CQF Final Project:
Pricing of k^{th} to default CDS

Submitted by: Anirbid Ghosh

Date: 4th January, 2021

TABLE OF CONTENTS

1 Introduction to Basket CDS

A single-name CDS is the simplest type of credit derivative and references a single credit entity. A basket CDS, on the other hand, references a group of entities (in this paper, we have restricted ourselves to a basket with 5 entities: Bank of America, Citibank, Goldman Sachs, JP Morgan Chase and Morgan Stanley). While it is theoretically possible to buy protection on all the names in the basket, in practice the contract is typically restricted to the first "k" names. For example:

- First to default: the credit protection kicks in after the first default and the contract is thereafter terminated
- Second to default: the credit protection kicks in after the second default only (and not after the first)
- k^{th} to default: the credit protection kicks in after the k^{th} entity defaults

Basket CDSs provide buyers with a cheaper option as compared to buying a single name CDS on all the reference entities; on the other hand, it provides sellers with a higher premium as compared to selling protection on individual credit entities. An example below illustrates this point – where the first-to-default basket is priced at 185 bps – and provides both the buyer and the seller an economic rationale to engage in the trade.

Reference Entity	Credit Rating	Premium (bps)
Bank	AA/Aa	53
Non-bank financial	A/A2	120
Airline	BBB-/Baa2	130
Utility	BBB+/Baa1	110
Transport	BBB+/Baa2	120

With the introduction of index CDS products such as the iTraxx and CD-X, the volumes of basket CDSs have come down significantly. Currently, they are primarily used for more granular exposure management and hedging, with the first-to-default basket being the most commonly traded instrument.

Illustration of a Basket CDS

This paper revolves around the pricing of a basket with 5 names: Bank of America, Citibank, Goldman Sachs, JP Morgan Chase and Morgan Stanley. The notional amount specified for each name in the basket is \$ 2M; therefore the total value of the portfolio is \$ 10M. For the purpose of this illustration, let us assume it is a 3^{rd} to default basket with a maturity of 5 years and the annual premium is set at 50 bps.

During the first year of the transaction, let's assume one of the entities default. The defaulted name drops out of the basket and the terms of the basket changes: the total notional reduces to \$ 8M covering the 4 remaining entities. In the second year, let us assume another entity defaults – resulting in a further reduction of the portfolio value to \$ 6M covering 3 entities. During the third year, there are no defaults. In the 4^{th} year, another entity defaults leading to a payout of \$ 2M X (1 – 0.4) = \$ 1.2M to the protection buyer (assuming a recovery rate of 40%) and then the contract terminates since the threshold of 3 defaults is reached. To keep this illustration simple, it is being assumed that the defaults occur at the end of year 1, 2 and 4 and the time value of money is ignored.

Year	Cash Flow Paid – Protection Buyer	Cash Flow Paid – Protection Seller
1	\$ 50K	No payment made
2	\$ 40K	No payment made
3	\$ 30K	No payment made
4	\$ 30K	\$ 1.2M
5	Contract terminated	Contract terminated
Total	**\$ 150K**	**\$ 1.2M**

2 Pricing of CDS Baskets: An Overview

The pricing of CDS baskets is predicated upon the no-arbitrage condition that the NPV of premium payments is equal to the NPV of loss conditional upon default. In other words, NPV(spread * premium leg) = NPV(default leg); this implies that spread = NPV (default leg/ premium leg).

Premium Leg

The expected value of the premium leg across the lifecycle of the contract can be written as:

Premium Leg PV = $\sum_{i=1}^{N} \boldsymbol{spread} * \boldsymbol{D(0,\ T_i) * S(T_i) * \Delta t_i}$

Where:

$D(0, T_i)$: Discount rate between 0 and T_i

$S(T_i)$: Probability that the entity survives up to time T_i

Δt_i : Length of the time interval (for the purpose of this paper, Δt is set to 1 year)

Default Leg

Similar to the premium leg, the expected value of the default leg across the lifecycle of the contract can be written as:

Default Leg PV = $\boldsymbol{(1 - R) \sum_{i=1}^{N} D(0,\ T_i)\ (S(T_{i-1}) - S(T_i))}$

Where:

R : Recovery rate associated with the underlying

$D(0, T_i)$: Discount rate between 0 and T_i

$S(T_{i-1})$ - $S(T)$: Probability that the entity defaults within the time interval T_{i-1} and T

Spread Calculation

Equating the premium and default legs, we get:

$$\text{spread} = \frac{\boldsymbol{(1-R) \sum_{i=1}^{N} D(0,T_i)\ (S(T_{i-1}) - S(T_i))}}{\boldsymbol{\sum_{i=1}^{N} D(0,T_i)\ S(T_i)\ \Delta t_i}}$$

This equation can be rewritten in terms of default probabilities as shown below.

$$\text{spread} = \frac{\boldsymbol{(1-R) \sum_{i=1}^{N} D(0,T_i)\ F(T_i)}}{\boldsymbol{\sum_{i=1}^{N} D(0,T_i)\ (1 - F(T_i))\ \Delta t_i}}$$

Now, $F(T) = F(t_1, t_2, t_3, t_4, t_5)$ – i.e., the time to default for the basket is based on the joint distribution of the default times for each of the underlying single names. This joint distribution is typically estimated using Copula methods (with Gaussian and ‘t’ being the two most common choices). It should be noted in this context that this distribution will be different depending on “k” or the number of entities that need to default before credit protection kicks in – in other words, a k^{th} to default basket needs to priced as a different instrument for each “k”.

The above calculation may be simplified by defining the total expected loss, an expectation over the joint distribution. Let E $[F_k(t)]$ = L_k where L_i - L_{i-1} = 1/5 * Notional. Using this identify, the spread equation from above can be written as:

E[spread] = $\frac{(1-R)\sum_{i=1}^{N} D(0,T_i)\,(L_i - L_{i-1})}{\sum_{i=1}^{N} D(0,T_i)\,(Notional - L_i)\,\Delta t_i}$

The above equation can be used to calculate the spread for k^{th} to default swap. For example:

spread (k=1) = $\frac{(1-R)\,D(0,T_1) * 1/5}{D(0,T_1)T_1 * 5/5}$

Spread (k=2) = $\frac{(1-R)\,D(0,T_2) * 1/5}{D(0,T_1)T_1 * \frac{5}{5} + D(0,T_2)(T_2 - T_1) * \frac{4}{5}}$

And so on for k=3, k=4 and k=5. These equations will be used in determining the spreads for the CDS basket (as presented in sections 4, 5 and 6).

3 Mathematical Preliminaries

The pricing problem explored in this paper requires the application of various mathematical tools and techniques. A short exposition of each of these techniques is provided in this section to allow readers to better follow the workings in sections 4, 5 and 6.

3.1 Modeling defaults using Poisson Processes

Defaults are assumed to be governed by Poisson processes. A Poisson process with intensity “λ” is a stochastic process N_t: t >= 0 taking such values in S = {0, 1, 2, …..} such that:

a) $N_0 = 0$

b) $If\ s < t, then\ N_s < N_t$

c) $If\ s < t, then\ the\ increment\ N_t - N_s\ is\ ind. of\ what\ happened\ during\ [0, s]$

d) $Let\ h \rightarrow 0^+, then$

$$\begin{aligned} \Pr(N_{t+m} = n + m \mid N_t = n) &= \lambda h + O(h) && \text{when } m = 1 \\ &= O(h) && \text{when } m = 0 \\ &= 1 - \lambda h + O(h) && \text{when } m > 1 \end{aligned}$$

In the context of pricing credit sensitive instruments, we are interested in the timing of defaults rather than the number of defaults. Therefore, we need to derive the PDF of default times, which is worked out below:

For the purpose of this derivation, let “ T ” be the time to default denoting the first event. Then $T = \inf\{t \mid N_t > 0\}$. Leveraging (d) from above, we get;

$\Pr(N_{t+h} = 1 \mid N_t = 0) = \lambda h + O(h)$

Re-writing the above equation in terms of default time, we get;

$\Pr(T < t + h \mid T > t) = \lambda h + O(h)$

$\Rightarrow \lambda = \lim_{h \to 0^+} \frac{1}{h} \Pr(T < t + h \mid T > t) + \frac{O(h)}{h}$

$\Rightarrow \lambda = \lim_{h \to 0^+} \frac{1}{h} \Pr(T < t + h \mid T > t)$ $[\text{Since } \frac{O(h)}{h} \rightarrow 0]$

$\Rightarrow \lambda = \lim_{h \to 0^+} \frac{1}{h} \left[\frac{\Pr(t < T < t+h)}{\Pr(T > t)}\right]$ [Applying Bayes’ Theorem]

Re-writing the above equation in terms of survival probabilities , we get;

$\Rightarrow$ **λ = lim**$_{h \to 0+} \frac{1}{h} \left[(\mathbf{S}(\mathbf{t}) - \mathbf{S}(\mathbf{t}+\mathbf{h}))/\,\mathbf{S}(\mathbf{t})\right]$

$\Rightarrow$ **λ =** $-\frac{\mathbf{S}'(\mathbf{t})}{\mathbf{S}(\mathbf{t})]}$

Integrating both sides with respect to “t”, we get;

$\Rightarrow -\boldsymbol{\lambda}\mathbf{t} = \mathbf{log}\,\mathbf{S}(\mathbf{t}) - \mathbf{log}\,\mathbf{S}(\mathbf{0})$

$\Rightarrow -\boldsymbol{\lambda}\mathbf{t} = \mathbf{log}\,[\mathbf{S}(\mathbf{t})/\,\mathbf{S}(\mathbf{0})]$

$\Rightarrow -\,\boldsymbol{\lambda}\mathbf{t} = \mathbf{log}\,\mathbf{S}(\mathbf{t})$ since S(0) = 1 by definition

$\Rightarrow$ **S(t)** $= \boldsymbol{exp}\,(-\boldsymbol{\lambda}\mathbf{t})$, which implies that the survival times follow an exponential distribution with parameter “λ”.

When “λ” is assumed to be constant across the entire time period, the process is referred to as a Homogeneous Poisson Process. Alternately, if “λ” can vary with time, the process is called an Inhomogeneous Poisson Process (the treatment in this paper is based on Inhomogeneous Poisson Processes). In addition to constant and deterministic “λ”, the intensity rate may also be modeled as a stochastic random variable (referred to as Cox processes) – please note this is outside the scope of this paper.

3.2 Bootstrapping hazard rates

To be able to price a basket swap, the term structure of hazard rates needs to be derived from market observable CDS quotes. This is done through a bootstrapping process as explained below:

Step 1: Estimating Survival Probabilities

Firstly, the survival probabilities need to be calculated based on the following iterative equation. In other words, we need to first calculate $S(T_1)$; then we calculate $S(T_2)$ using $S(T_1)$. We carry on this process until $S(T_N)$ is calculated.

$$\mathbf{S(T_N)} = \frac{\sum_{i=1}^{N-1} D(0,T_i)\,[\ LS(T_{i-1}) - (L+\Delta t_i spread_N) S(T_i)\ ]}{D(0,T_N)\ X\ (L+\ \Delta t_N spread_N)} + \frac{LS(T_{N-1})}{L+\Delta t_N spread_N}$$

Where:
L : 1 – R (Recovery rate associated with the underlying)
$D(0, T_i)$: Discount rate between 0 and T_i
$S(T_i)$: Probability that the entity survives up to time T_i
$spread(T_N)$: Market observable CDS spread for time T_N

Step 2: Bootstrapping Hazard Rates

Once the survival probabilities across various time periods have been calculated, the corresponding hazard rates can be bootstrapped as shown below:

We know that:

$$\mathbf{S(T_N)} = \exp\left(-\int_0^{T_N} \lambda_s ds\right) = \exp\left(-\sum_{i=1}^{N} \lambda_i \Delta t_i\right)$$

$$\Rightarrow \mathbf{S(T_N)} = \exp\left(-\sum_{i=1}^{N-1} \lambda_i \Delta t_i - \lambda_N \Delta t_N\right)$$

$$\Rightarrow \mathbf{S(T_N)} = \exp\left(-\sum_{i=1}^{N-1} \lambda_i \Delta t_i\right) \exp\left(-\lambda_N \Delta t_N\right)$$

$$\Rightarrow \mathbf{S(T_N)} = \mathrm{S(T_{N-1})} \exp\left(-\lambda_N \Delta t_N\right)$$

$$\Rightarrow -\lambda_N \Delta t_N = \log \mathrm{S(T_N)} - \log \mathrm{S(T_{N-1})}$$

$$\Rightarrow \lambda_N = -\frac{1}{\Delta t}\left[\log \mathrm{S(T_N)} - \log \mathrm{S(T_{N-1})}\right]$$

Step 3: Converting Hazard Rates to Default Times

From a simulation, we get correlated random numbers that are uniformly distributed; i.e., (u_1, u_2, u_3, u_4, u_5). Using the $\boldsymbol{\lambda_i's}$ computed in the previous step, we need to convert the u_i's to T_i's. We know that:

$\mathbf{F}(\mathrm{T}) = \mathbf{u} = \mathbf{1} - \mathbf{exp(}\text{-}\,\boldsymbol{\lambda t)}$ since default times are exponentially distributed

$\Rightarrow \mathbf{log}(\mathbf{1}-\mathbf{u}) = -\,\boldsymbol{\lambda}_{\mathrm{T}}\mathrm{T}$

Given that the above expression contains 2 unknowns, $\boldsymbol{\lambda}_{\mathrm{T}}$ and T, we need to solve this iteratively by adding up the hazard rates such that:

$\mathrm{T} = \mathbf{inf}\,\{\,\mathbf{t} > \mathbf{0} : \mathbf{log}(\mathbf{1}-\mathbf{u}) \geq -\sum_{t}\boldsymbol{\lambda}_{\mathbf{m}}\,\}$ where default occurs if inequality holds and $t_{m-1} \leq \mathrm{T} \leq t_m$. In other words, based on the above relationship, we can determine the default time as $\mathrm{T} = \mathbf{t_{m-1}} + \boldsymbol{\delta t}$ where $\boldsymbol{\delta t}$ represents the year fraction. There are 2 ways to calculate this year fraction:

- Option I: Assume $\boldsymbol{\delta t}$ **= 0.5**. This is known as the accrual method and has been implemented as part of this paper
- Option II: Calculate the exact year fraction based on the following relationship:

$$\mathbf{1}-\mathbf{u} = \mathbf{exp}\,(\,\int_{\mathbf{0}}^{t_{m-1}+\delta t}\boldsymbol{\lambda}_{\mathbf{s}}\,\boldsymbol{ds}\,)$$

$$= \mathbf{P}(\mathbf{0}, \boldsymbol{t_{m-1}})\,\mathbf{exp}\,(\,\int_{t_{m-1}}^{t_{m-1}+\delta t}\boldsymbol{\lambda}_{\mathbf{s}}\,\boldsymbol{ds}\,)$$

$$\Rightarrow \mathbf{log}(\frac{\mathbf{1-u}}{\mathbf{P}(\mathbf{0},\boldsymbol{t_{m-1}})}) = -\,\boldsymbol{\delta t\lambda}_{\mathbf{m}}$$

$$\Rightarrow \boldsymbol{\delta t} = -\frac{\mathbf{1}}{\boldsymbol{\lambda}_{\mathbf{m}}}\mathbf{log}(\frac{\mathbf{1-u}}{\mathbf{P}(\mathbf{0},\boldsymbol{t_{m-1}})})$$

Once the default times are estimated as outlined above, they can be substituted into the spread calculation formula and spreads estimated for various k's (as explained in section 2).

3.3 Copulas & Joint Distributions

3.3.1 What is a Copula?

A copula is a function that takes in univariate marginal distributions as inputs and generates the corresponding multivariate joint distribution as the output. In other words, for "k" uniformly distributed random variables, the joint distribution function $C(U_1, U_2, U_3, \ldots; \sum)$ is called a copula. In this context, U_i represents the marginal distributions and $\sum$ represents the covariance matrix.

Since the CDF of any random variable is uniformly distributed (i.e., $U_i = F_i(X_i)$), the copula function can be used to link marginal distributions to a joint distribution. Suppose there are "k" random variables $X_1, X_2, X_3, \ldots X_k$ and their marginal distributions are represented by $F_1(X_1)$, $F_2(X_2)$, $F_3(X_3), \ldots F_k(X_k)$, then

$$\mathbf{F(X_1, X_2, X_3, \ldots) = C(F_1(X_1), F_2(X_2), F_2(X_2) \ldots; \textstyle\sum)}$$

Sklar's theorem proves the converse of the above: i.e. any joint distribution can be written in the form of a copula function and if the joint distribution is continuous, then the copula function is unique. In other words, for any joint distribution, the marginal distribution and the dependence structure can be isolated, with the later completely described by the copula.

3.3.2 Modeling Joint Default Times Using a Copula Function

Let τ_1, τ_2, τ_3…represent random variables modeling the default time of credit sensitive instruments. We need to derive an expression representing the joint distribution of such random variables.

$$\mathbf{F(\tau_1, \tau_2, \tau_3 \ldots) = \mathbb{P}(\tau_1 \leq t_1, \tau_2 \leq t_2, \tau_3 \leq t_3 \ldots \ldots)}$$

$$\Rightarrow \mathbf{F(\tau_1, \tau_2, \tau_3 \ldots) = \mathbb{P}(F_1^{-1}(U_1) \leq t_1, F_2^{-1}(U_2) \leq t_2, F_3^{-1}(U_3) \leq t_3, \ldots \ldots) \text{ since } U_i = F_i(\tau_i)}$$

$$\Rightarrow \mathbf{F(\tau_1, \tau_2, \tau_3 \ldots) = \mathbb{P}(U_1 \leq u_1, U_2 \leq u_2, U_3 \leq u_3, \ldots \ldots)}$$

$$\Rightarrow \mathbf{F(\tau_1, \tau_2, \tau_3 \ldots) = = C(U_1, U_2, U_3, \ldots; \textstyle\sum)}$$

The above derivation provides the mathematical rationale and justification for using a copula to model the joint distribution of default times. In the subsequent sub-sections, we briefly discuss two specific copula functions (the "gaussian" copula and the "t" copula) which are commonly used to model the joint distribution.

3.3.3 Modeling the joint distribution using the "Gaussian" copula

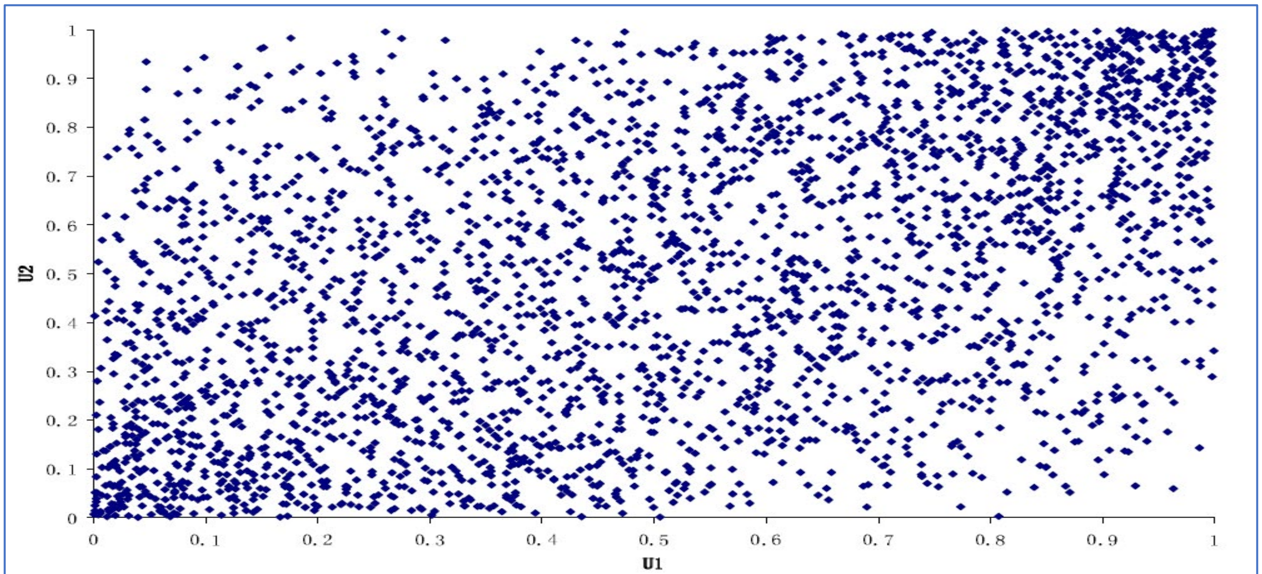
Let Φ_n be the normalized multivariate standard normal distribution function and Φ be the univariate standard normal distribution function. Then the multivariate Gaussian Copula function is defined as:

$$\mathbf{C}(u_1, u_2, u_3, \ldots\ldots) = \Phi_n\ (\Phi^{-1}(u_1), \Phi^{-1}(u_2), \Phi^{-1}(u_3), \ldots\ldots; \Sigma)$$

And the corresponding density function is given by (in terms of a vector of uniform random variables **U**):

$$\mathbf{C}(u_1, u_2, u_3, \ldots\ldots) = \frac{\mathbf{1}}{\sqrt{|\boldsymbol{\Sigma}|}}\ \boldsymbol{exp}\ [\frac{\mathbf{1}}{\mathbf{2}}\Phi^{-1}(\mathbf{U}')(\Sigma^{-1} - \mathbf{I})\Phi^{-1}(\mathbf{U})]$$

An example: Simulated Bivariate Gaussian Copula with ρ = 0.5:



Source: CQF Class Notes – Dr. Siyi Zhou

3.3.4 Modeling the joint distribution using the "t" copula

Let T_v be the normalized multivariate Student's "t" distribution function with "v" degrees of freedom, and t_v be the normalized univariate "t" distribution function also with "v" degrees of freedom, then the multivariate Student's "t" Copula is given by:

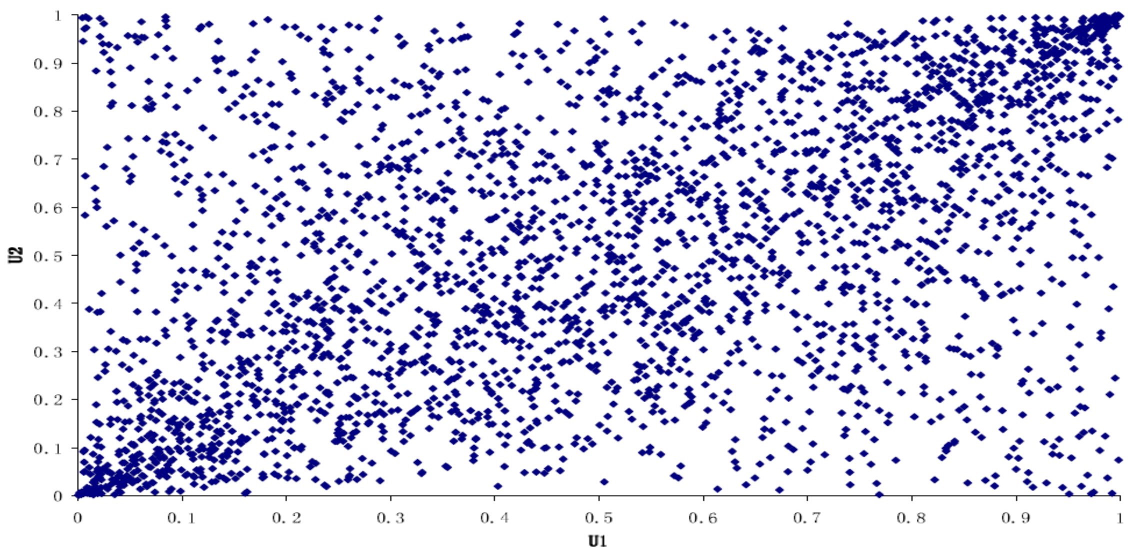
$$\mathbf{C}(u_1, u_2, u_3, \ldots\ldots) = T_v\ (t_v^{-1}(u_1), t_v^{-1}(u_2), t_v^{-1}(u_3), \ldots\ldots; \Sigma)$$

And the corresponding density function is given by (in terms of a vector of uniform random variables **U**):

$$\mathbf{C}(u_1, u_2, u_3, \ldots\ldots) = \frac{\mathbf{1}}{\sqrt{|\boldsymbol{\Sigma}|}} \frac{\Gamma\left(\frac{v+n}{2}\right)}{\Gamma\left(\frac{v}{2}\right)} \left(\frac{\Gamma\left(\frac{v}{2}\right)}{\Gamma\left(\frac{v+1}{2}\right)}\right)^{n} \frac{\left(1+\frac{T_v^{-1}\left(U^{-1}\right)\Sigma\, T_v^{-1}(U)}{v}\right)^{-\frac{v+n}{2}}}{\prod_{i=1}^{n}\left(1+\frac{T_v^{-1}(u_i)^2}{v}\right)^{-\frac{v+1}{2}}}$$

Where $\Gamma(\boldsymbol{v})$ is the gamma function.

An example: Simulated Student's "t" Copula with ρ = 0.5 and v = 2:



Source: CQF Class Notes – Dr. Siyi Zhou

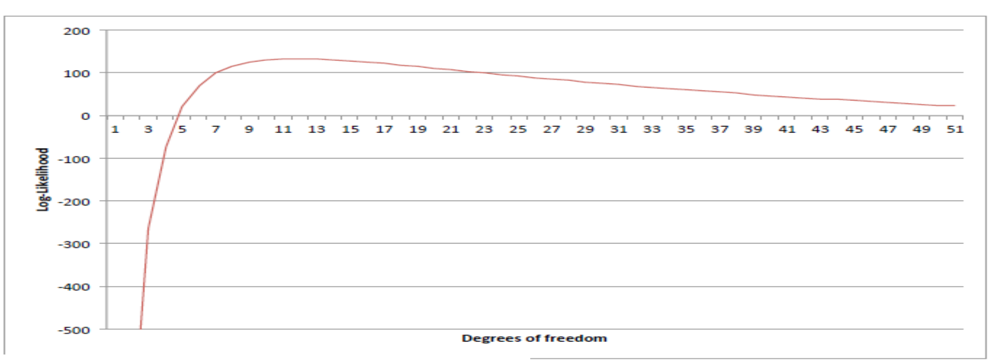
As can be seen from the graph above, the "t" Copula exhibits fatter tails while retaining the symmetric nature of the Gaussian Copula. Since financial variables typically exhibit more extreme values than predicted by the normal distribution, this Copula is often the tool of choice for modeling the dependence structure (i.e., within the family of elliptical distributions).

"t" Copula: Estimating the degrees of freedom

The "degrees of freedom" parameter in a "t" Copula needs to be estimated using MLE. We essentially need to maximize the following expression:

$argmax \{ \sum_{t=1}^{T} \log C \, (\boldsymbol{U_t},; v, \ \Sigma)$ where $\boldsymbol{U_t}$ is a row vector of observations for the 5 reference entities.

An illustration of the MLE process is shown for v = 1, 2, 3, ….50. It can be seen from the pot below that the maximum is reached when v = 11.



Source: CQF Project Notes – Dr. Richard Diamond

3.4 Low Discrepancy (Sobol) Sequences

Regular Monte Carlo simulation employs sequences of pseudo-random numbers which are constrained by slow convergence. A faster convergence is achieved by using low-discrepancy sequence, also called quasi-random sequences, which fill the space of possibilities more quickly and evenly.

There are several variations of low-discrepancy sequences such as van der Corput sequence, Halton sequence, Sobol's sequence etc. We describe generation of Sobol sequence using Gray code implementation. In addition to pseudo-random numbers, Sobol sequences are used for simulating Copulas as part of this paper.

To generate the j^{th} component of the points in a Sobol sequence, we choose a polynomial of degree s_j:

$$\boldsymbol{x^{s_j} + a_{1,j}x^{s_j-1} + a_{2,j}x^{s_j-1} + \cdots + a_{s_j-1,j}x + 1},$$

where $a_{i,j}$ are either 0 or 1. We define a sequence of positive integers, $m_{k,j}$, observing a recurring relationship:

$$\boldsymbol{m_{k,j} := 2a_{1,j}m_{k-1,j}\,(*)\,2^2a_{2,j}m_{k-2,j}\,(*)\ldots(*)\,2^{s_j-1}a_{s_j-1,j}m_{k-s_j+1,j}\,(*)\,2^{s_j}m_{k-s_j,j}\,(*)\,m_{k-s_j,j}}\ ,$$

where $(*)$ is the bit-wise exclusive-or operator and $1 \le k \le s_j$. Each $m_{k,j}$ may be assigned odd integer values less than 2^k. The direction number $v_{k,j}$ may be defined as

$$\boldsymbol{v_{k,j} := \frac{m_{k,j}}{2^k}}$$

Then $x_{i,j}$, the j^{th} component of the i^{th} point in a Sobol sequence can be generated recursively by

$$\boldsymbol{\bar{x}_{i,j} := \bar{x}_{i-1,j}\,(*)\,v_{c_{i-1},j}} \quad \text{and} \qquad \boldsymbol{\bar{x}_{0,j} := 0,}$$

where c_i is the index of the first 0 bit from the right in the binary equivalent of i

Illustration:

Starting with $s_j = 4$, $a_{1,j} = 1$, $a_{2,j} = 0$, and $a_{3,j} = 0$, the primitive polynomial is $x^4 + x^3 + 1$.

Assuming $m_{1,j} = 1, m_{2,j} = 3,\ m_{3,j} = 1,\ and\ m_{4,j} = 3$, we can compute

$$m_{5,j} = 22, m_{6,j} = 31, m_{7,j} = 47 \ldots\ldots$$

The direction numbers can be computed as

$v_{1,j} = (0.1)_2, v_{2,j} = (0.11)_2, v_{3,j} = (0.001)_2, v_{4,j} = (0.0011)_2,$

$v_{5,j} = (0.10110)_2, v_{6,j} = (0.011111)_2,\ v_{7,j} = (0.0101111)_2 \ldots.$

Hence, the j^{th} component of the first few points in the Sobol sequence may be computed as:

i	$(i)_2$	c_{i-1}	$x_{i-1,j}\ (*)\ v_{c_{i-1},j}$	$x_{i,j}$
0	$(000)_2$	–	–	0
1	$(001)_2$	1	$(0)_2\ (*)\ (0.1)_2 = (0.1)_2$	0.5
2	$(010)_2$	2	$(0.1)_2\ (*)\ (0.11)_2 = (0.01)_2$	0.25
3	$(011)_2$	1	$(0.01)_2\ (*)\ (0.1)_2 = (0.11)_2$	0.75
4	$(100)_2$	3	$(0.11)_2\ (*)\ (0.001)_2 = (0.111)_2$	0.875
5	$(101)_2$	1	$(0.111)_2\ (*)\ (0.1)_2 = (0.011)_2$	0.375
6	$(110)_2$	2	$(0.011)_2\ (*)\ (0.11)_2 = (0.101)_2$	0.625
7	$(111)_2$	1	$(0.101)_2\ (*)\ (0.1)_2 = (0.001)_2$	0.125

3.5 A Short Note on Correlations

Correlations provide an important input to the pricing process and enable the parameterization of Copulas that are necessary for estimating the joint distribution of default. For the Gaussian Copulas, Pearson's correlation may be used. However, the "t" Copula requires the use of rank correlation measures such as the Spearman rank correlation or the Kendall's Tau (this paper relies on Kendall's Tau for parameterizing the "t" Copula).

Kendall's Tau: A Brief Overview

Let (x_i, y_i) be a set of observations of the joint random variables X and Y. Any pair of observations (x_i, y_i) and (x_j, y_j) where $i < j$ are said to concordant if the sort order is maintained – i.e., either $x_i < x_j$ and $y_i < y_j$ holds or $x_i > x_j$ and $y_i > y_j$ holds; otherwise the pair is said to be discordant. Given this definition, Kendall's Tau is defined as:

$$\mathrm{T} = \frac{\textit{number of concordant pairs} - \textit{number of discordant pairs}}{\binom{n}{2}}$$

This identity can also be written in an equivalent explicit form:

$$\mathrm{T} = 2 * \frac{\sum_{i<j} sign(x_i - x_j) sign(y_i - y_j)}{n(n-1)}$$

The denominator represents the total number of possible combinations, hence the coefficient is bounded between (-1, +1). The boundary conditions include:

- If the agreement between the two rankings is perfect, the value of the coefficient is +1
- If the disagreement between the two rankings is perfect, the value of the coefficient is -1
- If the two rankings are independent, the value of the coefficient is close to 0

The rank correlation can then be converted into a linear correlation by applying the following transformation: $\rho = \mathrm{Sin}(\frac{\Pi}{2}\mathrm{T})$. Note that this transformation will need to be applied to all elements in the correlation matrix $\sum$; this linearized correlation matrix is subsequently used to correlate the independent random variables and generate the joint distribution of defaults.

4 Pricing Under the Gaussian Copula

4.1 Model Inputs

In this chapter, the pricing algorithm is implemented using the Gaussian Copula. Key inputs to the pricing algorithm are as follows:

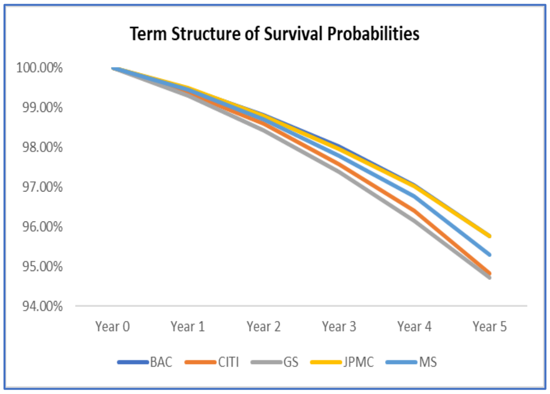
1. Survival probabilities and hazard rates of the underlying reference entities. Market observable CDS spreads are used to calculate the survival probabilities and hazard rates (further details provided below)
2. Correlations across reference entities which are used for parameterizing the Copula (further details provided below)
3. Recovery Rate (assumed to be constant at 40%)
4. Discount factors (the US Treasury curve as of 2^{nd} Oct, 2020 has been used for this purpose)

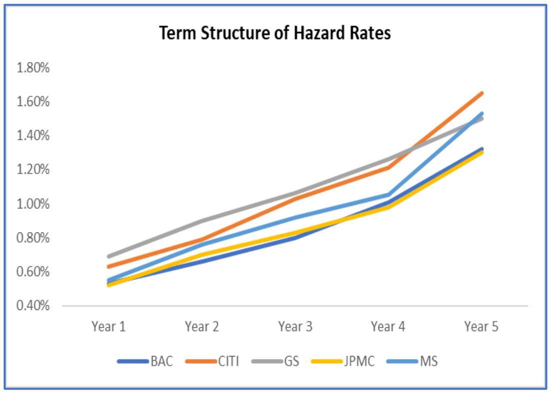
Estimating Survival Probabilities & Hazard Rates

For estimating survival probabilities and hazard rates, the CDS spreads available as of 2^{nd} October 2020 was considered (sourced from Bloomberg). The spreads are presented in the table below:

Spread in bps (as of 2^{nd} October, 2020)					
Entity	**Year 1**	**Year 2**	**Year 3**	**Year 4**	**Year 5**
BAC	31.8	35.85	39.92	45.07	51.86
Citi	37.92	42.81	49.14	54.96	63.65
GS	41.77	47.96	53.3	58.93	65.06
JPMC	31.21	36.79	41.16	45.52	51.97
MS	33	39.36	44.65	49.26	57.61

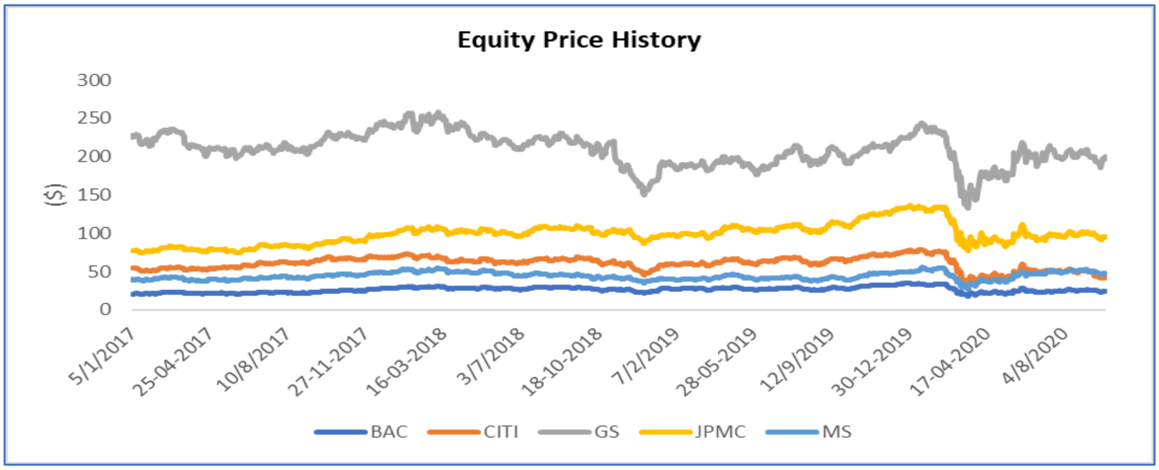
Using these spreads and the formulae described in Section 3.2 (“Bootstrapping Hazard Rates”), the survival probabilities and the corresponding piece-wise constant hazard rates were computed. The outputs are summarized in the charts below (note the monotonically increasing nature of hazard rates) :



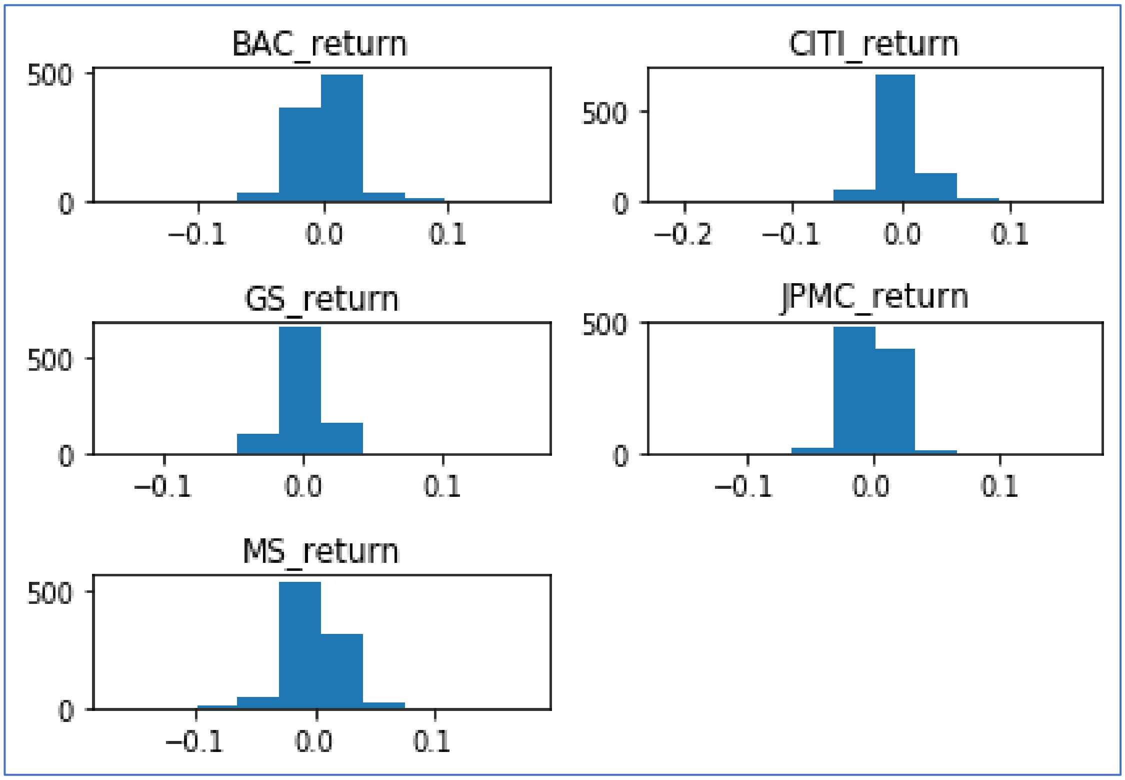


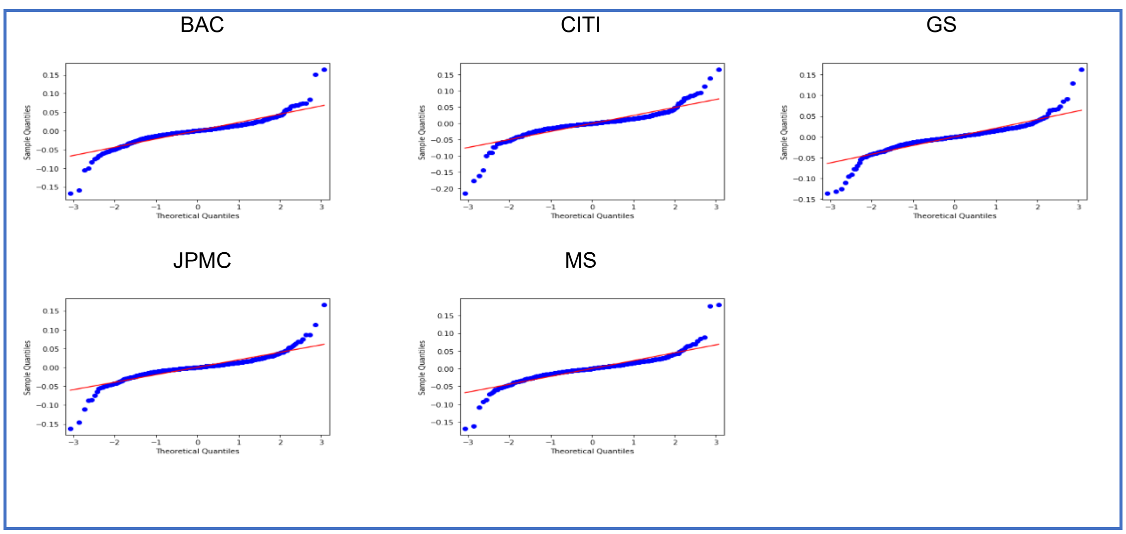
Correlations Between Reference Entities

Equity returns were used as a proxy to calculate correlations based on historical data (between 4th January 2017 – 30th September 2020). A visual analysis of the price time series confirms strong correlation across the entities.

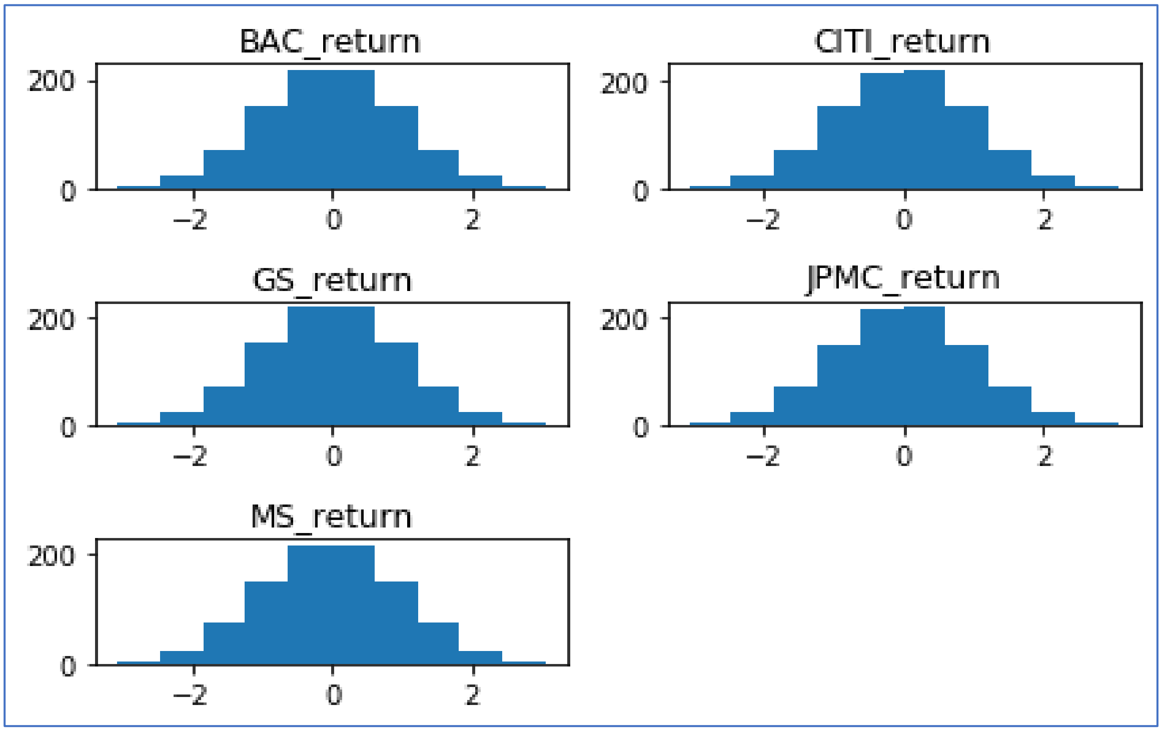


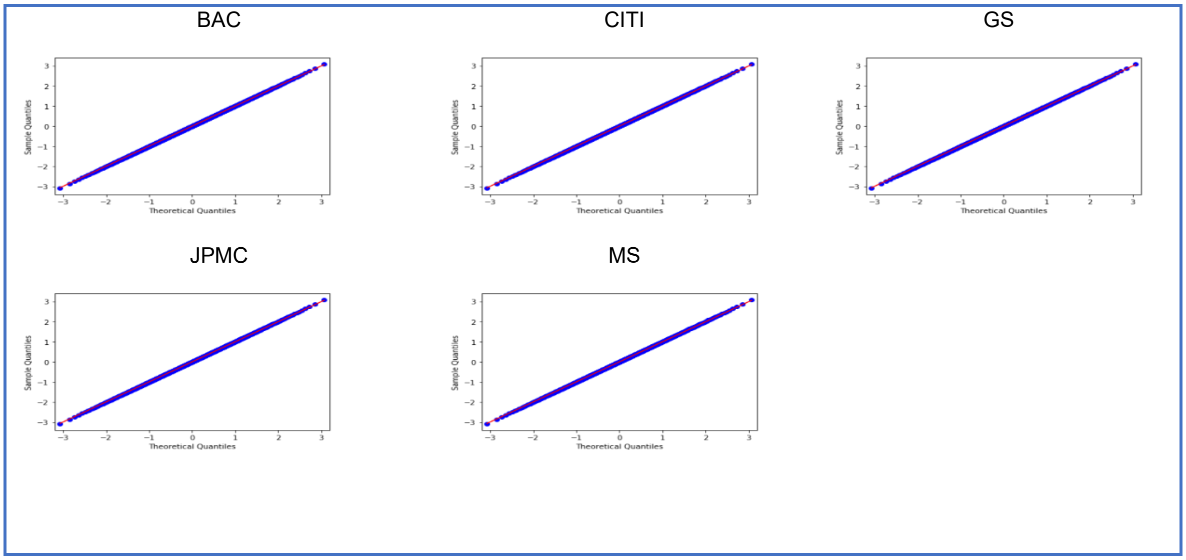
Before calculating Pearson's correlation coefficient, the data was checked for normality using histograms and Q-Q plots (a pre-condition for Pearson's correlation). The data was found to deviate significantly from Gaussian which was further validated using the Shapiro-Wilks test. The histograms and Q-Q plots are provided below.





In view of the above results, it was determined that the data needed to be suitably transformed into normal before Pearson's correlation could be calculated. In order to transform the data, empirical CDFs were obtained (using the ECDF function in Python). Subsequently, the return data was converted to uniform using empirical CDFs; the uniformly distributed returns were then converted into normal. Post this transformation, the histograms and Q-Q plots confirmed that the returns were indeed normally distributed (provided below); the Shapiro-Wilks test was also performed to confirm this assertion.





After transforming the data to normal, Pearson's correlations were calculated. The results are presented in the table below. It may be noted that all pair-wise correlations are extremely high and range between 81% and 92% - this is expected given they are all large US banks with similar business models, risks and opportunities.

	BAC	*CITI*	*GS*	*JPMC*	*MS*
BAC	100.00%	87.41%	83.00%	91.75%	84.92%
CITI	87.41%	100.00%	81.42%	87.84%	82.80%
GS	83.00%	81.42%	100.00%	83.58%	86.08%
JPMC	91.75%	87.84%	83.58%	100.00%	86.36%
MS	84.92%	82.80%	86.08%	86.36%	100.00%

Discount Curve

As mentioned above, the UST curve as of 2nd Oct 2020 was used for discounting purposes. The specific discount factors by tenor are provided below. Please note that 4Y was not available and was derived using linear interpolation.

1Y	2Y	3Y	4Y	5Y
0.12	0.13	0.16	0.22	0.28

A polynomial curve was also fitted to this data (of the form: $y = 0.0093x^2 - 0.0147x + 0.124$) to enable calculation of discount factors across all possible tenors. These discount factors are subsequently used in the simulations for calculating the NPV of both the premium and the default legs.

4.2 Pricing Algorithm

The pricing algorithm uses the following key steps:

1. From the market observable CDS spreads, compute the corresponding survival probabilities and hazard rates
2. Estimate the correlation matrix from historical equity returns (after appropriately transforming the data to normal)
3. Decompose the correlation matrix using the Cholesky method (i.e., correlation matrix = ***A.A'***). In case the correlation matrix is not positive definite, perturb the matrix and derive the closest matrix that is positive definite
4. Estimate the joint distribution of defaults
    a. Generate a set of 100,000 standard normal random numbers – either using the Pseudo random number generator or using Sobol sequences ($\boldsymbol{Z}$)
    b. Convert the set of independently distributed random numbers to a set of correlated random numbers (i.e., $\boldsymbol{Z_c} = \boldsymbol{A.\ Z}$)
    c. Convert the correlated random numbers to uniform random numbers (i.e., $\boldsymbol{U_c = \Phi}$ $(\boldsymbol{Z_c})$)
    d. Convert U_c to default times using the relationship: $\boldsymbol{T_i = F_i^{-1}\ (U_{ci})}$
    e. If the default time > 5 years, populate a dummy value "9999" to indicate a "non-default". If the default time is < 0.25 years, floor default time to 0.25 years to ensure algorithm stability
5. For each "k" (k = 1 to 5), flag the basket as "defaulted" or "not defaulted" depending on the number of defaults within each simulation run
6. For each "k", calculate the default and premium payout within each simulation run (Based on the formula described in Section 2: "Pricing of CDS Baskets: An Overview")
7. Calculate the average default and premium payout across all simulation runs. Calculate spread as: spread = (sum of default payouts/ sum of premium payouts)*10000

4.3 Numerical Results

Using the algorithm mentioned above, the spreads calculated for the "k^{th}" to default baskets are provided below (k = 100,000):

	K=1	K=2	K=3	K=4	K=5
Pseudo random numbers	24.33	13.93	9.07	5.86	3.30
Sobol numbers	24.96	14.18	9.19	5.88	3.25

There are 3 observations that are worth noting:

I. The spread for a basket CDS (for k =1) is less than the spread for single-name CDSs. While this may seem like an anomaly, such dynamics may be expected when the reference entities are very highly correlated (as in this particular case). With very high correlations, the probability of simultaneous defaults increases significantly, thereby increasing the spreads of baskets with higher k's and correspondingly reducing the spreads of baskets with lower k's

II. The spreads are monotonically decreasing as expected. Violation of this condition would point to model deficiencies since by definition a default for a higher "k" would imply a default for a lower "k" as well

III. The spreads calculated using the pseudo-random numbers and the Sobol sequences are similar, across all k's. Except for k = 5, the spreads estimated using the Sobol sequences are marginally higher than those estimated using pseudo-random numbers

Rand vs Sobol Convergence:

As expected, simulations using Sobol sequences converged at a much faster pace as compared to pseudo-random numbers (except for k=5; even in this case, the unexpected behavior could be attributed to one outlier). The spreads, along with the range and standard deviation, are presented below for reference:

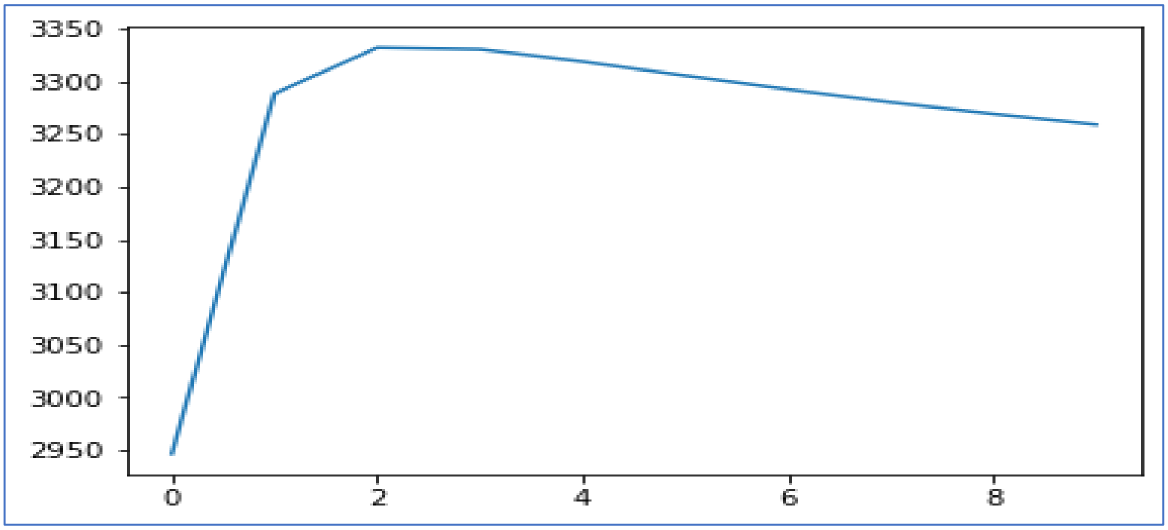
k=1	n = 1K	n = 5K	n = 10K	n = 25K	n = 50K	n = 75K	n = 100K	Range	SD
Pseudo-random	26.76	22.67	23.26	24.06	24.12	24.41	24.33	4.09	1.19
Sobol	25.61	24.74	24.86	24.98	24.98	24.98	24.96	0.87	0.26
k=2	n = 1K	n = 5K	n = 10K	n = 25K	n = 50K	n = 75K	n = 100K	Range	SD
Pseudo-random	15.46	12.62	12.99	13.63	13.73	13.95	13.93	2.85	0.84
Sobol	14.79	14.39	14.34	14.19	14.20	14.11	14.18	0.68	0.22
k=3	n = 1K	n = 5K	n = 10K	n = 25K	n = 50K	n = 75K	n = 100K	Range	SD
Pseudo-random	9.31	8.33	8.55	8.97	8.96	9.14	9.07	0.98	0.32
Sobol	9.84	9.45	9.06	9.15	9.19	9.14	9.19	0.77	0.25
k=4	n = 1K	n = 5K	n = 10K	n = 25K	n = 50K	n = 75K	n = 100K	Range	SD
Pseudo-random	5.01	5.48	5.54	5.83	5.81	5.97	5.86	0.96	0.31
Sobol	6.42	6.00	5.97	5.96	5.96	5.93	5.88	0.54	0.17
k=5	n = 1K	n = 5K	n = 10K	n = 25K	n = 50K	n = 75K	n = 100K	Range	SD
Pseudo-random	3.07	3.22	3.10	3.14	3.24	3.33	3.30	0.26	0.09
Sobol	2.60	3.31	3.24	3.26	3.23	3.25	3.25	0.71	0.23

The spread data across various simulation runs (from number of sims = 1000 to 100,000) is also presented graphically below. The charts conclusively prove that Sobol convergence is faster and spreads typically stabilize after 10,000 runs. Given the computational efficiency of Sobol sequences and faster convergence, all subsequent simulations in this paper have relied solely on low-discrepancy sampling.

5 Pricing Under the "t" Copula

5.1 Model Inputs

In addition to the inputs required for the Gaussian Copula, the "t" Copula requires an extra parameter representing the "degrees of freedom" of the underlying multivariate distribution. As explained in Section 3.3 ("Copula and Joint Distributions"), degrees of freedom is estimated using MLE based on historical equity returns. The chart below shows that the likelihood is maximized when degrees of freedom = 3 (please note that the log-likelihood was executed in a loop with the initial counter set at 0; hence all X-axis values need to shifted by 1).



Another key requirement for the "t" Copula is the usage of rank correlations while parameterizing the Copula. Kendall's Tau has been used for estimating rank correlations – the results are presented in the table below. It may be noted that the correlations have been appropriately linearized before being used to specify the dependence structure between the marginals (i.e., $\rho = \mathrm{Sin}(\frac{\Pi}{2}\mathrm{T})$).

Kendall's Tau	*BAC*	*CITI*	*GS*	*JPMC*	*MS*
BAC	100.00%	88.50%	83.02%	91.65%	85.91%
CITI	88.50%	100.00%	82.34%	88.57%	84.72%
GS	83.02%	82.34%	100.00%	83.57%	87.09%
JPMC	91.65%	88.57%	83.57%	100.00%	87.12%
MS	85.91%	84.72%	87.09%	87.12%	100.00%

5.2 Pricing Algorithm

While the pricing algorithm is broadly similar to that described in section 4.2, there are a few key differences to reflect the generation of the joint distribution of defaults under the "t" Copula. The specific steps are listed below:

The pricing algorithm uses the following key steps:

I. From the market observable CDS spreads, compute the corresponding survival probabilities and hazard rates

II. Estimate the correlation matrix (Kendall's Tau) from historical equity returns. Linearize the correlations using the relationship: $\rho = \mathrm{Sin}(\frac{\Pi}{2}\mathrm{T})$

III. Decompose the correlation matrix using the Cholesky method (i.e., correlation matrix = ***A.A'***). In case the correlation matrix is not positive definite, perturb the matrix and derive the closest matrix that is positive definite

IV. Estimate the joint distribution of defaults

   a. Generate a set of 100,000 standard normal random numbers – either using the Pseudo random number generator or using Sobol sequences (***Z***)

   b. Draw an independent chi- square random variable $s \sim \chi_v^2$. Prior to the drawings, estimate the appropriate degree of freedom using MLE

   c. Compute the corresponding "t" vector $\mathrm{Y} = \mathrm{Z} / \sqrt{\frac{s}{v}}$

   d. Convert the set of independently distributed random numbers to a set of correlated random numbers (i.e., $\boldsymbol{Z_c} = \boldsymbol{A.\ Y}$)

   e. Convert the correlated random numbers to uniform random numbers (i.e., $\boldsymbol{U_c = T_v}$ $(\boldsymbol{Z_c})$)

   f. Convert U_c to default times using the relationship: $\boldsymbol{T_i = F_i^{-1}\ (U_{ci})}$

   g. If the default time > 5 years, populate a dummy value "9999" to indicate a "non-default". If the default time is < 0.25 years, floor default time to 0.25 years to ensure algorithm stability

V. For each "k" (k = 1 to 5), flag the basket as "defaulted" or "not defaulted" depending on the number of defaults within each simulation run

VI. For each "k", calculate the default and premium payout within each simulation run (Based on the formula described in Section 2: "Pricing of CDS Baskets: An Overview")

VII. Calculate the average default and premium payout across all simulation runs. Calculate spread as: spread = (sum of default payouts/ sum of premium payouts)*10000

5.3 Numerical Results

Using the algorithm mentioned above, the spreads calculated for the "k^{th}" to default baskets are provided below (k = 100,000):

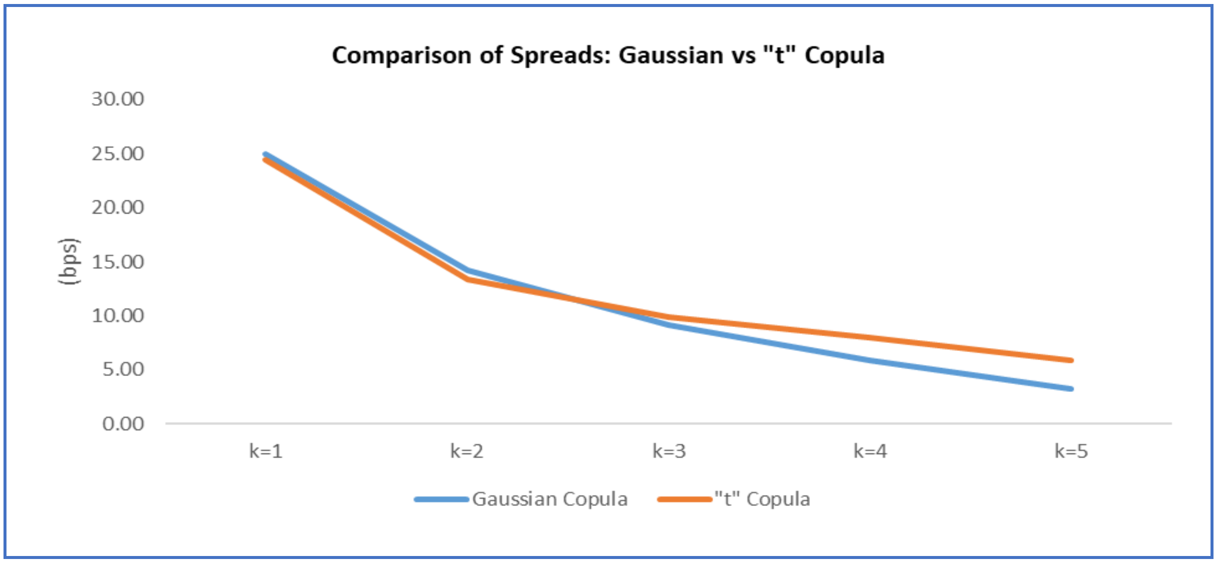
	K=1	K=2	K=3	K=4	K=5
"t" Copula	24.50	13.37	9.93	7.95	5.93

Comparison with the Gaussian Copula

The spreads under the "t" Copula were found to be similar for the lower k's (k=1, k=2, k=3) but materially different for the higher k's (k=4, k=5). For k=1 and k=2, the spreads under the Gaussian Copula were marginally higher; for k=3, the converse was true. For k=4 and k=5, spreads under the "t" Copula were 35% and 83% higher than the corresponding spreads under the Gaussian Copula. This is expected given that the "t" distribution has fatter tails, which would lead to an increased number of joint defaults.

Given these observations and the fact that most financial variables exhibit much fatter tails than those characterized by the normal distribution, market practitioners should be wary of pricing correlation products for higher k's using the Gaussian Copula. In fact, financial regulators have proposed that Gaussian Copulas should not be used (while this comment was in the context of operational risk modeling (EBA/CP/2014/08), the comment may apply to modeling of any joint distribution which exhibits fatter than normal tails).

The chart below summarizes the comparison of spreads between the Gaussian and the "t" Copula for k=1 to 5.



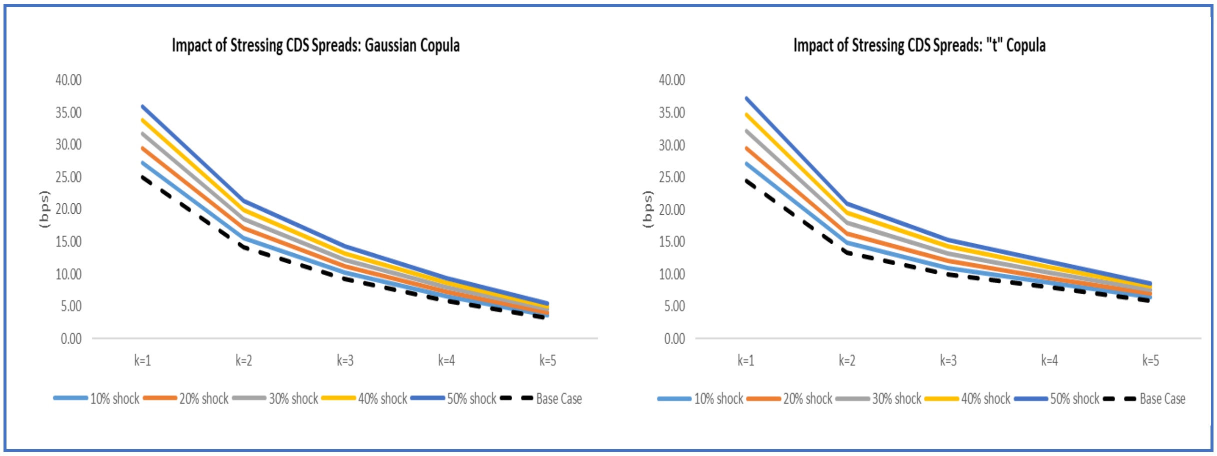
6 Sensitivity Analysis

As part of sensitivity analysis, the following 3 key inputs to the pricing process were subjected to single factor shocks. The changes were then analyzed to verify that they were consistent with economic rationale and market observed dynamics. Note that this analysis was performed both under the Gaussian and "t" Copulas.

1. CDS spreads of the reference entities
2. Recovery rates
3. Correlations

6.1 Sensitivity to CDS Spreads

All single-name spreads were subjected to 5 shocks: 10%, 20%, 30%, 40% and 50%. The results are presented in the charts below. For all k's, the spreads increased monotonically as the shock size increased. Predictably, the absolute changes were highest for k=1 and lowest for k=5. For the Gaussian Copula, the basket spread increased by 10.95 bps for a 50% increase in the underlying single-name spreads; for the "t" Copula, the corresponding increase was 12.80 bps (for k=1).



In addition to analyzing absolute shifts in spreads by Copula type, relative shifts (measured as % changes against the base scenario) across the two Copulas were also assessed. With respect to % changes, it was noticed that for lower k's (k=1 and k=2), the percentage changes were higher under the "t" Copula; for higher k's (k=3 and k=4), the trend was reversed and the percentage changes were higher under the Gaussian Copula. This pattern is clearly evident in the charts below:


Relative Changes in Spreads: Gaussian vs "t" (for k=1)
14.00%
12.00%
10.00%
8.00%
6.00%
4.00%
2.00%
0.00%
10% shock
20% shock
30% shock
40% shock
40% shock
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=2)
30.00%
25.00%
20.00%
15.00%
10.00%
5.00%
0.00%
10% shock
20% shock
30% shock
40% shock
40% shock
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=3)
45.00%
40.00%
35.00%
30.00%
25.00%
20.00%
15.00%
10.00%
5.00%
0.00%
10% shock
20% shock
30% shock
40% shock
40% shock
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=4)
60.00%
50.00%
40.00%
30.00%
20.00%
10.00%
0.00%
10% shock
20% shock
30% shock
40% shock
40% shock
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=5)
80.00%
70.00%
60.00%
50.00%
40.00%
30.00%
20.00%
10.00%
0.00%
10% shock
20% shock
30% shock
40% shock
40% shock
Gaussian Copula
"t" Copula

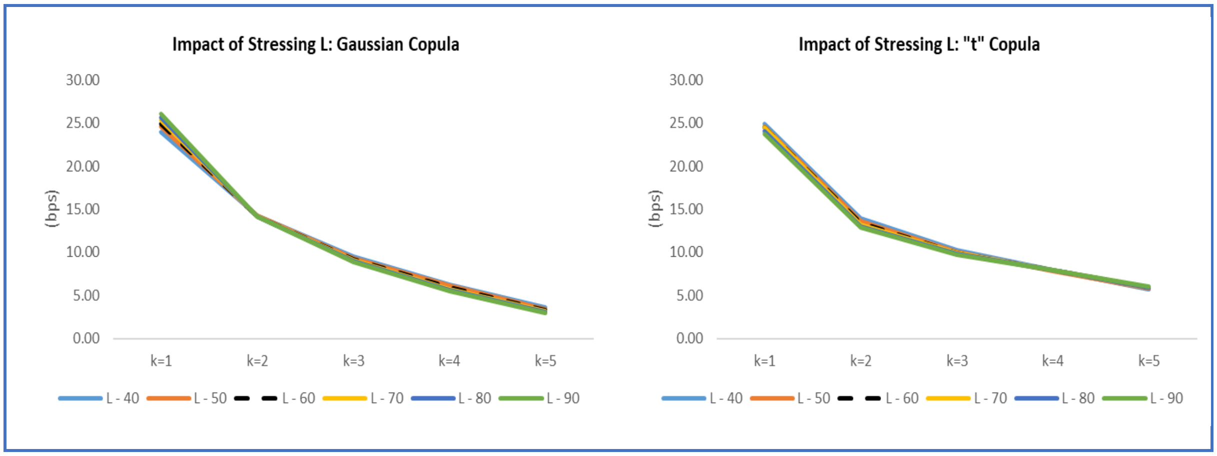
6.2 Sensitivity to Recovery Rates

The base recovery rate used for the pricing model was 40%, implying L = 60% (L = 1 – R). 2 negative and 3 positive shocks were applied to L: -20%, -10%, +10%, +20% and +30%. Therefore, the stressed values for L considered for sensitivity analysis included: 40%, 50%, 70%, 80% and 90%.

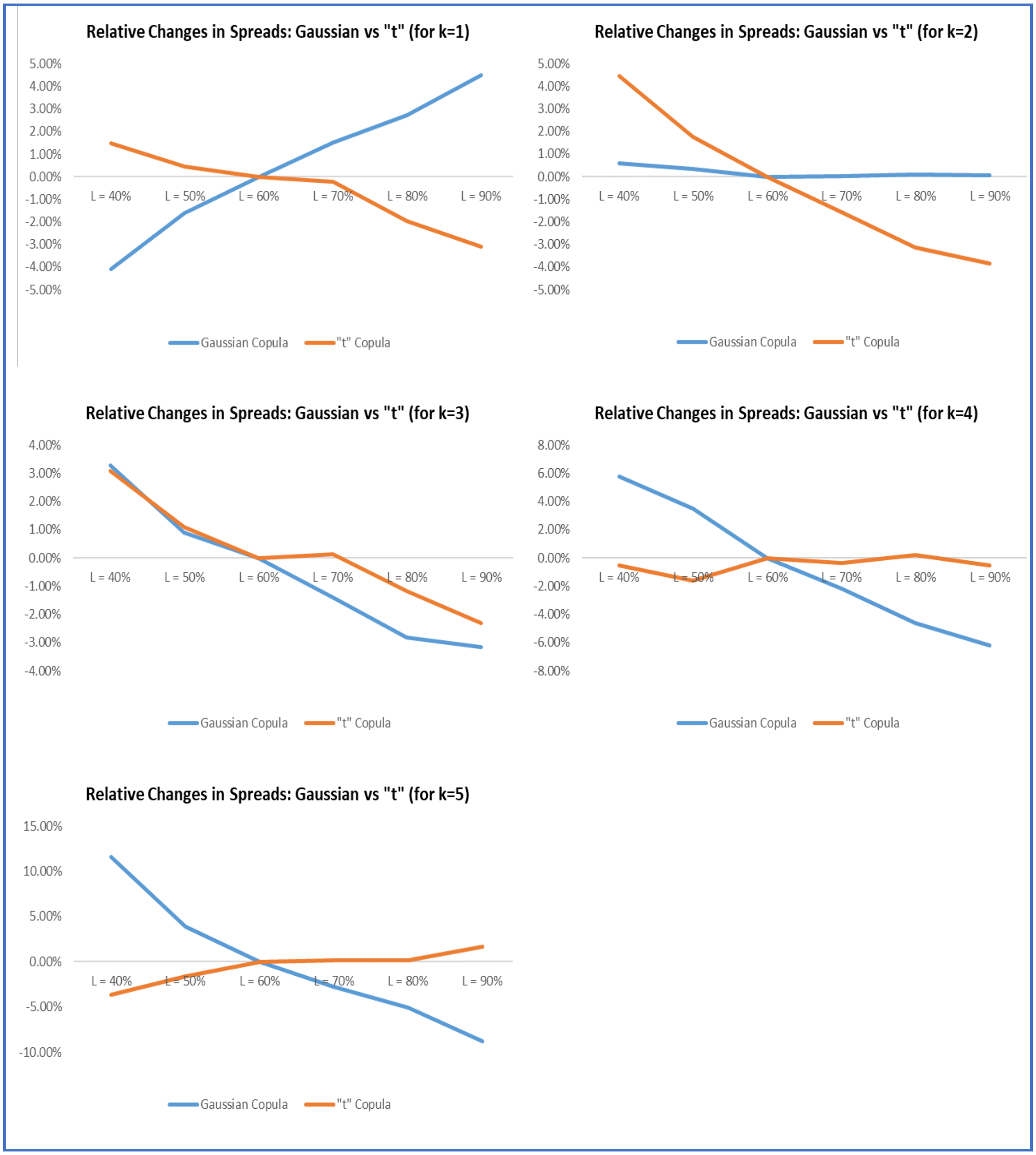
Stressing L involves two dynamics, acting in opposite directions:

- Increasing L leads to a decrease in hazard rates. A decrease in hazard rates reduces the default risk in the portfolio and hence should lead to a decrease in the basket spread
- Stressing L also impacts the default payout. An increase in L increases the default payout and hence should be reflected in higher spreads

The overall impact, i.e., whether spreads will increase or decrease, will depend on the relative magnitude of these two forces. In this particular example, these two forces almost cancelled each other out and the impact on the spreads were negligible. Under the Gaussian Copula, the maximum increase was 1.12 bps (k=1, L=90%); under the “t” Copula, the corresponding number was 0.60 bps (k=2, L=40%). This behavior can be seen in the charts below – the spread curves are almost indistinguishable from each other and exhibits an insensitivity to changes in L.



Similar to the previous case (stressed CDS spreads), relative shifts (measured as % changes against the base scenario) were also assessed. Under the Gaussian Copula, for k=1, changes in spreads and L were positively correlated; for all other k’s, the correlation was negative. Under the “t” Copula, for k=1, 2 and 3, the correlation was negative; for k=4, the correlation was close to zero; for k=5, the correlation was mildly positive. Unlike the previous case, no consistent pattern was found between stressed L and changes in spreads. Further details are provided in the charts below:

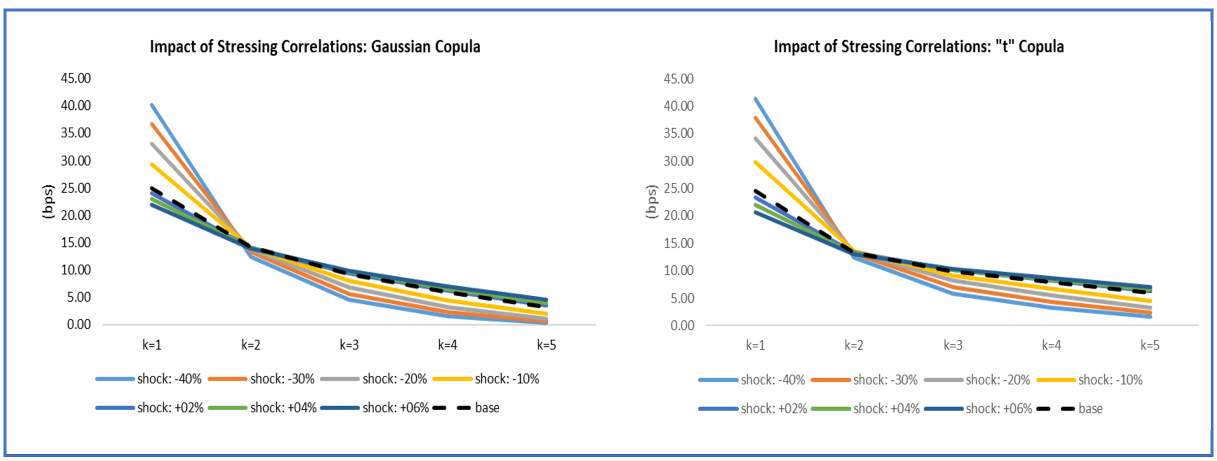

Relative Changes in Spreads: Gaussian vs "t" (for k=1)
5.00%
4.00%
3.00%
2.00%
1.00%
0.00%
-1.00%
-2.00%
-3.00%
-4.00%
-5.00%
L = 40%
L = 50%
L = 60%
L = 70%
L = 80%
L = 90%
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=2)
5.00%
4.00%
3.00%
2.00%
1.00%
0.00%
-1.00%
-2.00%
-3.00%
-4.00%
-5.00%
L = 40%
L = 50%
L = 60%
L = 70%
L = 80%
L = 90%
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=3)
4.00%
3.00%
2.00%
1.00%
0.00%
-1.00%
-2.00%
-3.00%
-4.00%
L = 40%
L = 50%
L = 60%
L = 70%
L = 80%
L = 90%
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=4)
8.00%
6.00%
4.00%
2.00%
0.00%
-2.00%
-4.00%
-6.00%
-8.00%
L = 40%
L = 50%
L = 60%
L = 70%
L = 80%
L = 90%
Gaussian Copula
"t" Copula
Relative Changes in Spreads: Gaussian vs "t" (for k=5)
15.00%
10.00%
5.00%
0.00%
-5.00%
-10.00%
L = 40%
L = 50%
L = 60%
L = 70%
L = 80%
L = 90%
Gaussian Copula
"t" Copula

6.3 Sensitivity to Correlations

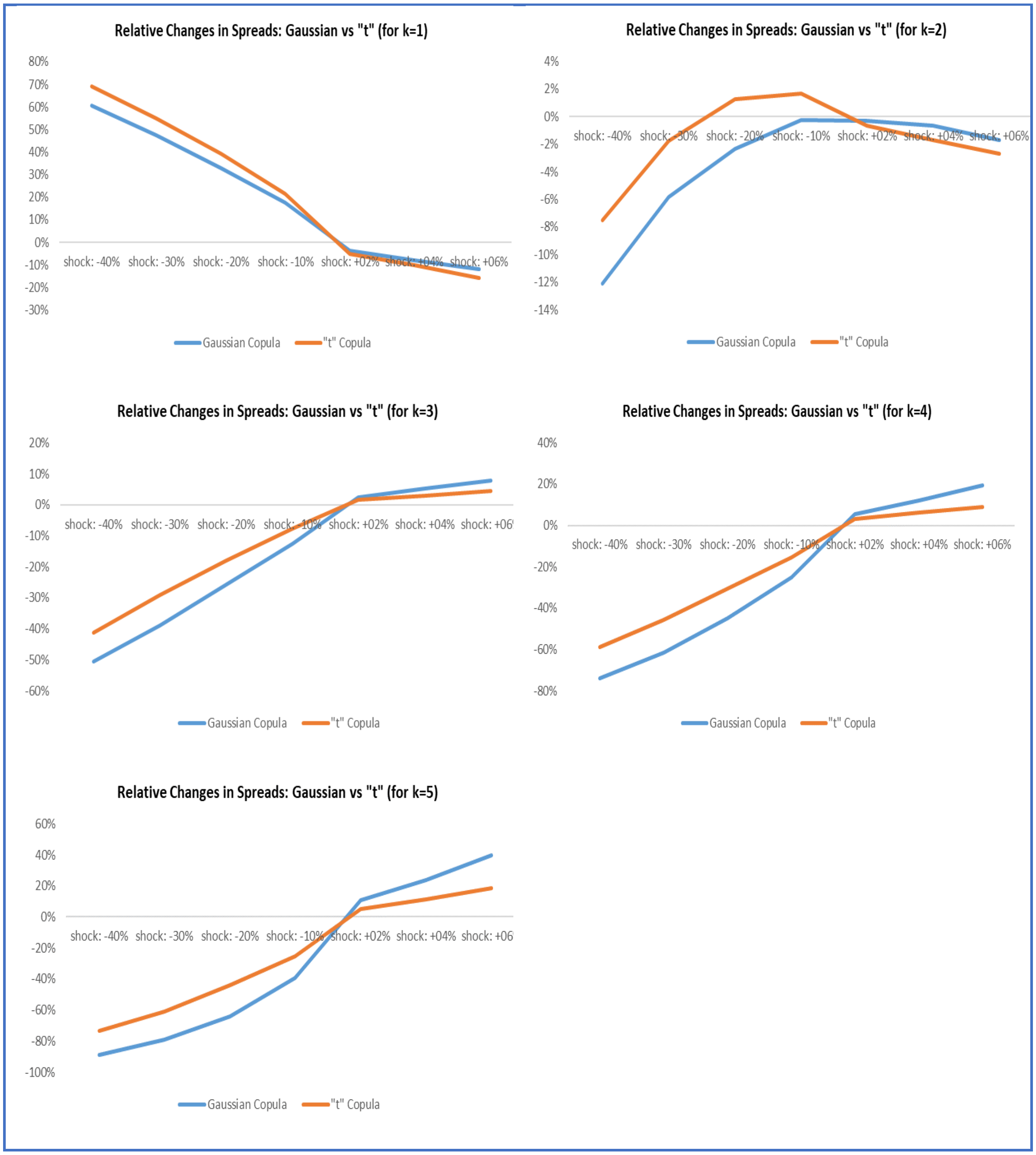
Amongst all input parameters, the prices of correlation products are most sensitive to correlation. In the context of this paper, correlations were stressed in both directions: on the negative side, 4 shocks were applied (-40%, -30%, -20%, -10%); on the positive side the shocks were significantly smaller given the already high correlations in the dataset (+.02%, +.04%, +.06%).

Increasing correlations should lead to an increased incidence of joint defaults; decreasing correlations, on the other hand, should do exactly the opposite. In terms of spreads, this can be stated as: all other factors held constant, higher correlations should lead to increased spreads for higher k's and reduced spreads for lower k's; for lower correlations, the converse should hold true.

The results presented in this paper were found to be consistent with the above hypothesis. When correlations were reduced by 40%, spreads for k=1 went up from 24.96 bps to 40.12 bps and for k=5 dropped from 3.25 bps to 0.35 bps under the Gaussian Copula; under the "t" Copula, spreads for k=1 went up from 24.50 bps to 41.39 bps and for k=5 the spreads dropped from 5.93 bps to 1.58 bps. When correlations were increased by 6%, the spreads for k=1 dropped from 24.96 bps to 21.96 bps and spreads for k=5 increased from 3.25 bps to 4.53 bps under the Gaussian Copula; under the "t" Copula, the spreads for k=1 dropped from 24.50 bps to 20.64 bps and for k=5 the spreads increased from 5.93 bps to 7.02 bps. This asymmetric impact of correlations on basket spreads can be seen in the charts below:



Similar to the two other input parameters (CDS spreads and recovery rates), relative changes were also analyzed (measured as % changes against the base scenario). The magnitude of relative changes across both the Gaussian and "t" Copulas were very similar (for k=1, changes under the "t' Copula were marginally higher; for all other k's, changes under the Gaussian Copula were marginally higher). In addition, the pattern of changes were also found to be similar across the two Copulas: when correlations were reduced, the spreads widened for k=1 and contracted for k >1; when correlations were increased, the converse was true (note that for k=2, the behavior was relatively less symmetric as compared to the other cases). The charts presented below summarizes these dynamics:



Appendix A: Package Structure

The submission contains the following contents:

1. Project Report – CQF Final Project Report_Anirbid Ghosh_v1.0.pdf
2. Declaration – Declaration_Anirbid Ghosh.pdf. This contains the declaration that this work, including drafting of the report, conducting data analysis, and writing up the code has been completely done by me. In this context, it may be worth stating that all the programs have been coded from first principles and no of the shelf softwares/packages have been used (except for standard functions for computing CDFs, estimating inverse probability functions, etc.)
3. Program – The program is primarily written in Python and consists of the following files:
   a. cds_basket_impl_all.ipynb: This is the primary iPython notebook and is grouped into 4 sections: Section I: definition and initialization of data structures, Section II: primary functions for estimating spreads, performing sensitivity analysis and testing convergence, Section III: various helper functions that are required by the primary functions in Section II, and Section IV: end-user function calls that invoke the primary functions in Section II
   b. corr.ipynb: This file contains the correlation analysis – for both Pearson's correlation and Kendall's Tau. For Pearson's correlation, the analysis includes the assessment of data including histograms and Q-Q plots and subsequent transformation of the data to normal before the correlation matrix is computed. For Kendall's Tau, the estimation of correlations – both pre and post linearization – are provided in this file
   c. dipping: The file is used for estimating the degree of freedom parameter used in the "t" Copula. The estimation was done using MLE (based on historical equity returns)
   d. sobol.R: The Sobol sequences were generated using R since a similarly robust package was not found in Python
4. Data files: There are 4 data files that have been included with this package:
   a. corr.csv: This file contains historical equity returns that have been used to estimate the correlation matrix and degree of freedom
   b. pseudo-random.csv: Contains a matrix of [100,000 X 5] pseudo-random numbers that are normally distributed
   c. sobol.csv: Contains a matrix of [100,000 X 5] Sobol numbers that are normally distributed
   d. chisq.csv: Contains a matrix of [100,000 X 5] pseudo-random numbers that follow a chi-square distribution

   Note: The random numbers were stored and accessed from files because: a) the Sobol number generation was done in R and could not be done at run-time, b) made the debugging process easier. However, if desired, the random numbers can be generated runt-time as well (except for Sobol sequences) – please uncomment this code within the "simulate copula" functions

Appendix B: Instructions to Run the Program

The following steps need to be followed while running the program:

Step I: Store all the data files (corr.csv, pseudo_random.csv, sobol.csv and chisq.csv) in the working directory

Step II: Store all the code files in the working directory (cds_basket_impl_al.ipynb, corr.ipynb, df.ipynb) in the working directory

Step III: Appropriately update the file paths in cds_basket_impl_all.ipynb (for pseudo_random.csv, sobol.csv and chisq.csv). These paths need to be updated within 2 functions: simulate_gaussian_copula and simulate_t_copula

Step IV: Go to Section IV within cds_basket_impl_all.ipynb. Invoke the first function call (calculate_spreads) to calculate the spreads under the Gaussian and "t" Copulas

Step V: Go to Section IV within cds_basket_impl_all.ipynb. Invoke the second function call (test_convergence) to calculate spreads and assess convergence across different simulation runs (1000, 5000, 10000, 25000, 50000, 75000, 100000) using pseudo-random numbers and Sobol sequences

Step VI: Go to Section IV within cds_basket_impl_all.ipynb. Invoke the third function call (calculate_stressed_spreads) to perform sensitivity analysis:

a. For stressing CDS spreads, set the first parameter in the function call to "True", the other two parameters should be set to "False"
b. For stressing correlations, set the second parameter in the function to "True", the other two parameters should be set to "False"
c. For stressing recovery rates, set the third parameter in the function to "True", the other two parameters should be set to "False"

Important Note: This function uses global variables that need to be reset after every simulation run (within Section I). For example, after stressing CDS spreads and before running the simulation for stressed correlations, please go to the data initialization cell within the iPython notebook (Section I) and execute the cell

Additional Instructions: All the data structures and global variables are initialized within Section I. The program allows users the flexibility to change any of these variables (e.g., change the default random number sequence from pseudo-random to Sobol) and recalculate the spreads

Appendix C: References

Doc Id	Document Name
1	Introduction to Quantitative Finance, Paul Wilmott
2	Monte Carlo Methods in Finance, Peter Jaeckel
3	Structured Credit Products: Credit Derivatives & Synthetic Securitization, Moorad Chowdhry
4	Credit Derivatives Pricing Models: Models, Pricing & Implementation, Phillip J. Schonbucher
5	CQF Class Notes, Dr. Alonso Pena
6	CQF Class Notes, Dr. Siyi Zhou
7	CQF Project Notes, Dr. Richard Diamond
8	Kendall's Rank Correlation Coefficient, Wikipedia
9	Construction & Comparison of High Dimensional Sobol Generators, Alexander Kreinin & Sergei Kucherenko
10	Notes of Generating Sobol Sequences, Stephen Joe & Frances Y. Kuo
11	CQF Final Project Submission (2014), Avtar Sehra
12	CQF Final Project Submission (2014), Bertrand Le Nezet