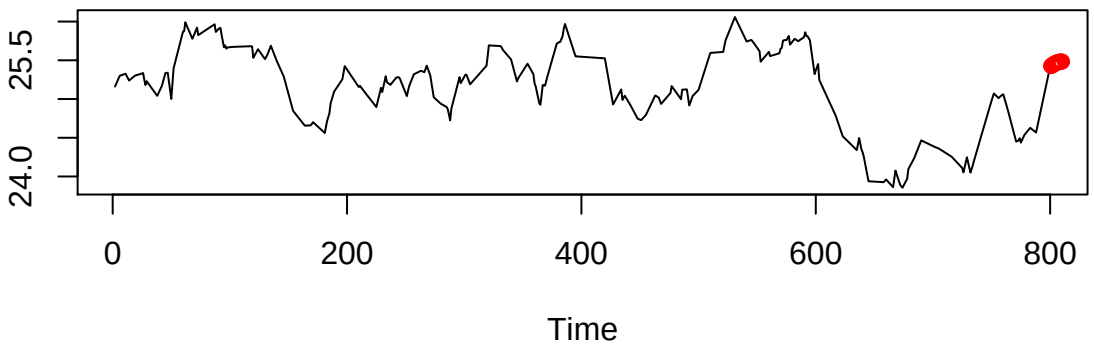

Figure 5. Forecasts for simulated data (in red).

3.3. Conclusion of the Simulated Study

The whole process of applying the simulated dataset is consistent with examples shown in the original work. Modelling the time-series data in a continuous time scale, sampled at an irregular time gap has been shown to be successful, along with obtaining information about the unobservable factors which are related to the variable of interest. Lastly, the forecast seems to be appropriate, as it combines the predictions from the three different unobservable components and produces forecasts. Hence, it is believed that the result is satisfactory, and may be applicable to any real time-series data that is irregularly sampled.

4. Reflection

As this project is based on a proposed methodology in time-series analysis, the data simulation is relatively easy, as there exists a number of different types of stochastic processes in statistics literature which are widely used in practice for analytic purposes. However, it is of a personal thought that constructing a time-series data with seasonal effect could have shown a more interesting result, especially if the data is decomposed into three different unobservable factors.

Secondly, application to real data may have provided a more meaningful conclusion to the project. The initial idea for a new dataset to be applied in the model was to apply the historical price data of the European Emission Allowance (EUA) futures contract, which can easily be obtained from the Bloomberg terminal. However, it was suggested that a simulated dataset may be sufficient to test whether the code performs similarly to the original work.

Thirdly, other than showing the statistical validity of the model, comparison with other time-series data may have convinced the readers about the performance of this model. As AIC and BIC may not be used for model comparison here due to different assumptions made in the continuous AR model and in the discrete AR model, some other measures may be required, which is beyond the scope of personal knowledge.

Lastly, some changes were made to the result of the application to new dataset, as the result shown in the final presentation had mistake in the forecast section.

Reference

1. Jones, R.H., 1981, Fitting a Continuous Time Autoregression to Discrete Data, *Applied Time Series Analysis II*, pp.651-682.
2. Belcher, J., Hampton, J.S., Tunnicliffe, W.G., 1994, Parametrization of Continuous Time Autoregressive Models for Irregularly Sampled Time Series Data, *Journal of the Royal Statistical Society B*, Vol.56, No.1, pp.141-155.
3. Wang, Z., 2013, cts: Continuous Time AR Models via Kalman Filter in R, *Journal of Statistical Software*, Vol.53, No.5.