CQG and MATLAB®

Unlimited Analysis.

September 2010

CQG is a Third Party Solution Provider for MATLAB. Traders and analysts in the financial community using MATLAB toolboxes can now use CQG as their data source.

The results from the MATLAB functions can be called back into CQG. The MATLAB output can be viewed in quote displays and charts, and can be a part of trade routing. CQG version 8.2 allows customers to send data to a MATLAB function and then call the results of the function using a CQG custom study. No additional API or other special coding is required. We will walk through two examples.

Our first example is a simple moving average calculation performed by MATLAB using CQG market data with the output of the MATLAB function displayed on the chart. This is the CQG custom study function to calculate a moving average in MATLAB:

matlab(@,20,mean, Close(@))

Where:

- matlab(@) is required for all CQG studies using MATLAB functions.
- 20 is the look back period used by MATLAB.
- mean is the moving average function in MATLAB.
- Close(@) is the closing price of the symbol used for the moving average calculation.

Figure 1 is an example of the MATLAB function for the 20 period mean applied to the chart of the E-mini S&P.

Figure 1 shows the 20-bar moving average calculated in MATLAB using data from CQG and then displayed on the chart. This study value can be tracked in the price ladder of the DOMTrader® and orders can be set to automatically update as the study values change. You can, therefore, create proprietary studies in MATLAB, use CQG data and have orders working in the market that automatically update as MATLAB calculations change with market movement. A more complex example follows.

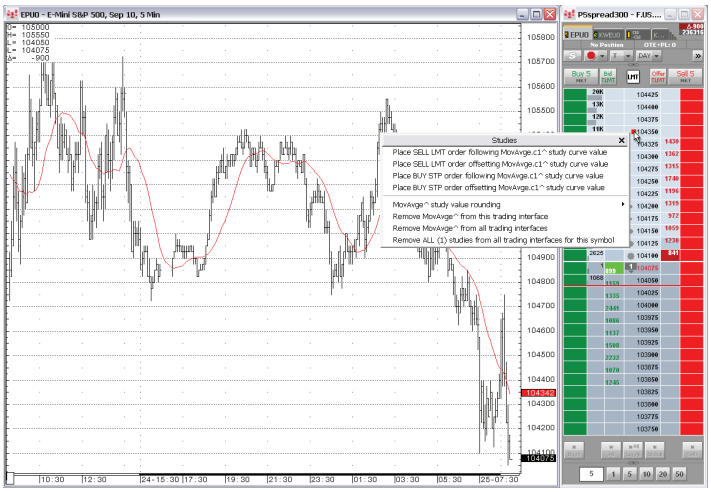


Figure 1: The 20-bar moving average displayed on the chart (red line) is calculated in MATLAB using data from CQG. The calculation results is then displayed on the chart. CQG can also display a study value on DOMTrader's price ladder and orders can be set to automatically follow the study's value.

MATLAB Implied Repo Rate Calculations

The US Treasury futures contract allows for a range of maturities that qualify for delivery by short sellers. Traders want to know which cash Treasury instrument is the cheapest to deliver as it is the security of choice. They use the purchase price of the cash Treasury and the holding period for delivery against the contract, and other factors to calculate the implied repo rate. The cash Treasury with the lowest implied repo rate is the cheapest to deliver.

By Thom Hartle

CQG

MATLAB has a function for calculating the implied repo rate:

ImpRepo = bndfutimprepo(Price, FutPrice, FutSettle, Delivery, ConvFactor, CouponRate, Maturity).

Where:

- Price: Cash treasury price
- FutPrice: The futures contract price
- FutSettle: The next business day used to calculate the settlement date for the purchase of the cash Treasury
- Delivery: Date of the last business day of the delivery month
- ConvFactor: A constant that is used to adjust the futures contract price to reflect the same coupon rate and maturity of the cash Treasury for calculating the invoice price.
- CouponRate: The coupon rate of the cash Treasury
- Maturity: The maturity date of the cash Treasury.

CQG-MATLAB Code Example

Here is a MATLAB example to compute the implied repo rate for a cash Treasury with a coupon of 8.75% and a maturity date of 8/15/2020 , a futures price of 98, settlement is September 21, delivery is December 29 and the conversion factor is 1.3136:

bndfutimprepo (129,98,'9/21/2000','12/29/2000',1.3136,.0875,' 8/15/2020')
Returns: 0.0584
This is a 5.84% return for the holding period.

CQG-MATLAB Custom Study

We can create a custom study in CQG to calculate the implied repo rate for a cash Treasury in MATLAB and then display the value in CQG. There are required parameters in the MATLAB formula that are unavailable from CQG, but the MATLAB implied repo rate formula can be modified to determine these parameters and then perform the calculation. The parameters unavailable from CQG are the next business day (FutSettle), date of the last business day of the delivery month (Delivery), and the conversion factor (ConvFactor).

Two of the three parameters are dates and MATLAB has functions to calculate these specific dates. MATLAB can also calculate conversion factors as a separate function. The MATLAB conversion factor formula needs to be modified.

First, the steps for calculating the conversion factor are presented. MATLAB defaults to using the 30 year Treasury bond and the 10-year Treasury note futures contracts for the conversion factors. The MATLAB formula for calculating a conversion factor is:

s = convfactor(refDateStr, matDateStr, couRate)

CQG can send over the cash Treasury maturity date and coupon, but the "refDateStr", which is the delivery month of the contract, has to be added to MATLAB's formula. Here is the example of the modified MATLAB formula for calculating the conversion factor for delivery against September futures contracts:

```
function [ s ] = myConvFactor( price, tMaturity, couRate)
refDate = fbusdate(2010, 9)
refDateStr = datestr(refDate)
matDate = addtodate(datenum('01-Jan-1900 00:00:00'),
tMaturity, 'day')
matDateStr = datestr(matDate)
couRate = couRate/100
s = convfactor(refDateStr, matDateStr, couRate)
```

Where:

- function [s] = myConvFactor(price, tMaturity, couRate) is the name of the new formula.
- refDate = fbusdate(2010, 9) sets the contract delivery month to September.
- refDateStr = datestr(refDate) converts September to be a string value (required by MATLAB).
- matDate = addtodate(datenum('01-Jan-1900 00:00:00'), tMaturity, 'day') converts the maturity date value CQG sends over from starting in 1900 to be starting at zero (required by MATLAB).
- matDateStr = datestr(matDate) converts the maturity date to be a string value (required by MATLAB).
- couRate = couRate/100 converts the CQG value to be a decimal (required by MATLAB).

Now, MATLAB has the necessary information to calculate the conversion factor (last line in the formula).

Modifying a MATLAB Formula

Next, is the modification of the MATLAB implied repo rate formula to include the date parameters and to include the new conversion factor calculation.

The modified MATLAB implied repo rate formula is:

```
function [ t ] = myImpliedRepoCFDates(price,tPrice, fPrice,
couRate, tMaturity);
cFactor = myConvFactorDateSept(price,tMaturity, couRate)
cFactor = round(cFactor*10000)/10000
Datenum = today
Busday = busdate(Datenum, 1)
futSettleStr = datestr(Busday)
Delivery = lbusdate(2010, 9)
deliveryStr = datestr(Delivery)
couRate = couRate / 100
Maturity = addtodate(datenum('01-Jan-1900 00:00:00'),
tMaturity, 'day')
tMaturityStr = datestr(Maturity)
t = bndfutimprepo( tPrice, fPrice, futSettleStr, deliveryStr,
cFactor, couRate, tMaturityStr)
```

Where:

- function [t] = myImpliedRepoCFDates(price,tPrice, fPrice, couRate, tMaturity) is the name of the new formula.
- cFactor = myConvFactorDateSept(price,tMaturity, couRate) uses the new conversion factor.
- cFactor = round(cFactor*10000)/10000 modifies the conversion factor to be only four decimal places.
- Datenum = today is determining today's date.
- Busday = busdate(Datenum, 1) is setting the settlement date for purchasing the cash Treasury.
- futSettleStr = datestr(Busday) converts the settlement date to a string value.
- Delivery = lbusdate(2010, 9) determines the last business date of the delvery month.
- deliveryStr = datestr(Delivery) converts the date to a string.
- couRate = couRate / 100 coverts the coupon rate to a decimal form
- Maturity = addtodate(datenum('01-Jan-1900 00:00:00'), tMaturity, 'day') converts the maturity date value CQG sends over from starting in 1900 to be starting at zero
- tMaturityStr = datestr(Maturity) converts the maturity date to be a string value.

Now Matlab has all of the necessary information to calculate the implied repo rate (last line in the formula).

CQG Custom Study

The custom study that sends the data to MATLAB for calculating the implied repo rate of a cash Treasury for deliver against the September 10-year T-note is as follows:

```
matlab (@, 1, myImpliedRepoCFDatesSept,sharescale(Close(@),inte
ger1), Label(@,LastTradeorSettle), Label(TYAU0,LastTradeorSettle),
Label(@,CouponRate), Label(@,MaturityDate))
```

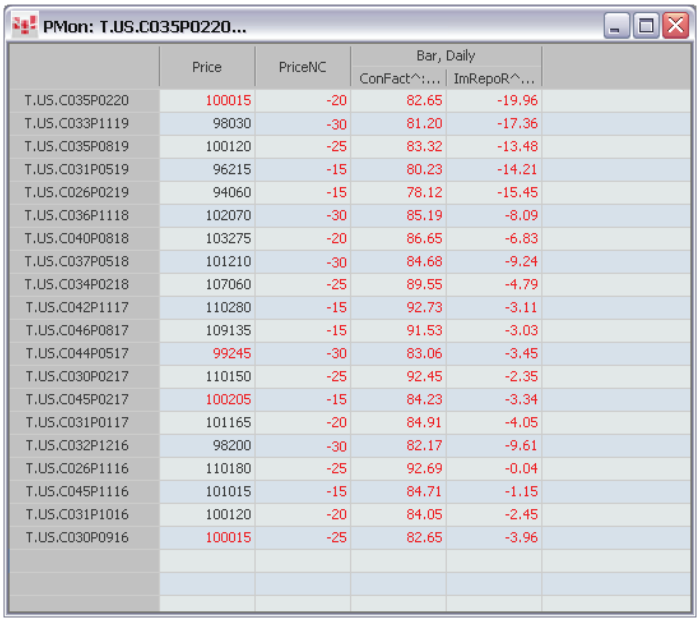
Where:

- matlab (@, 1, is used to call from MATLAB the current value of the MATLAB function.
- myImpliedRepoCFDatesSept is the MATLAB modified formula detailed above.
- sharescale(Close(@),integer1) is converting the MATLAB output to a decimal.
- Label(@,LastTradeorSettle) sends to MATLAB the last price of the cash Treasury.
- Label(TYAU0,LastTradeorSettle) sends to MATLAB the last price of the September 10-year T-note futures price.
- Label(@,CouponRate) sends to MATLAB the coupon rate of the cash Treasury.
- Label(@,MaturityDate) sends to MATLAB the maturity date of the cash Treasury.

Figure 2 is an example of using the implied repo rate study and conversion factor study presented here using the CQG Portfolio Monitor. Both the conversion factor and the implied repo rate are calculated and displayed using two custom studies where MATLAB functions are applied to a basket of deliverable cash Treasuries for the 10-year futures contract.

Summary

As a Third Party Solution Provider for MATLAB, CQG opens up a wide array of research opportunities and functionality for users of both CQG and MATLAB. Here, we have shown just two examples of using MATLAB for common calculations for traders in the fixed income arena. CQG data, graphical analytics, and order routing capabilities integrated with MATLAB tools boxes creates a powerful combination.



	Price	PriceNC	Bar, Daily ConFact^:...	Bar, Daily ImRepoR^...
T.US.C035P0220	100015	-20	82.65	-19.96
T.US.C033P1119	98030	-30	81.20	-17.36
T.US.C035P0819	100120	-25	83.32	-13.48
T.US.C031P0519	96215	-15	80.23	-14.21
T.US.C026P0219	94060	-15	78.12	-15.45
T.US.C036P1118	102070	-30	85.19	-8.09
T.US.C040P0818	103275	-20	86.65	-6.83
T.US.C037P0518	101210	-30	84.68	-9.24
T.US.C034P0218	107060	-25	89.55	-4.79
T.US.C042P1117	110280	-15	92.73	-3.11
T.US.C046P0817	109135	-15	91.53	-3.03
T.US.C044P0517	99245	-30	83.06	-3.45
T.US.C030P0217	110150	-25	92.45	-2.35
T.US.C045P0217	100205	-15	84.23	-3.34
T.US.C031P0117	101165	-20	84.91	-4.05
T.US.C032P1216	98200	-30	82.17	-9.61
T.US.C026P1116	110180	-25	92.69	-0.04
T.US.C045P1116	101015	-15	84.71	-1.15
T.US.C031P1016	100120	-20	84.05	-2.45
T.US.C030P0916	100015	-25	82.65	-3.96

Figure 2: MATLAB functions can be displayed as values or charted. Here are the conversion factors and implied repo rates for a basket of deliverable cash Treasuries against the 10-year Treasury note futures contract.

Contact Us

www.cqg.com | websales@cqg.com | Or call your CQG sales representative
US 1-800-525-7082 | **Japan** +81 (0) 3-3286-6633 | **UK** +44 (0) 20-7827-9500 | **Russia** +7 495-795-2410
France +33 (0) 1-74-18-07-81 | **Germany** +49 (0) 69-6677-7558-0 | **Singapore** +65 6494-4911 | **Australia** +61 (2) 9235-2009