June 2022 – Final Project Brief

Please see below instructions for the Final Project.

Assessment for Module Six is carried out by the means of a programming project. It is designed to give an opportunity for further study of numerical methods required to implement and validate a quantitative model.

CQF Final Project is numerical techniques and backtest or sensitivity-test of model output (prices, portfolio allocations, pairs trading) as appropriate. Some numerical techniques call for being implemented in code from first principles. There might be numerical methods either too involved or auxiliary which you would not need to implement if good ready functionality is available. Re-use and adoption of code permitted. Marks earned will strongly depend on coding of numerical techniques and presentation of how you explored and tested a quantitative model.

A capstone project require own study and ability to work with documentation on packages that implement numerical methods in your coding environment e.g., Python, R, Matlab, C#, C++, Java. You do not need to pre-approve the coding language and use of libraries, including very specialized tools such as Scala, kdb+ and q. However, software like EViews is not coding.

To complete the project, you must code the model(s) and its numerical techniques form **one** topic from the below options and write an analytical report. If you continue from a previous cohort, please review topic description because tasks are regularly reviewed. It is not possible to submit past topics.

1. Credit Spread for a Basket Product (**CR**)
2. Deep Learning for Financial Time Series (**DL**)
3. Interest Rate Modeling or Counterparty Credit Risk – IR Forward / LIBOR (**IR**)
4. Long / Short Trading Strategy Design & Backtest (**TS**)
5. Portfolio Construction using Black-Litterman Model and Factors (**PC**)

Submission Requirements

- First recommendation: If you begin with Python Notebook, there is work to be done to transform it into an analytical report. Remove printouts of large tables/data/output. Write up mathematical sections (with LaTeX markup). Write up analysis and comparison for results and stress-testing (or alike analysis). Explain your plots. Think like a quant consider convergence/accuracy/variance and bias as well as computational properties. Make a table of the numerical techniques you coded/utilized.
- It is optimal to save Python Notebook reports as HTML but do include a PDF (with page numbers for us to reference to).
- Submit working code, that faculty can run independently together with a well-written report and originality declaration.
- Project report to have an exact title from the list above and content must correspond to it.
- There is no set page length. Report must have an analytical quality and discussion of results/robustness/sensitivity/backtesting as appropriate to the topic.
- Use charts, test cases and comparison to empirical research papers where available.
- Report must contain sufficient mathematical model, numerical methods with attention to their convergence/accuracy/computational properties.
- Please feature the numerical techniques you coded – make a table.
- Mathematical sections can be prepared using LaTeX or Equation Editor (Word). Printing out Python notebook code and numbers on multiple pages, without your own analysis text, explained plots, relevant maths is not an acceptable format.

Submissions to be uploaded to online portal only. Upload format: one written report, (PDF), one zip archive with code and data files, and one signed declaration (PDF).

Submission date for the project is **Monday 23rd January 2023, 23.59 GMT**

Absolutely no extensions will be accepted, any delegate who fails to hand in their work will be deferred with no exceptions.

Work done must match the Brief. There is no extension to the Final Project.
Projects without declaration or working code are incomplete and will be returned.
All projects are checked for originality. We reserve an option of a *viva voce* before a qualification is to be awarded.

Please note, the project results will be made available Monday 20th March 2023

Upload Instructions

Please upload only your final completed version.

- To submit your answer file, please click start assessment under module 6. Use the upload function to upload one zip file containing a PDF report, coding and data files and project declaration.
- For our download and processing scripts to work, it is absolutely necessary to name the project report as ONE file, HTML OR PDF, named with two-letter project code as follows: TS John Smith REPORT.pdf or PC Xiao Wang REPORT.pdf . The name must be as registered on CQF Portal.
- All other files, code and a pdf declaration (if not front page of the report) has to be uploaded as ONE zip file, for example TS John Smith CODE.zip. In this zip include pdf-convertedreport (from Python or HTML).
- Do not submit files with generic names, such as CODE.zip, FinalProject.zip, Final Project Declaration.pdf, etc.
- Do not submit unzipped .py and other code files as cloud anti-virus likely to have the issue.

For any queries on these requirements contact CQFProgram@fitchlearning.com

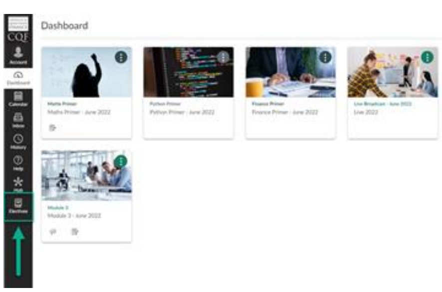
Project Support

Advanced Electives
To gain background knowledge in a focused way, we ask you to review two Advanced Electives. Electives canvass knowledge areas and can be reviewed before/at the same time/closer to writing up Analysis and Discussion (explanation of your results).

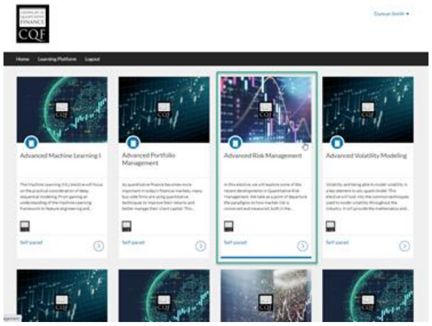
- There is no immediate match between Project Topics and Electives
- Several viable combinations for each Project Topic are possible
- One elective learning strategy is to select one `topical elective' and one `coding elective'

To access the electives:

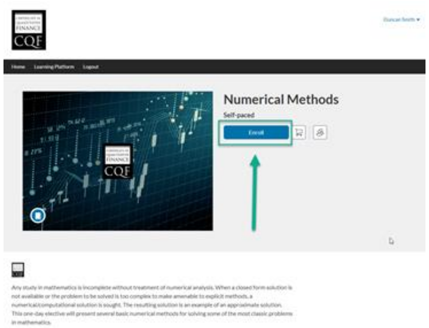
Login to the CQF Learning Hub
Click the *Learning Platform* button to sign into Canvas



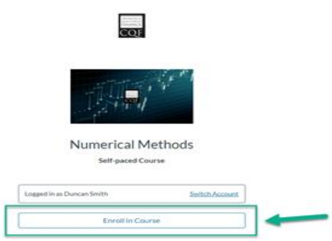
Click on *Electives* button on global navigation menu



You will be redirected to the electives Catalogue, where you can view and review all electives available to you. Full descriptions for each elective can be found here.

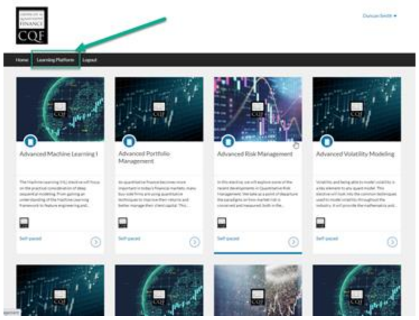


When on an elective click the *enrol* button



You will see the confirmation page, click the *enrol in Course* button to confirm your selection

You will land on the successful enrolment page, where you can click to start the elective or return to the catalogue page



When on the catalogue page you can click the *Learning Platform* link to return to Canvas. Your electives selected will appear on your learning dashboard

Workshop & Tutorials

Each project title is supported by a faculty alongside of set of project workshops and tutorials.

DATE	TITLE	TIME
10/12/2022	Final Project Workshop I	13:00 – 15:30 GMT
17/12/2022	Final Project Workshop II	13:00 – 15:30 GMT
16/12/2022	Final Project Tutorial I	12:00 – 13:30 GMT 18:00 – 19:30 GMT
19/12/2022	Final Project Tutorial II	12:00 – 13:30 GMT 18:00 – 19:30 GMT
21/12/2022	Final Project Tutorial III	12:00 – 13:30 GMT 18:00 – 19:30 GMT

Faculty Support Details

Title: Credit Spread for a Basket Product
Project Code: CR
Lead: Riaz Ahmad
Email: Riaz.Ahmad@fitchlearning.com

Title: Deep Learning for Financial Time Series
Project Code: DL
Lead: Kannan Singaravelu
Email: Kannan.Singaravelu@fitchlearning.com

Title: Interest Rate Modeling or Counterparty Credit Risk – IR Forward / LIBOR
Project Code: IR
Faculty Lead: Richard Diamond
Email: Richard.Diamond@fitchlearning.com

Title: Long / Short Trading Strategy Design & Backtest
Project Code: TS
Faculty Lead: Richard Diamond
Email: Richard.Diamond@fitchlearning.com

Title: Portfolio Construction using Black-Litterman Model and Factors
Project Code: PC
Faculty Lead: Panos Paras
Email: panos.parpas@fitchlearning.com

Portfolio Construction using Black-Litterman Model and Factors

Summary

Construct a factor-bearing portfolio, compute at least two kinds of optimisation. Within each optimisation, utilise the Black-Litterman model to update allocations with absolute and relative views. Compute optimal allocations for three common levels of risk aversion (Trustee/Market/Kelly Investor). Implement systematic backtesting: which includes both, regressing results of your portfolio on factors and study of the factors themselves (wrt the market excess returns).

Kinds of optimisation: mean-variance, Max Sharpe Ratio, higher-order moments (min coskewness, max cokurtosis) – implement <u>at least two</u>. Min Tracking Error also possible but for that your portfolio choice will be measured against a benchmark index. Computation by ready formula or specialised for quadratic programming. Adding constraints improves robustness: most investors have margin constraints / limited ability to borrow / no short positions.

OPTIONALLY, Risk Contributions can also computed *ex ante* for any optimal allocation, whereas computing ERC Portfolio requires solving a system of risk budget equations (non-linear). ERC computation is not an optimisation, however can be 'converted' into one – sequential quadratic programming (SQP).

Portfolio Choice and Data

The choice of portfolio assets must reflect optimal diversification – according to your own understanding and approach. The simplest technical choice is choosing assets with the least correlation. For exposure/tilts to factor(s) – you need factor betas *a priori*, and include assets with either high or low beta, depending on the purpose specified by you.

A naive portfolio of S&P500 large caps can be said to be exposed to one factor, which is insufficient. A specialised portfolio for an industry, emerging market, credit assets should have 5-15 names (guide number), and > 3 notionally uncorrelated assets, such as commodity, VIX, bonds, credit spreads, real estate.

Factor time series (or several) can represent your uncorrelated asset – including factors as assets at the start and giving them preferential BL views is one way of systemic implementation of factor tilts.

- Mean-variance optimisation was specified by Harry Markowitz for simple returns (not log) which are *in excess* of the r_f. For risk-free rate, 3M US Treasury from pandas FRED dataset/ECB website rates for EUR/some small constant rate/zero rate – all are acceptable. Use 2-3 year sample, which means > 500 daily returns.
- Source for prices data is Yahoo!Finance (US equities and ETFs). Use code libraries to access that, Google Finance, Quandl, Bloomberg, Reuters and others. If benchmark index not available, equilibrium weights computed from the market cap (dollar value).
- In this variation of PC topic, it is necessary to introduce 2-3 factor time series and treat them as investable assets (5 Fama-French factors). If using Smart Beta ETFs present on their structure – you might find there is no actual long/short factors, just a long-only collection of assets with particularly high betas.

Coding for Quant Finance

Choose programming environment that has appropriate strengths and facilities to implement the topic (pricing model). Common choice is Python, Java, C++, R, Matlab. Exercise judgement as a quant: which language has libraries to allow you to code faster, validate easier.

Use of R/Matlab/Mathematica is encouraged. Often there a specific library in Matlab/R gives fast solution for specific models in robust covariance matrix/cointegration analysis tasks.

Project Brief give links to nice demonstrations in Matlab, and Webex sessions demonstrate Python notebooks { doesn't mean your project to be based on that ready code.

Python with pandas, matplotlib, sklearn, and tensorow forms a considerable challenge to Matlab, even for visualization. Matlab plots editor is clunky and it is not that difficult to learn various plots in Python.

‘Scripted solution' means the ready functionality from toolboxes and libraries is called, but the amount of own coding of numerical methods is minimal or non-existent. This particularly applies to Matlab/R.

Projects done using Excel spreadsheet functions only are not robust, notoriously slow and do not give understanding of the underlying numerical methods. CQF-supplied Excel spreadsheets are a starting point and help to validate results but coding of numerical techniques/use of industry code libraries is expected.

The aim of the project is to enable you to code numerical methods and develop model prototypes in a production environment. Spreadsheets-only or scripted solutions are below the expected standard for completion of the project.

What should I code? Delegates are expected to re-code numerical methods that are central to the model and exercise judgement in identifying them. Balanced use of libraries is at own discretion as a quant.

Produce a small table in report that lists methods you implemented/adjusted. If using ready functions/borrowed code for a technique, indicate this and describe the limitations of numerical method implemented in that code/standard library.

It is up to delegates to develop their own test cases, sensibility checks and validation. It is normal to observe irregularities when the model is implemented on real life data. If in doubt, reflect on the issue in the project report.

The code must be thoroughly tested and well-documented: each function must be described and comments must be used. Provide instructions on how to run the code.

Step-by-Step Instructions

Part I: Factor Data and Study(Backtesting)

1. Implement Portfolio Choice based on your approach to optimal diversification. Usually the main task is to select a few assets that gives risk-adjusted returns the same as/outperforms a much larger, naturally diversified benchmark such as S&P500. See Q&A document distributed at the Workshop.

2. Experiment which factors you are going to introduce, collect their time series data or compute.

   - The classic Fama-French factors are HML (value factor) and SMB (small business). RMW (robust vs. weak profitability) and CMA (conservative vs aggressive capex) are the new factors and you can experiment with them.
   - Exposure to sector or style can also be considered a factor.
   - It very recommended that you introduce an interesting, custom factor such as Momentum, BAB (betting against beta) – likely you will need to compute time series of its returns, however that can be as simple as returns from a short portfolio of top five tech stocks.

3. The range of portfolios, for which factors are backtested, is better explained at source `http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html`

4. Present P&L returns and Systematic Backtesting of your factors vs the Market (index of your choice), which includes performance, present plots of rolling beta and changing alpha. Ideally, you can present results for each factor beta independently and then, in combination. This work to be presented even before you engage in portfolio optimisation

Part II: Comparative Analysis of BL outputs

1. Plan your Black-Litterman application. Find a ready benchmark or construct the prior: equilibrium returns can come from a broad-enough market index. Implement computational version of BL formulae for the posterior returns.

2. Imposing too many views will make seeing impact of each individual view difficult.

3. Describe analytically and compute optimisation of **at least two kinds**. Optimisation is improved by using sensible constraints, eg, budget constraint, 'no short positions in bonds' but such inequality constraints $\forall w_i > 0$ trigger numerical computation of allocations. e.

4. You will end up with multiple sets of optimal allocations, even for a classic mean-variance optimisation (your one of two kinds). Please make your own selection on which results to focus your Analysis and Discussion – the most feasible and illustrative comparisons.

   - Optimal allocations (your) vs benchmark for active risk. Expected returns (naïve) vs implied equilibrium returns (alike to Table 6 in BL Guide by T. Idzorek.)
   - BL views are not affected by covariance matrix – therefore, to compute allocations shifted by views (through Black-Litterman model) with naive or robust covariance is your choice.

- Three levels of risk aversion – it is recommended that you explore at least for classical Min Var optimisation.

5. There is no rebalancing task for the project, particularly because posterior BL allocations expected to be durable.
6. Compare performance of your custom portfolio vs factors and market (rolling beta), independently and jointly. OPTIONALLY, compare performance of your portfolio to $1/N$ allocations / Diversification Ratio portfolio / Naive Risk Parity kind of portfolio and perform the systematic backtesting of that portfolio *wrt* to factors.

Pairs Trading Strategy Design & Backtest

Estimation of cointegrated relationship between prices allows to arbitrage a mean-reverting spread. Put trade design and backtesting in the centre of the project, think about your signal generation and backtesting of the P&L. You will have a hands-on experience with regression but will not run the regression on returns. The numerical techniques are regression computation in matrix form, Engle-Granger procedure, and statistical tests. You are encouraged to venture into A) multivariate cointegration (Johansen Procedure) and B) robustness checking of cointegration weights, ie, by adaptive estimation of your regression parameters with statistical filters.

Contegrating weights that you use to enter the position form the long/short allocation that produces a mean-reverting spread. Signal generation and suitability of that spread for trading depend on its fitting to OU process recipe. For optimisation, comparative backtesting, rolling ratios and other industry-level backtesting analytics use the ready code libraries. However, project that solely runs pre-programmed statistical tests and procedures on data is insufficient. It is not recommended using VBA for this topic due to lack of facilities.

Signal Generation and Backtesting

- Be inventive beyond equity pairs: consider commodity futures, instruments on interest rates, and aggregated indices.
- Arb is realised by using cointegrating coefficients $\boldsymbol{\beta}_{Coint}$ as allocations $\boldsymbol{w}$. That creates a long-short portfolio that generates a mean-reverting spread. All project designs should include trading signal generation (from OU process fitting) and backtesting (drowdown plots, rolling SR, rolling betas).
- Does cumulative P&L behave as expected for a cointegration arb trade? Is P&L coming from a few or many trades, what is half-life? Maximum Drawdown and behaviour of volatility/VaR?
- Introduce liquidity and algorithmic flow considerations (a model of order flow). Any rules on accumulating the position? What impact bid-ask spread and transaction costs will make?

Step-by-Step Instructions

Can utilise the ready multivariate cointegration (R package *urca*) to identify your cointegrated cases first, especially if you operate with the system such as four commodity futures (of different expiry but for the period when all traded. 2-3 pairs if analysing separate pairs by EG.

Part I: Pairs Trade Design

1. Even if you work with pairs, re-code regression estimation in matrix form – your own OLS implementation which you can re-use. Regression between stationary variables (such as DF test regression/difference equations) has OPTIONAL model specification tests for (a) identifying optimal lag p with AIC BIC tests and (b) stability check.
2. Implement Engle-Granger procedure for each your pair. For Step 1 use Augmented DF test for unit root with lag 1. For Step 2, formulate both correction equations and decide which one is more significant.

3. Decide signals: common approach is to enter on bounds $\mu_e \pm Z\sigma_{eq}$ and exit on e_t everting to about the level μ_e.

4. At first, assume $Z = 1$. Then change Z slightly upwards and downwards – compute P&L for each case of widened and tightened bounds that give you a signal. Alternatively run an optimisation that varies Z_{opt} for $\mu_e \pm Z_{opt}\sigma_{eq}$ and either maximises the cumulative P&L or another criterion.

   Caution of the trade-off: wider bounds might give you the highest P&L and lowest N_{trades} however, consider the risk of co-integration breaking apart.

5. OPTIONALLY you can use own understanding of multivariate cointegration analysis and R/Python VECM packages in order select the best candidates for pairs trading (or even basket trading). Do not use all five Zivot's 'deterministic trends' in coint residual – in practice one only need a constant inside the residual e_{t-1}.

Part II: Backtesting (below is OPTIONAL COMBINATION)

It is your choice as a quant to decide which elements you need to argue successfully that your trading strategy (a) will not fall apart and (b) provides 'uncorrelated return'.

4. Perform systematic backtesting of your trading strategy (returns from a pairs trade) to produce drawdown plots and rolling Sharpe Ratio. Explain why rolling beta might not be as relevant to stat arb/alternative algorithmic and market-making strategies).

5. Industry backtesting relies on rolling betas, while scientific research will test for breakouts using LR test. Cointegrated relationship supposed to persist and β'_{Coint} should stay the same: continue delivering the stationary spread over 3-6 months without the need to be updated. Is this realistic for your pair(s)?

   Discuss benefits and disadvantages of regular re-estimation of cointegrated relationships by shifting data 1-2 weeks (remember to reserve some future data), and report not only on rolling β'_{Coint}, but also Engle-Granger Step 2, the history of value of test statistic for the coefficient in front of EC term.

   Would you implement something like Kalman filter/particle filter adaptive estimation [applied to cointegrated regression] in order to see the updated β'_{Coint} and μ_e? Reference: `www.thealgoengineer.com/2014/online_linear_regression_kalman_filter/`.

6. Think of Machine Learning-inspired backtesting, such as splitting data into train/test subsets, preprocessing, and crossvalidaiton as appropriate and feasible (beware of cross-validation issues with time series analysis).

Time Series Project Workshop, Cointegration Lecture and Pairs Trading tutorial are your key resources.

Credit Spread for a Basket Product

Price a fair spread for a portfolio of CDS for 5 reference names (Basket CDS), as an expectation over the joint distribution of default times. The distribution is unknown analytically and so, co-dependent uniform variables are sampled from a copula and then converted to default times using a marginal term structure of hazard rates (separately for each name). Copula is calibrated by estimating the appropriate default correlation (historical data of CDS differences is natural candidate but poses market noise issue). Initial results are histograms (uniformity checks) and scatter plots (co-dependence checks). Substantial result is sensitivity analysis by repricing.

A successful project will implement sampling from both, Gaussian and t copulae, and price all k-th to default instruments (1st to 5th). Spread convergence can require the low discrepancy sequences (e.g., Halton, Sobol) when sampling. Sensitivity analysis *wrt* inputs is required.

Data Requirements

Two **separate** datasets required, together with matching discounting curve data for each.

1. **A snapshot of credit curves** on a particular day. A debt issuer likely to have a USD/EUR CDS curve – from which a term structure of hazard rates is bootstrapped and utilised to obtain exact default times, $u_i \to \tau_i$. In absence of data, spread values for each tenor can be assumed or stripped visually from the plots in financial media. The typical credit curve is concave (positive slope), monotonically increasing for $1Y, 2Y, \ldots, 5Y$ tenors.

2. **Historical credit spreads time series** taken at the most liquid tenor $5Y$ for each reference name. Therefore, for five names, one computes 5×5 default correlation matrix. Choosing corporate names, it is much easier to compute correlation matrix from equity returns.

   Corporate credit spreads are unlikely to be in open access; they can be obtained from Bloomberg or Reuters terminals (via your firm or a colleague). For sovereign credit spreads, time series of ready bootstrapped PD_{5Y} were available from DB Research, however, the open access varies. Explore data sources such as `www.datagrapple.com` and `www.quandl.com`.

   Even if CDS_{5Y} and PD_{5Y} series are available with daily frequency, the co-movement of daily changes is market noise *more* than correlation of default events, which are rare to observe. Weekly/monthly changes give more appropriate input for default correlation, however that entails using 2-3 years of historical data given that we need at least 100 data points to estimate correlation with the degree of significance.

If access to historical credit spreads poses a problem remember, default correlation matrix can be estimated from historic equity returns or debt yields.

Step-by-Step Instructions

1. For each reference name, bootstrap implied default probabilities from quoted CDS and convert them to a term structure of hazard rates, $\tau \sim Exp(\hat{\lambda}_{1Y}, \ldots, \hat{\lambda}_{5Y})$.

2. Estimate default correlation matrices (near and rank) and d.f. parameter (ie, calibrate copulæ). You will need to implement pricing by Gaussian and t copulæseparately.

3. Using sampling form copula algorithm, repeat the following routine (simulation):

   (a) Generate a vector of correlated uniform random variable.

   (b) For each reference name, use its term structure of hazard rates to calculate exact time of default (or use semi-annual accrual).

   (c) Calculate the discounted values of premium and default legs for every instrument from *1st* to *5th*-to-default. Conduct MC separately or use one big simulated dataset.

4. Average premium and default legs across simulations separately. Calculate the fair spread.

Model Validation

- The fair spread for k*th*-to-default Basket CDS should be less than k-1 to default. Why?

- Project Report on this topic should have a section on **Risk and Sensitivity Analysis** of the fair spread *w.r.t.*

  1. default correlation among reference names: either stress-test by constant high/low correlation or ± percentage change in correlation from the actual estimated levels.
  2. credit quality of each individual name (change in credit spread, credit delta) as well as recovery rate.

  Make sure you discuss and compare sensitivities for all five instruments.

- Ensure that you explain historical sampling of default correlation matrix and copula fit (uniformity of pseudo-samples) – that is, Correlations Experiment and Distribution Fitting Experiment as will be described at the Project Workshop. Use histograms.

Copula, CDF and Tails for Market Risk

The recent practical tutorial on using copula to generate correlated samples is available at: `https://www.mathworks.com/help/stats/copulas-generate-correlated-samples.html`
Semi-parametric CDF fitting gives us percentile values with fitting the middle and tails. Generalised Pareto Distribution applied to model the tails, while the CDF interior is Gaussian kernel-smoothed. The approach comes from Extreme Value Theory that suggests correction for an Empirical CDF (kernel fitted) because of the tail exceedances.

`http://uk.mathworks.com/help/econ/examples/using-extreme-value-theory-and-copulas-to-evaluate-market-risk.html`

`http://uk.mathworks.com/help/stats/examples/nonparametric-estimates-of-cumulative-distribution-functions-and-their-inverses.html`

Interest Rate Modeling for Counterparty Risk

Please do not attempt this topic without prior experience in interest rates.
We offer this topic as a challenge for experienced quants.

This topic recognises the importance of credit and counterparty risk adjustments (incremental risk) to the derivatives business. The task here is to compute CVA on the Interest Rate Swap. Full exposure analytics can only be produced by a model that simulates the full curve suitably calibrated on the recent data. With the LIBOR Market Model, please do not be confused by the name: LIBOR fixings practice has been stopped, however the quantities to model, term forward rates and their log-normal SDE remain the same. Post-LIBOR transition encouraged hybrid LMM/HJM models (see Fabio Mercurio talks and SSRN on Forward Market Model for your own advanced study). However, the fidelity of LMM being truthful/simulating from market quantities always gave way to the practice of factor-reduced form of LMM, where factors alike principle components were used instead of term volatilities.

Data Requirements The inputs for IRS valuation are forward rates and discounting factors as simulated by the model. Simple version of LMM calibration requires market cap data – can be easily assumed or copied from Gatarek textbook to begin with. Advanced LMM calibration (very optional) takes into account swaption volatilities (Rebonato method).

Calculate the credit valuation adjustment taken by Counterparty A for an interest rate swap instrument using credit spreads for Counterparty B (the payer of floating leg). Only one-day CDS curve is required for default probabilities bootstrap – if market data is a problem assume a slightly convex curve, eg, 30 bps for 1Y, 35 bps for 2Y, 45 bps for 3Y and alike. At each tenor, plot IRS MtM values and produce full exposure distribution at each MtM valuation point: max, median, quartiles and 97.5th percentile. Expected Exposure is defined as $\max\left(\text{MtM}_\tau, 0\right)^+$ which is very drastic and overstates the CVA. Potential Future Exposure (PFE) at the 97.5th percentile and median of positive exposure should show less convex exposure over tenors. Having compared the exposure at maximum vs. percentiles, consider the very small or negative rates and develop sensitivity analysis. Illustrate the concept of the wrong-way risk.

Steps-By-Step Instructions for IRS Pricing and CVA

- Probability of default is bootstrapped from credit spreads for a reference name in $0.5Y$ increment. Linear interpolation over spreads and ready PD bootstrapping spreadsheet are acceptable, $RR = 40\%$. CVA LGD also 40%.

- Assume the swap is written on 6M LIBOR over 5Y. Notional $N = 1$ and $\tau = 0.5$. For the values of Forward LIBOR L_{6M} at times $T_1, T_2, T_3, \ldots, T_{N-1}$ use the full simulated curve.

- Define MtM position as Floating Leg − Fixed Leg = $(L_{6M} - K)$ appropriately discounted. Depending today's curve, choose fixed leg (rate) K to have a positive exposure.

Libor Market Model provides an iterative output: the curve is evolved column by column – the last column has only one terminal rate $L(T_{n-1})$ which SDE has zero drift, reflecting the terminal measure $\mathbb{Q}(T_n)$. Each tenor of forward rate has is own SDE and the terminal tenor will have a very simple drift and diffusion (the most complex to compute is the short-term rates drift). Be prepared if the *abcd* fitted volatility delivers a poor re-pricing of market caps from which you calibrated. To match those market cap prices, an ad-hoc curve-wide adjustment to simulated forward rates would be required but no need to strive here.

Simplifying Assumptions: 1) Gatarek example implement calendar dates and so, their year fraction 0.2528 or 0.2558 depends on day count caplet covers but 30/360 convention acceptable. 2) Commonly $\text{cpl}(3M, 6M)$, $\text{cpl}(6M, 9M)$ are computed just using flat volatility of $\sigma_{Mkt}(t, 12M)$ 3) Do not perform time homogeneity check, instead use assignment described in Gatarek 7.4.2. 4) LMM is calibrated for ATM strike. Each OTM/ITM strike of the smile will need its own calibration.

Step-by-Step for LMM

Part I: Data

1. You will need market price data – cap price and discount factor (two columns). Caps are quoted in terms of implied volatility $\sigma^{cap}(t, T)$ for period **from** $t = 0$ **to** 1Y, 2Y, 3Y,...

   (a) For pre-simulated caplet data (ie, from the HJM model) the Black formula is conventional means of converting the caplet's cashflow to the implied volatility number.

2. Alternatively but completely OPTIONAL, LMM volatilities can be calibrated from a table of co-terminal vanilla swaptions (European options on forward-starting swaps) where Rebonato method makes the Black formula suitable.

Part II: Calibration by Caplet Volatility Stripping

3. Forward 3M caplet volatilities $\sigma^{cpl}(T_i, T_{i+3M})$ are stripped from market cap volatilities by using cashflow equivalence $\text{cap}_T = \sum_i^T \text{cpl}_i$. Caplet stripping methodology requires Black formula and root-finding for $\text{cpl}(T_{i-1}, T_i) \Leftrightarrow \sigma^{cpl}(T_{i-1}, T_i)$ and pre-computation of

   (a) Forward Swap Rates $S(t; T_{i-3M}, T_i)$ to be used as ATM strikes of forward caplets.

   (b) $\sigma(t, 9M)$, $\sigma_{Mkt}(t, 1Y)$, $\sigma(t, 15M)$, $\sigma(t, 18M)$, $\sigma(t, 21M)$, $\sigma_{Mkt}(t, 2Y)$, $\sigma(t, 27M)$, etc. where non-market values are interpolated.[2]

$$\begin{aligned}
\text{cpl}(6M, 9M) &= \text{cap}(t, 9M) - \text{cpl}(3M, 6M) \\
\text{cpl}(9M, 12M) &= \text{cap}(t, 12M) - \text{cpl}(6M, 9M) - \text{cpl}(3M, 6M)
\end{aligned}$$

[2]You have MANY choices for interpolation 1) linear in variance $\sigma^2(0, T)$, 2) *abcd* function over this implied vol. which is separate from over instant. vol., 3) cubic spline τ, τ^2, τ^3, or monotone-preserving convex spline (Hagan&West 2005). Interpolation techniques applied in other domains like forward curve, but no reason not to make new kind applications. The underlying caplets **must be stripped**, not taken from any interpolated function.

4. Fit the *abcd* instantaneous volatility function, each tenor might have a separate fitting parameter ϕ_i. This is done by sum of least squares argmin $\sum [\sigma(i) - \sigma_{\text{Fitted}}(i)]^2$.

5. Decide on correlation structure, from simple parametric $\rho_{ij} = exp\,(-\beta(T_i - T_j))$ to Schoenmakers and Coffey (2003) vs. empirical correlations.

With these steps you have all inputs in order to set up LMM SDE and output the forward curves under Monte-Carlo. Remember that LMM output is direct simulation over $dt = 0.25$.

Deep Learning for Financial Time Series

Summary

Trend prediction has drawn a lot of research for many decades using both statistical and computing approaches including machine learning techniques. Prediction of asset direction is paramount for investment management as accurate forecasts could ensure asset managers outperform the market. Trend prediction remains a challenging task due to the semi-strong form of market efficiency, high noise-to-signal ratio, and the multitude of factors that affect asset prices including but not limited to the stochastic nature of underlying instruments. However, financial time series can be modeled effectively using sequence modeling approaches like a recurrent neural network.

Objective

Your objective is to produce a model to predict positive moves (up trend) using the Long Short-Term Memory model. Your proposed solution should be comprehensive with the detailed model architecture, evaluated with a backtest applied to a trading strategy.

- Choose one ticker of your interest from the index, equity, ETF, crypto token, or commodity.
- Predict trend only, for a short-term return (example: daily, 6 hours). Limit prediction to binomial classification: the dependent variable is best labeled [0, 1]. Avoid using [-1, 1] as class labels.
- Analysis should be comprehensive with detailed feature engineering, data pre-processing, model building, and evaluation.

Note: You are free to make study design choices to make the task achievable. You may redefine the task and predict the momentum sign (vs return sign) or direction of volatility. Limit your exploration to ONLY one asset. At each step, the process followed should be expanded and explained in detail. Use of TensorFlow is strongly recommended [avoid PyTorch]. Merely presenting python codes without a proper explanation shall not be accepted. The report should present the study in a detailed manner with a proper conclusion. Code reproducibility is a must and the use of modular programming approaches is recommended. Under this topic, you do not recode existing indicators, libraries, or optimization to compute neural network weights and biases.

Step-by-Step Instructions

1. The problem statement should be explicitly specified without any ambiguity including the selection of underlying assets, datasets, timeframe, and frequency of data used.

    - If predicting short-term return signs (for the daily move), then training and testing over up to 5 years should be sufficient. If you attempt the prediction of 5D, 10D return for equity or 1W, 1M for the Fama French factor, you'll have to increase the data required to at least 10 years.

2. Perform exhaustive Feature Engineering (FE).

    - FE should be detailed including the listing of derived features and specification of the target/label. Devise your approach on how to categorize extremely small near-zero returns (drop from the training sample or group with positive/negative returns). The threshold will strongly depend on your ticker. *Example:* small positive returns below 0.25% can be labelled as negative.
    - Class imbalances should be addressed - either through model parameters or via label definition.
    - Use of features from cointegrated pairs and across assets is permitted but should be tactical about design. There is no one recommended set of features for all assets; however, the initial feature set should be sufficiently large. Financial ratios, advanced technical indicators including volatility estimators, and volume information can be a predictor for price direction.
    - OPTIONAL Use of news heatmap, credit spreads (CDS), historical data for financial ratios, history of dividends, purchases/disposals by key stakeholders (director dealings) or by large funds, or Fama-French factor data can enhance your prediction and can be sourced from your professional subscription.

3. Conduct a detailed Exploratory Data Analysis (EDA).

    - EDA helps in dimensionality reduction via a better understanding of relationships between features and uncovers underlying structure, and invites detection/explanation of the outliers. The choice of feature scaling techniques should be determined by EDA.

4. Proper handling of data is a must. The use of a different set of features, lookback length, and datasets warrant cleaning and/or imputation.

5. Feature transformation should be applied based on EDA.

    - Multi-collinearity analysis should be performed among predictors.
    - Multi-scatter plots presenting relationships among features are always a good idea.
    - Large feature sets (including repeated kinds, and different lookbacks) warrant a reduction in dimensionality in features. Self Organizing Maps (SOM), K-Means clustering, or other methods can be used for dimensionality reduction. Avoid using Principal Component Analysis (PCA) for non-linear datasets/predictors.

6. Perform extensive and exhaustive model building.

    - Design the neural network architecture after extensive and exhaustive study.
    - The best model should be presented only after performing the hyperparameter optimization and compared with the baseline model.
    - The choice and number of hyperparameters to be optimized for the best model are design choices. Use experiment trackers like MLFlow or TensorBoard to present your study.

7. The performance of your proposed classifier should be evaluated using multiple metrics including backtesting of the predicted signal applied to a trading strategy.

    - Investigate the prediction quality using AUC, confusion matrix, and classification report including balanced accuracy.
    - Predicted signals should be evaluated by applying them to a trading strategy.

* * *

Reading List 2022-June

The list puts together some initial – you will find more topical resources inside Additional Material.zip, provided for each Topic within the relevant Project Workshop I or II, or within the anchor core lecture, such as on Neural Nets. Please do not email tutors for a copy.

Reading List: Credit Portfolio

- Very likely you will revisit *CDO & Copula Lecture* material, particularly slides 48-52 that illustrate Elliptical copula densities and discuss Cholesky factorisation.
- *Sampling from copula* algorithm is in *relevant Workshop* and *Monte Carlo Methods in Finance* textbook by Peter Jaekel (2002) – see Chapter 5.
- Rank correlation coefficients are introduced *Correlation Sensitivity Lecture* and P. Jaekel (2002) as well. CR Topic Q&A document gives the clarified formulae and explanations.

Reading List: Portfolio Construction

- CQF Lecture on *Fundamentals of Optimization and Application to Portfolio Selection*
- *A Step-by-step Guide to The Black-Litterman Model* by Thomas Idzorek, 2002 tells the basics of what you need to implement.
- *The Black-Litterman Approach: Original Model and Extensions* Attilio Meucci, 2010. `http://ssrn.com/abstract=1117574`
- On LW nonlinear shrinkage / Marcenko-Pastur denoising, either method to make a covariance matrix robust, resources and certain code provided with the relevant Workshop and Tutorial.

Reading List: Cointegrated Pairs

- *Modeling Financial Time Series*, E. Zivot & J. Wang, 2002 – one recommended textbook, we distribute Chapter 12 on Cointegration with the relevant Project Workshop.
- Instead of a long econometrics textbook, read up *Explaining Cointegration Analysis: Parts I and II* by David Hendry and Katarina Juselius, 2000 and 2001. *Energy Journal.*
- Appendices of this work explain key econometric and OU process maths links, *Learning and Trusting Cointegration in Statistical Arbitrage* by Richard Diamond, *WILMOTT* `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2220092`.