Capstone Project Summary:

Aggregation and Visualization of Counterparty Risk

Ozan Kaya

The Client, in its intention to increase its visibility on their business with their counterparties, has asked for a new ETL pipeline where information with various granularity will be extracted from different relational databases, put into shape and loaded for further use of different stakeholders. Client is also aiming for a more agile stress testing capacity and wants to be able to conduct various stress scenarios more frequently which requires recurrent and streamlined testing, controls and reporting.

To provide more clarity, let's think of the overall counterparty risk assessment and review exercise to be consisting of two simple approaches. For assets with sufficiently high trading volume, i.e high liquidity, the relevant end of day quote could be taken as the value of the given asset and/or derivative to calculate the overall value of the respective portfolio. For relatively illiquid assets/positions on the other hand, the value of a given portfolio can be calculated using a model driven approach. Additionally, with the latter approach, stressed counterparty risk metrics can be calculated by changing the given market data for a given day according to a prescribed scenario.

The Client does exactly the same. As an example, we can think about an S&P 500 Future where the Client is a counterparty of. The value of this position is typically found by the end of day quote, so it is market driven. Let's assume a stressed scenario where US Equity markets are declining by 30% instantaneously. This prescribed shock could be used to calculate the new value of our derivative with the help of some financial models. In this example, according to the Firm's position, and assuming no change on interest rates, the derivative's value will either increase or decrease by 30% of its notional value.

This is of course one of the simplest examples one can think of about Counterparty Risk and Stress Testing. Most of the derivatives in question have more complex and irregular payoffs and the stress scenarios involve more than only one factor being stressed. Therefore, it may take days to receive reliable results of a topical stress testing exercise.
To alleviate at least a portion of the above problem, we can use sensitivity based estimates (assuming we have indeed risk sensitivities available to our disposal for our trading portfolio) where trade level stress results would become available much faster than the typical model based approach. The aim is to use these estimates for swift review and action but also to see if the actual results will reconcile with our estimates.

In order to achieve above results, I was asked i) to prepare an aggregator for counterparty risk metrics that will aggregate all the available sensitivity based estimates and calculate the counterparty risk metrics such as EAD and JTD, ii) to create controls for data quality issues which are inherent to sensitivity based estimates.

For the first issue outlined above, I will be offering an SQL and R based data engineering solution, and in addition to that, I will provide some additional information about the underlying data and put in place some checks and controls to the above data engineering exercise using various statistical/econometric methods to account for the second issue outlined above. Finally, I am planning on using data visualization (or an interactive dashboard) to summarize and communicate some aspects of the data that have been aggregated and transformed with the above data engineering and analysis exercises.

This will be the part where I will be using a handful of fixed reporting templates to communicate the stressed counterparty risk metrics with the senior management on a regular basis. At the end, I am planning to provide actionable outputs in 2-4 hours right after the upstream data regarding the sensitivity based estimates are provided.

The ultimate objective of this project on the other hand is to simplify high-dimensional data and to communicate it with various stakeholders. The final output should include different cuts of financial information, certain statistical summaries and controls along with a library of visualizations. That is why I have decided to create a detailed statistical model which could explain the level of expected collateral shortfalls under a given stress scenario, and how long would a short term fluctuation be expected to return to normal. In order to do that, I have tried my chances with a First Difference Panel Data Estimation and an Error Correction Model which are working in parallel.

Collateral shortfall is a simple counterparty risk metric which shows how much is at risk if a given counterparty defaults after accounting for bankruptcy-remote collateral. Typically, collateral is defined by various legal agreements, and it closely follows the level of MtM values of a given portfolio. There are different types of legal agreements that are relevant but out of this summary's scope. So, in short, to account for unobserved heterogeneity by removing the counterparty dependent factors, I have proposed to use a First Difference Estimator. To see if there is a long run relationship between my variables, I have proposed to use an Error Correction Model.

Additionally, I have prepared a simple visual library using ggplot2 in R to summarize the shocks applied to each product type (e.g. securities lending, listed futures, interest rate swaps and so on) in the above-mentioned data engineering exercise to see the impacts of the shocks themselves and also to see if there are any outliers available that could potentially signal a data quality problem. Since the output of my data engineering project is bound to include some outliers with data quality problems (since I am using sensitivity based estimates), it feels natural to draw them in some plots to see if these issues are easily noticeable. It turned out to be just the case.

In a nutshell, I was able to deliver the data engineering part of the project where I successfully reduced the run time of my code to 80-100 minutes (thanks to data.table and its authors). The results are saved down to a shared folder in csv format showing counterparty metrics in counterparty and netting set levels. I was also able to build the above-mentioned models. Their results are definitely not written in stone and they are only there to provide guidance to the end-users on what they should be expecting from a given position change. Finally, I was also able to create a handful of visual templates which I am using to create either portfolio level or counterparty level summaries showing/translating the prescribed shocks with different levels of granularity and format into an easily understandable percentage points that are showing the shock applied to the most material products available in the Client's trading book.