Calibration and Simulation of Interest Rate Models

Kevin Shea, CFA
kevin.shea@mathworks.com
Agenda

- Calibration to Market Data
- Calibration to Historical Data
- Simulation and Valuation
- Counterparty Credit Risk Analysis
- Questions and Answers
Interest Rate Models

Cox-Ingersoll-Ross

\[ dr(t) = a(b - r)dt + \sigma \sqrt{r}dW(t) \]

Hull-White

\[ dr(t) = (\theta(t) - ar)dt + \sigma dW(t) \]

G2++

\[ r(t) = x(t) + y(t) + \varphi(t) \]
\[ dx(t) = -ax(t)dt + \sigma dW_1(t) \]
\[ dy(t) = -by(t)dt + \eta dW_2(t) \]
\[ dW_1(t)dW_2(t) = \rho dt \]
Calibrate to Market Data

- Choose a set of liquid calibration instruments – typically caps, floors, swaptions.
- Find the set of model parameters that matches as closely as possible the observed prices.

\[
\sum_{k=0}^{n} (P_i - \hat{P}_i(\theta))^2
\]

\(P_i\): Market Price
\(\hat{P}_i\): Model Price
\(\theta\): Model Parameters
Calibrate CIR Model using MLE of Transition Density

\[ dr(t) = a(b - r)dt + \sigma \sqrt{r}dW(t) \]

- \( a \): mean reversion speed
- \( \sigma \): volatility of the short rate
- \( b \): level

Stochastic Differential Equation Models

- Suite of models including: \( bm, gbm, cir, hwv, \\) heston, cev
- Simulate methods
- Framework for creating custom models
Calibrate using Kalman Filter

- Formulate models as state space systems.
- Use Kalman filter to estimate parameters.
- Estimate parameters from historical yield curves.

State Space formulation for G2++ Model

Transition Equation

\[ x_t = A x_{t-1} + B \mu \]

Measurement Equation

\[ y_t = C x_t + D \epsilon \]

\[ A = \begin{bmatrix} e^{-a\Delta t} & 0 \\ 0 & e^{-b\Delta t} \end{bmatrix} \]

\[ B = \begin{bmatrix} \sigma \sqrt{\frac{1 - e^{-2a\Delta t}}{2a}} & 0 \\ 0 & \eta \sqrt{\frac{1 - e^{-2b\Delta t}}{2b}} \end{bmatrix} \]

State Space Model

New state space model, ssm in Econometrics Toolbox™.

- Supports time-invariant and time-varying, linear state-space models.
- Perform univariate and multivariate time-series data analysis.
- Functionality to: estimate, filter, smooth, simulate, forecast
Interest Rate Model Simulation

Specify models and simulate entire term structure

- Support for Hull-White, G2++ and LIBOR Market Model.
- `simTermStructs` simulates entire term structure.
## Swap Portfolio

- Store data in a MATLAB Table.
- Easy to read in data.
- Tabular display.

```matlab
>> SwapPort = readtable('SwapPortfolio.xlsx')

SwapPort =

<table>
<thead>
<tr>
<th>Notional</th>
<th>Maturity</th>
<th>RecType</th>
<th>PayType</th>
<th>RecRate</th>
<th>PayRate</th>
<th>RecReset</th>
<th>PayReset</th>
</tr>
</thead>
<tbody>
<tr>
<td>1e+07</td>
<td>'1/15/2018'</td>
<td>1</td>
<td>0</td>
<td>0.031</td>
<td>10</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>5e+06</td>
<td>'2/15/2018'</td>
<td>0</td>
<td>1</td>
<td>20</td>
<td>0.032</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>1e+06</td>
<td>'3/15/2019'</td>
<td>1</td>
<td>0</td>
<td>0.033</td>
<td>30</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>2e+06</td>
<td>'4/15/2019'</td>
<td>0</td>
<td>1</td>
<td>40</td>
<td>0.034</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>1e+07</td>
<td>'5/15/2020'</td>
<td>1</td>
<td>0</td>
<td>0.036</td>
<td>50</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>7e+06</td>
<td>'6/15/2020'</td>
<td>0</td>
<td>1</td>
<td>65</td>
<td>0.036</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>7.5e+06</td>
<td>'7/15/2021'</td>
<td>1</td>
<td>0</td>
<td>0.0385</td>
<td>70</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>8e+06</td>
<td>'8/15/2021'</td>
<td>0</td>
<td>1</td>
<td>75</td>
<td>0.04</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>3e+06</td>
<td>'9/15/2022'</td>
<td>1</td>
<td>0</td>
<td>0.039</td>
<td>85</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>3.5e+06</td>
<td>'10/15/2022'</td>
<td>0</td>
<td>1</td>
<td>95</td>
<td>0.04</td>
<td>12</td>
<td>12</td>
</tr>
</tbody>
</table>
```
Valuing the Portfolio

- Value portfolio using `swapbyzero`
- Use `parfor` to loop over simulation dates.

```matlab
>> Values = zeros(nDates,nSwaps,nTrials);
>> parfor dateidx=1:nDates
    Values(dateidx,:, :) = swapbyzero(...)
end
```
Counterparty Credit Risk Functions

Compute exposures and CCR profiles

- Support for computing credit exposures.

- Support for computing various credit exposure profiles, including potential future exposure and expected exposure.

```
>> Exposures = creditexposures(Values);
>> Profiles = exposureprofiles(SimDates,Exposures);
```
Computing Credit Valuation Adjustment

- Compute exposure from exposure profiles
- Compute default probabilities from cdsbootstrap

\[
CVA = (1 - R) \int_0^T DiscExp(t) dPD(t)
\]
Summary

- Calibration Approaches
  - Market Data: lsqnonlin, simulannealbnd
  - Historical Data: mle, ssm
- Monte Carlo Simulation in MATLAB
  - cir
  - HullWhite1F, LinearGaussian2F
- Counterparty Credit Risk
  - creditexposures, exposureprofiles
  - cdsbootstrap