Forecasting interest rate with CIR model

Francesca Paccagnella   Alessandro Torelli   Francesco Zorzi

Problem formulation

In this work we wanted to estimate the interest rate for H days forecasting using the Cox-Ingersoll-Ross (CIR) model. The CIR is a square root diffusion model:

$$dx_t = \alpha(\mu - x_t)\Delta t + \sigma\sqrt{x_t}dW_t \tag{1}$$

We used two different discretization scheme:

1. The Euler scheme

$$x_{t+\Delta t} = x_t + \alpha(\mu - x_t)\Delta t + \sigma\sqrt{x_t}\sqrt{\Delta t}Z_t \tag{2}$$

2. The Milstein scheme

$$x_{t+\Delta t} = \frac{x_t + \alpha\mu\Delta t + \sigma\sqrt{x_t}\sqrt{\Delta t}Z_t}{1 + \alpha\Delta t} \tag{3}$$

For parameters estimation we used the ordinary least squares estimation procedure. After finding parameter estimations, we simulated a number of Monte Carlo simulations using the CIR model. We used this procedure for predicting the future rates by historical ones. With these results it is possible to price some contracts as Vanillas or Asian.

So we had as input:

1. An array with the historical rates of LIBOR
2. A date as start point to begin the prediction

The start point and the dimension of the historical rates arrays are chosen by the user. Dimension means the number of backward observations you want to include in parameter estimations.

Our output will be an array of double values, which are our predict rates. To make easier the interpretation of results, we print the predicts in a graph comparing them with the real ones.

Data representation

As we told above, we have the rates and the date as input. We loaded all the data (choosing only dates and twelve-month rates) in a .CSV file from the web. We fed it to the algorithm. Then we received an array of double estimated rates as output. Subsequently we plotted the estimated curve compared to the real curve in a graph.

When we loaded the data, we used the Split method to menage the spaces between the values. While for dates we used the class *DateTime*, which had the same format of the web data.

One of problems we met is about dates not found. This happened because the market is not always open, so we had to manage the choosing days. Furthermore, even if there are dates, sometimes interest rate values are missing. We managed this problem using the CIR model to estimated the missing value. Another problem was the management of the rolling window, we imposed the *start position* could not be a date included in the previous n-days of the file (where n is the number of days you want to use for estimating data). The reason why we imposed this was because we wanted to compare our estimated and historical rates n days after. All these problems were managed with appropriate exception. For example, when we checked if the date existed, if the input date was not found, a message on the display said that we searched a date where the market was closed. For this reason we cannot have the correspondent rate value.

Statistical model

In our analysis we used the CIR model. This model is widely used for modeling interest rates. The following equation represents the dynamic representation of the CIR:

$$dx_t = \alpha(\mu - x_t)\Delta t + \sigma\sqrt{x_t}dW_t \tag{4}$$

where x_t is the interest rate and $\theta = (\alpha, \mu, \sigma)$ are the models parameters. One of Cir model properties is the mean reverting, i. e. interest rate x_t moves in the direction of its mean μ at speed α. The drift function is known as $\mu(x_t, \theta) = \alpha(\mu - x_t)$ and is linear. The diffusion function $\sigma^2(x_t, \theta) = x_t\sigma^2$ is proportional to the interest rate x_t.

We choose to estimate the parameters with the maximum likelihood estimate. We took the conditional density function of x_{i+1} and we obtained the log-likelihood function. The log-likelihood function needs to be maximized by taking partial derivatives with respect to μ, α, and σ, putting them equal to zero.

First, the simulation of the dynamics above is illustrated as:

$$x_{t_{i+1}} - x_{t_i} = \alpha(\mu - x_{t_i})\Delta t + \sigma\sqrt{x_{t_i}}\epsilon_{t_i} \tag{5}$$

where $\epsilon_{t_i} \sim N(0, 1)$ and also as:

$$x_{t_{i+1}} = \alpha\mu\Delta t + (1 - \alpha\Delta t)x_{t_i} + \sigma\sqrt{x_{t_i}\Delta t}\epsilon_{t_i} \tag{6}$$

In order to use the OLS, the first equation must be transform to:

$$\frac{x_{t_{i+1}} - x_{t_i}}{\sqrt{x_{t_i}}} = \frac{\alpha\mu\Delta t}{\sqrt{x_{t_i}}} - \alpha\sqrt{x_{t_i}}\Delta t + \sigma\epsilon_{t_i} \tag{7}$$

The sum square of the error $\sum_{i=1}^{n-1}(\sigma\epsilon_{t_i})^2$ must be minimized in terms of α and μ to obtain $\hat{\alpha}$ and $\hat{\mu}$ such that:

$$(\hat{\alpha}, \hat{\mu}) = arg\min_{\alpha,\mu}\sum_{i=1}^{n-1}(\sigma\epsilon_{t_i})^2 = arg\min_{\alpha,\mu}\sum_{i=1}^{n-1}\left[\frac{x_{t_{i+1}} - x_{t_i}}{\sqrt{x_{t_i}}} - \frac{\alpha\mu\Delta t}{\sqrt{x_{t_i}}} - \alpha\sqrt{x_{t_i}}\Delta t\right]^2 \tag{8}$$

then the estimated parameters will be:

$$\hat{\alpha} = \frac{n^2 - 2n + 1 + \sum_{i=1}^{n-1} x_{t_{i+1}}\sum_{i=1}^{n-1}\frac{1}{x_{t_i}} - \sum_{i=1}^{n-1} x_{t_i}\sum_{i=1}^{n-1}\frac{1}{x_{t_i}} - (n-1)\frac{x_{t_{i+1}}}{x_{t_i}}}{\left(n^2 - 2n + 1 - \sum_{i=1}^{n-1} x_{t_i}\sum_{i=1}^{n-1}\frac{1}{x_{t_i}}\right)\Delta t} \tag{9}$$

$$\hat{\mu} = \frac{(n-1)\sum_{i=1}^{n-1} x_{t_{i+1}} - \sum_{i=1}^{n-1} \frac{x_{t_{i+1}}}{x_{t_i}} \sum_{i=1}^{n-1} x_{t_i}}{\left(n^2 - 2n + 1 + \sum_{i=1}^{n-1} x_{t_{i+1}} \sum_{i=1}^{n-1} \frac{1}{x_{t_i}} - \sum_{i=1}^{n-1} x_{t_i} \sum_{i=1}^{n-1} \frac{1}{x_{t_i}} - (n-1)\frac{x_{t_{i+1}}}{x_{t_i}}\right)} \tag{10}$$

$$\hat{\sigma} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n-1}\left(\frac{x_{t_{i+1}} - x_{t_i}}{\sqrt{x_{t_i}}} - \frac{\hat{\mu}}{\sqrt{x_{t_i}}} + \hat{\alpha}\sqrt{x_{t_i}}\right)^2} \tag{11}$$

The reason why we choose to estimate LIBOR rates was that this model cannot estimate negative interest rates as short EURIBOR ones. We used two discretizations:

1. The Euler scheme

$$x_{t+\Delta t} = x_t + \alpha(\mu - x_t)\Delta t + \sigma\sqrt{x_t}\sqrt{\Delta t}Z_t \tag{12}$$

2. The Milstein scheme

$$x_{t+\Delta t} = \frac{x_t + \alpha\mu\Delta t + \sigma\sqrt{x_t}\sqrt{\Delta t}Z_t}{1 + \alpha\Delta t} \tag{13}$$

These are two possible discretizations for the CIR model using as parameters: α, μ, σ. It exists more than one possible discretization but parameters should be estimated in another way.

Algorithmic approach

We implemented the algorithm by ourselves using also libraries:

```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Windows.Forms.DataVisualization.Charting;
using System.Drawing;
using PlotCharts;
```

The main algorithms are about:

- *Data load*: count how many data there are in a specific .CSV file and load the data using the *split* method, if the value associated with a date does not exists, this method estimates a value using the CIR model
- *MonteCarlo simulation*: we used the previous formula for the dynamics of the CIR model; applying the least square method we found all the parameters we needed; we discretized the time Δt in a market year (255 days) taking count also of the bissextile year ($\frac{1}{255*0.75+256*0.25}$ because the bissextile year occurs every 4 years); finally we used the *NormalDistribution* class to create an object with zero mean, unitary variance and random probability from zero to one

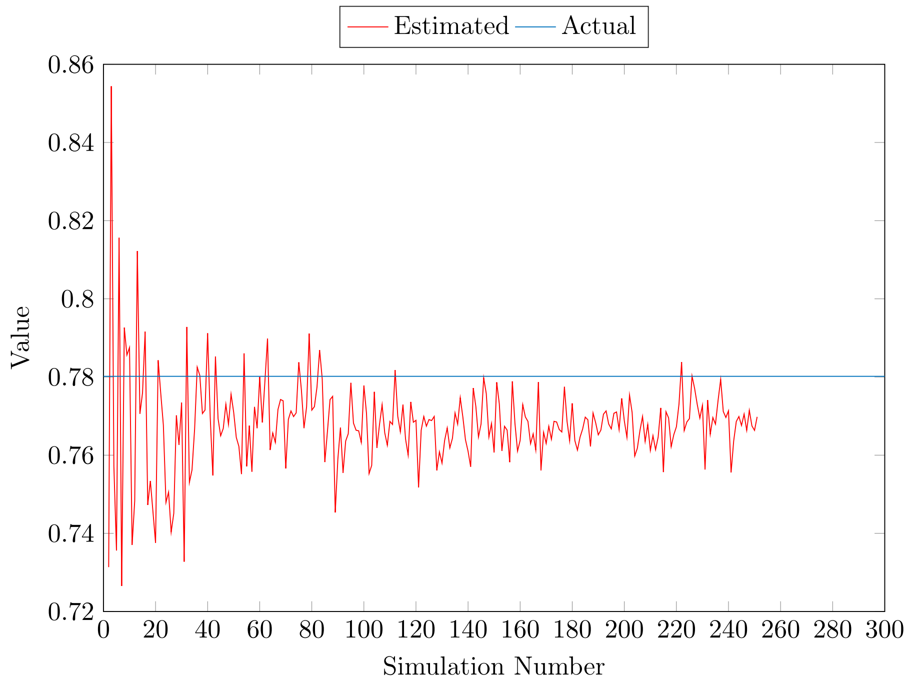

Figure 1: Convergence Euler scheme

- *Date finder*: find the date that we want to begin the simulation with, using the binary search and the *DateTime* method to compare two dates
- *Backtesting*: recursive algorithm for *MonteCarlo* method takes count of the previous estimated values doing a rolling window and the estimated ahead in the time
- *Plot*: we create a class that plot all results in *form* using the characteristics we want, like the color of the curves, the legend, labels of axisis and so on

Evaluation and results

We can see that the estimate curve follows the real curve and so the target is centered! We have some *bad* results because the spread between the curve of *Actual* and *Estimated* values is bigger than others. This could happen because our estimation of the variance is not the best, so it could produce bad estimated values. Furthermore the Milstein scheme does not work as well as Euler one, maybe for the estimated parameter method chosen. We show all results in *form* that plot our estimated rates compared to the real ones. Then in another graphic it is possible to see how fast the *Monte Carlo* method goes to the stability.

Figure 4 shows that increasing the number of observations the result is more stable.

Instead, Figure 5 shows that increasing the number of days ahead the model will become less accurate.

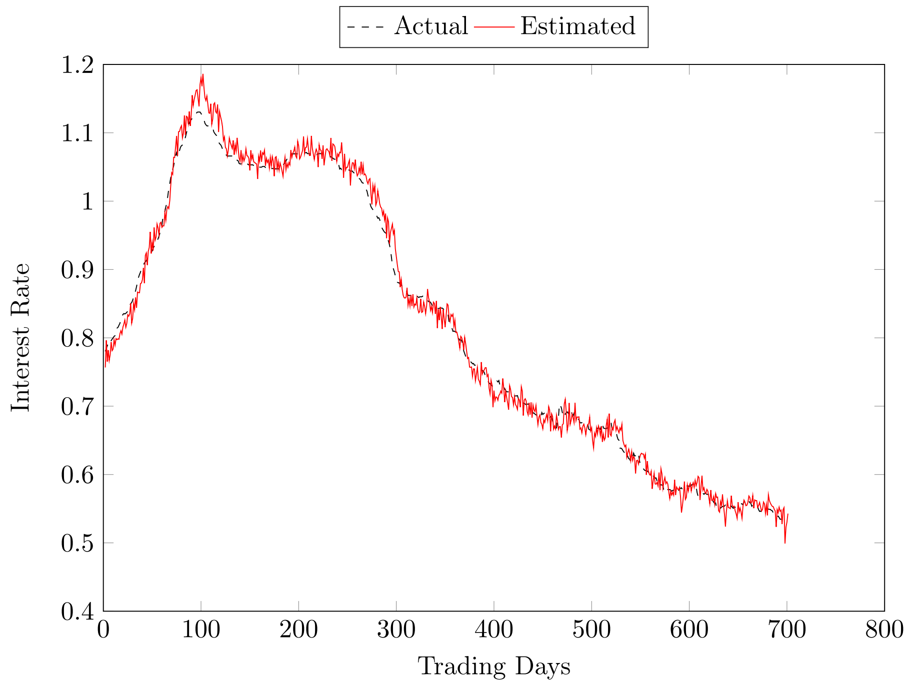

Figure 2: 5-days a-ahead backtest for the 12-month LIBOR interest rate series with 50 simulations Euler scheme

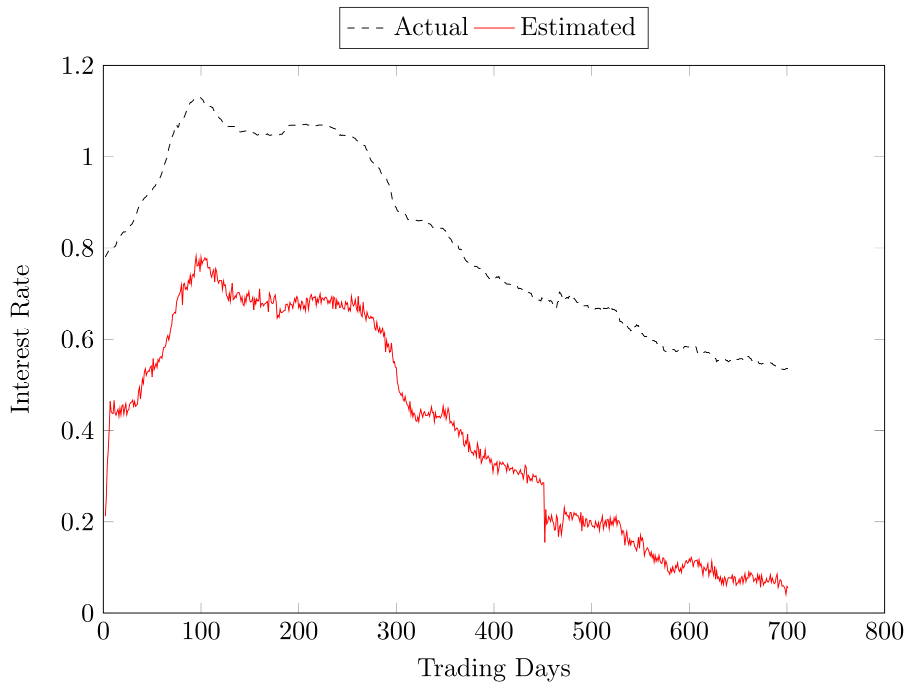

Figure 3: 5-days a-ahead backtest for the 12-month LIBOR interest rate series with 50 simulations Milstein scheme

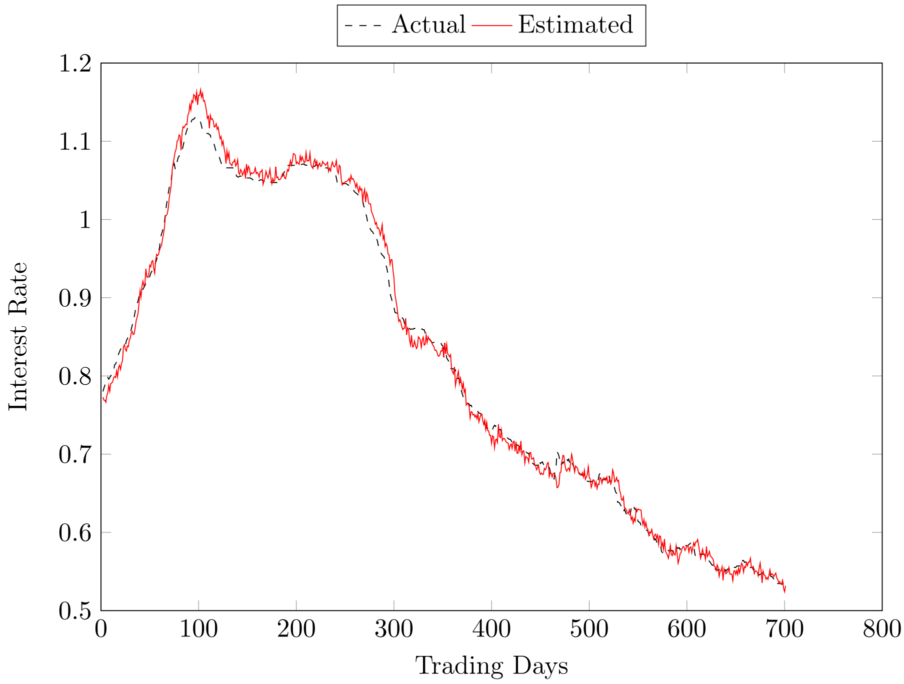

Figure 4: 5-days a-ahead backtest for the 12-month LIBOR interest rate series with 250 simulations Euler scheme

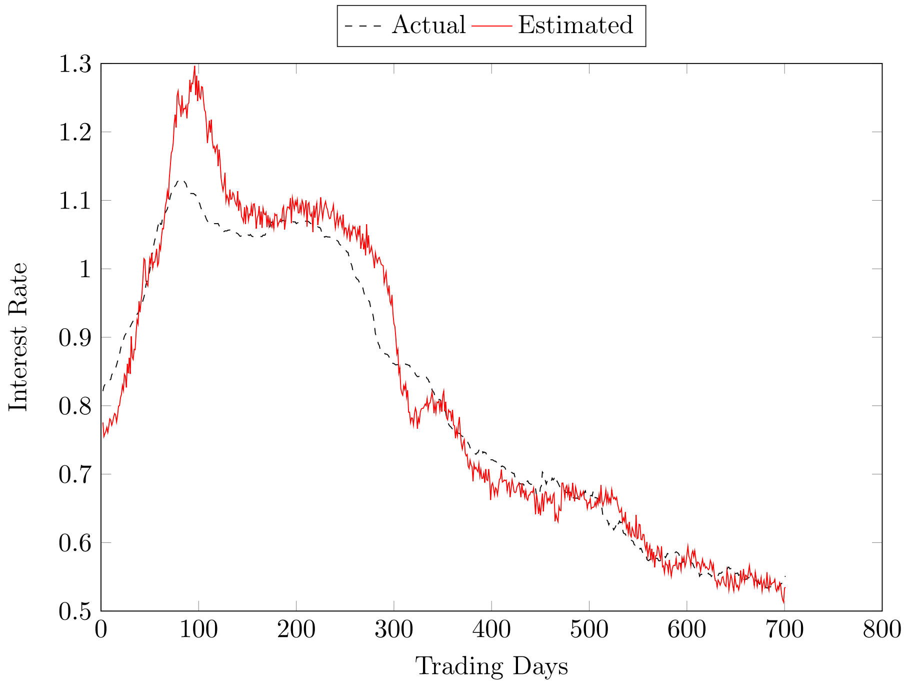

Figure 5: 20-days a-ahead backtest for the 12-month LIBOR interest rate series with 250 simulations Euler scheme

Conclusion

We can see that the algorithm, which estimates the coefficients, is precise but we cannot obviously have two identical graphics because there is always an error (even if minimum):

$$\dot{f} = f - \epsilon \tag{14}$$

where ϵ is the error, f is the function, $\dot{f}$ is the estimated function.

We could have more precise estimated data if we took a longer period of historical rates. Furthermore, if we had more time we could solve problems like the Visual Studio-Excel communication so it could print the results and have a more efficient algorithm. For plotting our results we used a graphic library. We created two class: *ResulPlots* and *ResultPlot.Designer*.

We managed the problem regarding the outlier data smoothing the divergent result using an *if* cycle. This cycle does an average between the two previous predictions if the estimated value exceeds 50% out of the previous one. In the same way we bound the variance to not exceed 20% out of the previous one.

In conclusion we can say that this project was difficult both in mathematical and computational fields. The latter because we have learn by ourselves how ti print the curve and how to implement the algorithm to predict the rates. With project it is possible to do some applications, like the pricing of Vanillas and Asian (more difficult), that could be the next project.

Bibliography

[1] http://www.broda.co.uk/gsa/CIR_model_S_Plus.pdf

[2] https://financetrainingcourse.com/education/2012/06/
cox-ingersoll-ross-cir-interest-rate-model-parameter-calibration-short-rates-simulat

[3] https://msdn.microsoft.com

[4] http://www.thetaris.com/wiki/CIR_model

[5] http://uir.unisa.ac.za/bitstream/handle/10500/4049/
dissertation_sypkens_r.pdf

[6] S.Zeytun, A.Gupta, *A comparative study of the Vasicek and CIR model of the short rate*

[7] Fevrier, *Estimation of the Historical mean reverting parameter in the CIR model* 2007