

Copula functions in Credit Metrics' VaR estimation

Capstone Project by Shynggys Magzanov

Supervisor: Dr.Dongming Wei, Second Reader: Dr.Zhenisbek Assylbekov

Abstract

Credit risk modelling of a portfolio of exposures is essential part of activity of every financial institution. However this procedure is complicated since the joint behavior of chosen exposures must be known. In this paper Value at Risk percentile of the portfolio consisting of three corporate bonds issued by Lukoil, Gazprom and Norilsk Nickel was estimated at three different significance levels within the frame of Credit Metrics approach proposed by J.P.Morgan. Following the Asset value model, Monte-Carlo simulations were performed to obtain possible portfolio values in one year time horizon. Where the joint distribution of asset returns of three companies was constructed by means of pair-copula construction method discussed in Aas, Czado, Frigessi,Bakken (2009). Results reveal that for particular portfolio of bonds at 90%, 95% and 99% confidence levels the value of our portfolio will not fall below 2057.915 ,1798.117 and 1375.011 dollars respectively.

1. Introduction

Credit Metrics is a methodology of estimating possible changes in value of the portfolio of exposures provided by J.P.Morgan [3]. The type of risk under the consideration is the one caused by the change in a credit rating of the issuer of the financial instrument. Since the subject of the interest is the whole portfolio value, the joint behavior of the exposures must be analyzed and properly modelled. Conventional Credit Metrics' approach is based on the assumption that asset returns of each issuer company is univariate Normally distributed and their joint behavior is modelled by multivariate Gaussian distribution. Being relatively easy to apply, given assumption underestimates possible risks due to inability of the model to properly describe the dependence structure between companies' asset returns. Within the frame of the Credit Metrics' Asset value model, the main objective of the project is to capture the dependence structure in a portfolio of bonds issued by Lukoil, Gazprom and Norilsk Nickel by means of Copula functions. Copula functions are powerful tools that were discovered by Sklar back in 1959, but found their application in finance only by the late 90s of the last century. Copulas are used in modelling a multivariate distributions given margins, and are useful in separating the information about margins from their dependence structure [2]. Before getting started with portfolio analysis, this paper gives a general overview of Copula theory, conventional dependence measures and method of pair-copula construction.

2. Theory

2.1 Copulas

2.1.1. Concept of Copula and properties of Copulas

Every joint distribution function of random variables contains information about its margins and dependence structure between them. Concept of Copula comes in handy when it is needed to isolate that dependence structure from margins. Copula is a function that links univariate margins and their joint multivariate distributions. Let us consider the properties that a function must satisfy in order to be a copula.

Definition 2.1. A d -dimensional copula is a distribution function on $[0, 1]^d$ with standard uniform marginal distributions. C is a mapping such that $C : [0, 1]^d \rightarrow [0, 1]$ and satisfying the following properties:

1. $C(u_1, \dots, u_d)$ is increasing in each component u_i .
2. $C(1, \dots, 1, u_i, 1, \dots, 1) = u_i$ for all $i \in \{1, \dots, d\}, u_i \in [0, 1]$
3. For all $(a_1, \dots, a_d), (b_1, \dots, b_d) \in [0, 1]^d$ with $a_i \leq b_i$ we have

$$\sum_{i_1=1}^2 \dots \sum_{i_d=1}^2 (-1)^{i_1+\dots+i_d} C(u_{1i_1}, \dots, u_{di_d}) \geq 0, \quad (1)$$

where $u_{j1} = a_j$ and $u_{j2} = b_j$ for all $j \in \{1, \dots, d\}$. [1]

It turns out that for every copula function there are upper and lower bounds, called Frechet-Hoeffding bounds, that are described in the following theorem.

Theorem 2.2. For every copula $C(u_1, \dots, u_d)$ there are upper and lower bounds [1]:

$$\max \left\{ \sum_{i=1}^d u_i + 1 - d, 0 \right\} \leq C(\mathbf{u}) \leq \min_{1 \leq i \leq d} \{u_i\} \quad (2)$$

However, it can be proven that the concept of the lower bound cannot be extended to dimensions higher than two. [1]

The following theorem is a fundamental building block that states the existence of copula functions and allows one to construct joint distribution function using copulas and margins.

Theorem 2.3.(Sklar's Theorem) Let F be a joint distribution function with margins F_1, \dots, F_d . Then there exists a copula $C : [0, 1]^d \rightarrow [0, 1]$ such that, for all x_1, \dots, x_d in $\bar{\mathbb{R}} = [-\infty, \infty]$,

$$F(x_1, \dots, x_d) = C(F_1(x_1), \dots, F_d(x_d)). \quad (3)$$

If the margins are continuous, then C is unique; otherwise C is uniquely determined on $RanF_1 \times RanF_2 \times \dots \times RanF_d$, where $RanF_i = F_i(\overline{\mathbb{R}})$ denotes the range of F_i . Conversely, if C is a copula and F_1, \dots, F_d are univariate distribution functions, then the function F defined in (3) is a joint distribution function with margins F_1, \dots, F_d .

Proposition 2.4.(Invariance Principle) Let (X_1, \dots, X_d) be a random vector with continuous margins and copula C and let T_1, \dots, T_d be strictly increasing functions. Then $(T_1(X_1), \dots, T_d(X_d))$ also has copula C .

Sklar's theorem along with Proposition 2.4 and Probability integral and Quantile transformations yields a powerful result making it possible to construct a multivariate distribution given a vector of any continuous random variables.

2.1.2. Families of Copulas

There exist different types of Copulas that can be classified into three major groups: Fundamental copulas, Implicit and Explicit copulas. Special dependence structures as independence or comonotonicity are represented by so called fundamental copulas. Copulas that are derived from known multivariate distributions but not having an explicit closed-form expressions are known as Implicit copulas. Whereas in contrast, copulas that do possess simple closed-form are Explicit copulas[1].

Fundamental Copulas

The *independence copula*, as the name suggests, describes independence of random variables and has a form:

$$\Pi(u_1, \dots, u_d) = \prod_{i=1}^d u_i \quad (4)$$

Which means that random variables U_1, \dots, U_d are independent if and only if their Copula is Π . *Comonotonicity copula* is the upper bound of every copula function mentioned in equation (2). It represents perfectly positive dependence structure between random variables. While perfectly negative dependence structure is described by the lower Frechet bound in (2) and called *countermonotonicity copula*.

Implicit Copulas

One of the most popular representatives of implicit copulas is Gaussian copula, which was introduced in valuation of financial derivatives as Credit Swaps, First-to-Default Contracts by Li (1999). Being widely adopted by finance professionals due to its simplicity before the Financial Crisis in 2008, later it was accused as "The formula that killed Wall Street" by Felix Salmon. General representation of Gaussian

copula is as follows:

$$C_P^{Ga}(\mathbf{u}) = P(\Phi(X_1) \leq u_1, \dots, \Phi(X_d) \leq u_d) = \Phi_P(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_d)) \quad (5)$$

Where Φ_P is a standard univariate normal distribution function, and Φ_P stands for the joint distribution function of the vector of random variables $\mathbf{X} \sim N_d(\mathbf{0}, P)$.

As it was proven by Li (1999), Asset Value method of the Credit Metrics approach implicitly uses Gaussian Copula in it's correlation formula, even though the Credit Metrics Technical Document does not refer to the concept of copula.

Another member of the given family is a d -dimensional t Copula, expressed as

$$C_{\nu, P}^t(\mathbf{u}) = \mathbf{t}_{\nu, P}(t_{\nu}^{-1}(u_1), \dots, t_{\nu}^{-1}(u_d)) \quad (6)$$

t_{ν} in equation (6) represents standard univariate t distribution, while the joint distribution function of the vector $\mathbf{X} \sim t_d(\nu, \mathbf{0}, P)$ is denoted as $\mathbf{t}_{\nu, P}$.

According to Mashal & Zeevi (2002) and Breymann (2003) when it comes to modelling multivariate joint distribution of financial returns data, t copula outperforms Gaussian copula. This is partially related to the ability of the t copula to capture phenomena of extreme values, which is a prevalent occurrence in financial return data [1].

Explicit Copulas

As an illustration of explicit copulas, representatives of Archimedean copulas, Gumbel and Clayton, are chosen.

Bivariate Gumbel copula

$$C_{\theta}^{Gu}(u_1, u_2) = \exp\{-((-\ln u_1)^{\theta} + (-\ln u_2)^{\theta})^{\frac{1}{\theta}}\}, 1 \leq \theta < \infty \quad (7)$$

Bivariate Clayton copula

$$C_{\theta}^{Cl}(u_1, u_2) = (u_1^{-\theta} + u_2^{-\theta} - 1)^{-\frac{1}{\theta}}, 0 < \theta < \infty \quad (8)$$

For further discussion of Copula families the readers are encouraged to refer to Nelsen (1999).

2.2. Measures of Dependence

In a modern finance theory the concept of correlation is of high importance. However, Embrechts, McNeil and Straumann (1999) argue that this concept is a source of confusion in that frame. Financial theory refers to the term ‘‘correlation’’ to describe any type of dependence between random variables. When in fact, correlation is a misuse of a particular dependence structure. While it might be a decent

measure of dependence when risks are elliptically distributed, this will not be true in realm where risks do not follow elliptical or spherical distributions [5]. Section 2.2 concentrates on types of dependence measures, their limitations and relationships with copula functions.

2.2.1. Linear correlation

Definition 2.5. Given random variables X and Y with finite variances, the linear correlation coefficient is defined as

$$r = \frac{Cov[X, Y]}{\sqrt{Var(X)Var(Y)}} \quad (9)$$

Where Covariance between two random variables is $Cov[X, Y] = E[XY] - E[X]E[Y]$.

As the name suggests, Pearson's linear correlation coefficient measures linear dependence between random variables. $-1 \leq r[X, Y] \leq 1$, where -1 corresponds to the perfect negative correlation and 1 represents perfect positive correlation. However, there are some limitations of the given metric:

- Random variables must follow Normal distribution for the linear correlation coefficient to be a reasonable measure of dependence.
- It can be noticed from eq.(9) that linear correlation coefficient requires finite variances of considered random variables. This may cause a problem when dealing with heavy-tailed distributions [5].
- Independence of random variables imply zero linear correlation between them, but the converse is not necessarily true.
- Even though it is invariant under strictly increasing linear transformations, it is *not* invariant under *non-linear* strictly increasing transformations[5].

2.2.2. Rank correlation coefficients

The two widely used nonparametric measures of association are Kendall's tau and Spearman's rho. Both of them are measures of concordance between two continuous random variables X and Y . Kendall's tau is represented as:

$$\tau = P[(X_1 - X_2)(Y_1 - Y_2) > 0] - P[(X_1 - X_2)(Y_1 - Y_2) < 0] \quad (10)$$

In the following theorem Nelsen (1999) represents the way in which Kendall's tau depends only on Copula function.

Theorem 2.6. Let X and Y be continuous random variables whose copula is C . Then the population version of Kendall's tau for X and Y is given by

$$\tau_{X,Y} = 4 \int_0^1 \int_0^1 C(u, v) dC(u, v) - 1 \quad (11)$$

In case of *Spearman's rho* consider three independent random vectors $(X_1, Y_1), (X_2, Y_2)$, and (X_3, Y_3) .

Then Spearman's rho is defined as:

$$\rho_{X,Y} = 3(P[(X_1 - X_2)(Y_1 - Y_3) > 0] - P[(X_1 - X_2)(Y_1 - Y_3) < 0]) \quad (12)$$

In the same fassion as Theorem 2.6, it can be shown that

$$\rho_{X,Y} = 12 \int_0^1 \int_0^1 C(u, v) dudv - 3 \quad (13)$$

Distinction between Kendall's tau and Spearman's rho comes from dependence structure that they measure. While the first one is a metric of average likelihood ratio dependence, the latter one focuses on measuring average quadrant dependence [2].

2.2.3. Coefficients of Tail dependence

If the subject of concern is the joint behavior of two random variables in the tails of the distributions, then coefficients of upper and lower tail dependence must be analyzed.

Definition 2.7. Let X and Y be random variables with distribution functions F_1 and F_2 . The coefficient of upper tail dependence of X and Y is

$$\lim_{\alpha \rightarrow 1^-} P[Y > F_2^{-1}(\alpha) | X > F_1^{-1}(\alpha)] = \lambda_U \quad (14)$$

provided a limit $\lambda \in [0; 1]$ exists. If $\lambda \in (0; 1]$, X and Y are said to be asymptotically dependent (in the upper tail); if $\lambda = 0$ they are asymptotically independent. Similarly, the lower tail dependence is

$$\lim_{\alpha \rightarrow 0^+} P[Y \leq F_2^{-1}(\alpha) | X \leq F_1^{-1}(\alpha)] = \lambda_L \quad (15)$$

Nelsen (1999) shows the relationship between tail dependence coefficients and copulas in the Theorem 2.8.

Theorem 2.8. Consider X, Y, F_1 and F_2 from Definition 2.7. Given the existence of upper and lower limits,

$$\lambda_U = 2 - \lim_{\alpha \rightarrow 1^-} \frac{1 - C(\alpha, \alpha)}{1 - \alpha} \quad (16)$$

$$\lambda_L = \lim_{\alpha \rightarrow 0^+} \frac{C(\alpha, \alpha)}{\alpha} \quad (17)$$

From equations (11),(13),(16) and (17) it can be seen that given a copula function, one can deduce different dependence measures.

2.3 Pair-Copula Construction

Modelling bivariate distribution by means of copulas is relatively easy task, since there are a lot of work devoted to that topic. However, things get complicated as the dimension of the distribution gets higher. This project makes use of the hierarchical model called Pair-Copula construction, described in Aas, Czado, Frigessi, Bakken (2009). The main idea is to decompose a multivariate density into blocks of bivariate pair copulas, applied on original variables and on their conditional and unconditional distribution functions [6]. Details of algorithms described below can be found in [6].

Let $\mathbf{X} = (X_1, \dots, X_n)$ be a vector of random variables with a joint pdf $f(x_1, \dots, x_n)$. Then this density can be expressed as

$$f(x_1, \dots, x_n) = f_n(x_n)f(x_{n-1}|x_n)\dots f(x_1|x_2, \dots, x_n) \quad (18)$$

From Sklar's theorem (eq.(3)) for absolutely continuous F with strictly increasing, continuous marginal CDFs F_1, \dots, F_n we obtain

$$f(x_1, \dots, x_n) = c_{1\dots n}\{F_1(x_1), \dots, F_n(x_n)\}f_1(x_1)\dots f_n(x_n) \quad (19)$$

where $c_{1\dots n}$ - n-variate copula density. The proof of (19) for the bivariate case was taken from Meucci (2011). Consider the same random vector of continuous random variables $\mathbf{X} \sim F$, where $X_i \sim F_i$ for $i = 1, \dots, n$. By the Definition of Copula

$$\begin{aligned} F_{\mathbf{U}}(\mathbf{u}) &= P\{U_1 \leq u_1, \dots, U_n \leq u_n\} = P\{F_1(X_1) \leq u_1, \dots, F_n(X_n) \leq u_n\} \\ &= P\{X_1 \leq F_1^{-1}(u_1), \dots, X_n \leq F_n^{-1}(u_n)\} = F(F_1^{-1}(u_1), \dots, F_n^{-1}(u_n)) \end{aligned} \quad (20)$$

By differentiating both sides, one can obtain Sklar's Theorem (for bivariate case the result is provided below)

$$\begin{aligned} f_{\mathbf{U}}(u_1, u_2) &= \partial_{u_1, u_2}^2 F_{\mathbf{U}}(u_1, u_2) = \partial_{u_1, u_2}^2 F(F_1^{-1}(u_1), F_2^{-1}(u_2)) \\ &= \partial_{x_1, x_2}^2 F(F_1^{-1}(u_1), F_2^{-1}(u_2)) d_{u_1} F_1^{-1}(u_1) d_{u_2} F_2^{-1}(u_2) \\ &= \frac{\partial_{x_1, x_2}^2 F(F_1^{-1}(u_1), F_2^{-1}(u_2))}{d_{x_1} F_1(F_1^{-1}(u_1)) d_{x_2} F_2(F_2^{-1}(u_2))} \\ &= \frac{f(F_1^{-1}(u_1), F_2^{-1}(u_2))}{f_1(F_1^{-1}(u_1)) f_2(F_2^{-1}(u_2))} \end{aligned} \quad (21)$$

By rearranging the equation (21) the following equality is obtained:

$$f(F_1^{-1}(u_1), F_2^{-1}(u_2)) = f_{\mathbf{U}}(u_1, u_2) \cdot f_1(F_1^{-1}(u_1)) \cdot f_2(F_2^{-1}(u_2)) \quad (22)$$

where $f_{\mathbf{U}}(u_1, u_2)$ is the Copula density.

Every conditional density term in (18) can be decomposed as follows:

$$f(x|\mathbf{v}) = c_{xv_j|\mathbf{v}_{-j}}\{F(x|\mathbf{v}_{-j}), F(v_j|\mathbf{v}_{-j})\}f(x|\mathbf{v}_{-j}), \quad (23)$$

where \mathbf{v} is a d -dimensional vector, v_j is a component of \mathbf{v} , and \mathbf{v}_{-j} represents a vector \mathbf{v} without j -th component. To conclude, according to Joe (1996), Bedford and Cooke (2001b,2002) cited in [6], a multivariate distribution can be decomposed as a product of pair-copulas, acting on distinct conditional PDFs under appropriate regularity conditions.

As it can be seen from equation(23) pair-copula construction model requires marginal conditional CDFs. Joe (1996) concluded that for $\forall j$,

$$F(x|\mathbf{v}) = \frac{\partial C_{x,v_j|\mathbf{v}_{-j}}\{F(x|\mathbf{v}_{-j}), F(v_j|\mathbf{v}_{-j})\}}{\partial F(v_j|\mathbf{v}_{-j})} \quad (24)$$

Where $C_{ij|k}$ in equation (24) represents bivariate copula distribution.

For univariate v , expression (24) is simplified as

$$F(x|v) = \frac{\partial C_{xv}\{F(x), F(v)\}}{\partial F(v)} \quad (25)$$

When x and v are uniform

$$h(x, v, \Theta) = F(x|v) = \frac{\partial C_{x,v}(x, v, \Theta)}{\partial v} \quad (26)$$

Where Θ - the set of parameters for the copula of the joint distribution of x and v .

For high-dimensional distributions there are number of distinct ways of possible pair-copula constructions. In order to arrange them Bedford and Cooke (2001b,2002) came up with a graphical model called *the regular vine* [6]. In this project the main subjects of concern are two special cases: the C- and D-vines.

In D-vine multidimensional distribution $f(x_1, \dots, x_n)$ is decomposed by the following expression:

$$\prod_{k=1}^n f(x_k) \prod_{j=1}^{n-1} \prod_{i=1}^{n-j} c_{i,i+j|i+1,\dots,i+j-1}\{F(x_i|x_{i+1}, \dots, x_{i+j-1}), F(x_{i+j}|x_{i+1}, \dots, x_{i+j-1})\} \quad (27)$$

Where index j identifies the trees, and index i identifies the edges in the tree.

While C-vine decomposition is

$$\prod_{k=1}^n f(x_k) \prod_{j=1}^{n-1} \prod_{i=1}^{n-j} c_{j,j+i|1,\dots,j-1}\{F(x_j|x_1, \dots, x_{j-1}), F(x_{j+i}|x_1, \dots, x_{j-1})\} \quad (28)$$

In trivariate case, like the one to be considered in this project, it can be easily deduced that C- and D-vine decompositions are the same

$$\begin{aligned}
f(x_1, x_2, x_3) &= f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \\
&\cdot c_{12}\{F_1(x_1), F_2(x_2)\} \cdot c_{23}\{F_2(x_2), F_3(x_3)\} \\
&\cdot c_{13|2}\{F(x_1|x_2), F(x_3|x_2)\}
\end{aligned} \tag{29}$$

It may be noticed that in (29) there are only three distinct decompositions, even though 6 permutations are possible.

Estimation of parameters for a C-vine

Since this project is focused on trivariate case, it is sufficient to mention parameter estimation only for the C-vine. For the C-vine, the log-likelihood function is expressed as

$$\sum_{j=1}^{n-1} \sum_{i=1}^{n-j} \sum_{t=1}^T \log[c_{j,j+i|1,\dots,j-1}\{F(x_{j,t}|x_{1,t}, \dots, x_{j-1,t}), F(x_{j+i,t}|x_{1,t}, \dots, x_{j-1,t})\}] \tag{30}$$

Note that in each sum the number of parameters to be estimated depends on the type of copula that was chosen.

Since the log-likelihood needs to be numerically maximized over all parameters, there is an algorithm for determining the starting values of optimization in Aas, Czado, Frigessi, Bakken (2006).

1. Estimation of the parameters of the copula in the first tree from the data.
2. Computing conditional distribution functions for the second tree using previously found copula parameters from tree 1 and the h-function defined in equation (26).
3. Estimation of the parameters of the copula in the second tree from the observations in step 2.
4. Computing observations for the third tree using previously found copula parameters from tree 2 and the h-function.
5. Estimation of the parameters of the copula in the third tree from the observations in step 4.
6. etc.

It is significant to mention that since the marginal distributions are unknown, parameter estimation must rely on the normalized ranks of the data. Implying that it is actually a pseudo-likelihood which is being maximized [6].

3. Application

The portfolio whose Value at Risk in one year time horizon is to be analyzed consists of corporate bonds issued by three Russian companies: Lukoil oil company, Gazprom and Norilsk Nickel. Table 1 represents detailed information about the chosen portfolio.

Bond	Issuer	Face value	Coupon(%)	Years till Maturity	S&P Rating	Seniority
1	LUKOIL	1000	4.750	7 years	BBB	Senior Unsecured
2	GAZPROM	1000	4.950	3 years	BBB-	Senior Unsecured
3	NORNICKEL	1000	6.625	3 years	BBB-	Senior Unsecured

Table 1: Bond Portfolio

As it was mentioned earlier, an algorithm to estimate VaR of the portfolio is based on Credit Metrics approach provided by J.P.Morgan [3]. Credit Migration Likelihood matrix and the Table of Recovery rates in case of default were provided by National Rating Agency of Russia and Moody's Investors Service respectively.(Figure 3 and Table 6) While one-year forward zero rates,which can be found in Table 7, were randomly generated from beta distribution by shifting parameters. According to Credit migration likelihood matrix, in one year credit ratings of our companies might migrate to 23 possible states, and therefore the value of the portfolio might change. Twenty three possible present values of each of the 3 bonds in year 2020 were calculated and discounted one year back, so that they are represented in today's money. Results are provided in Table 8. Note that there are no values of bonds in case of default, those are calculated by means of randomly generated recovery rates by using values from Table 6 as parameters of the beta distribution. In order to analyze the joint credit migration behavior of our portfolio of bonds, Asset value model of Credit Metrics was applied. According to this model, the process that drives change in credit rating of the company is it's asset value. The assumption is that there are certain thresholds of asset returns so that whenever asset value of a company in one year crosses that threshold, credit rating of the firm changes. But to find those thresholds for each company it is required to model the individual distributions of asset returns of every firm. For that purpose, stock returns of Lukoil, Gazprom and Norilsk Nickel were analyzed. Daily closing prices of stocks from 03.01.2006 to 22.02.2019 were provided by Yahoo Finance.

3.1. Determining stock returns distribution

Spearman's linear correlation coefficients reveal positive correlation between each pair of log-returns. Magnitudes of linear correlations are as follows: 0.8131511 between Lukoil and Gazprom, 0.6947848 between Lukoil and Norilsk Nickel, and 0.6998833 between Gazprom and Norilsk Nickel.

Table 2 contains some basic statistical information about log-returns of each company.

Company	Mean	St.dev	Skewness	Kurtosis
LUKOIL	0.0000966	0.0277087	-0.9949134	21.57685
GAZPROM	-0.0003492	0.02867866	-0.02841426	9.503326
NORNICKEL	0.0002536	0.03418533	-0.4783684	26.74476

Table 2: Statistics of log-returns

It can be noticed that log-returns of Lukoil and Norilsk Nickel have positive mean, while returns of Gazprom display negative mean. All the returns are left tailed and posses large kurtosis, which is a clear implication of non-normality of our log-returns. Small p-values of Jarque-Bera test for normality of our

log-returns confirm that. (Table 9)

The next step was to check if the log-returns possess serial correlation. By means of Ljung-Box test performed on squared log-returns for 20 lags it was concluded that all 3 log-return series do possess serial correlation at 1% significance level. Instead of performing a test for stationarity of given time series, it was decided to directly fit ARIMA model because it can make non-stationary time series stationary through differencing, as a result providing so called Integrated time series. It is a common appearance that the variance of stock return time series changes due to trends or seasonalities. Such variability is called Heteroscedasticity [1]. And most of the time Heteroscedasticity appears to be serially correlated and therefore, conditional on periods of increased variance. That is why such time series are referred as conditionally heteroscedastic. In order to capture these trends GARCH model was fitted. Parameters of ARIMA(p,d,q)-GARCH(m,n) are estimated by numerically optimizing the likelihood function. Number of lags p,d,q to be used in a model are determined by fitting ARIMA(p,d,q) models where p and q ranged from 1 to 4, and d ranged from 0 to 2. The best fit was determined by AIC criterion. On the residuals of appropriate ARIMA models, GARCH models were fitted using “tseries” library in R. As a result models in Table 3 were chosen.

Company	ARIMA(p,d,q)-GARCH(m,n)
LUKOIL	ARIMA(4,1,3)-GARCH(1,1)
GAZPROM	ARIMA(3,0,4)-GARCH(1,1)
NORNICKEL	ARIMA(4,1,3)-GARCH(1,1)

Table 3: ARIMA-GARCH model fit

Plots of ACF of residuals and PACF of squared residuals can be found in Figures 4,5,6. Furthermore, Ljung-Box test for serial independence, which was performed on the residuals of ARIMA-GARCH models, resulted in p-values 0.9652 for Lukoil, 0.9944 for Gazprom and 0.4609 for Norilsk Nickel, meaning that at 1% significance level residuals are serially independent.

3.2 Finding asset return thresholds

Another huge assumption made by Credit Metrics approach is that company’s asset returns are normally distributed and when portfolio consists of bonds issued by multiple companies, joint asset returns are multivariate normally distributed. In order to check the validity of that assumption, Normal distribution and Student t distributions were fitted to residuals obtained earlier. By comparing AIC and BIC criteria of both fits, it was found that t distribution describes returns data slightly better than Normal distribution for all 3 companies. Results of those fits and their corresponding parameters can be found in Tables 11,12. Then, using CDFs of these t distributions, asset return thresholds were calculated. As an illustration, let us consider thresholds of Lukoil. Let R denote the stock returns of Lukoil. From Table 1 it is evident that Lukoil is currently rated as BBB. To visualize the problem, consider the following Figure.

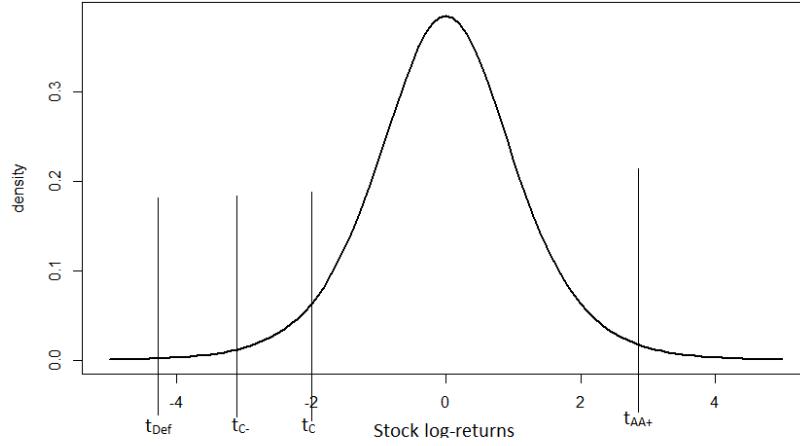


Figure 1: Lukoil stock log-return thresholds

The event of default of Lukoil equivalent to saying that log-returns of the firm's stocks fell below the default threshold. But since the probability of BBB rated company migrating to other ratings is provided by Credit Migration Likelihood matrix (Figure 3), the thresholds can be computed as follows:

$$\begin{aligned}
 Pr(\text{Default}) &= P(R \leq t_{Def}) = F_t(t_{Def}) = 0.1736 \\
 t_{Def} &= F_t^{-1}(0.1736) = -0.82389251 \\
 Pr(\text{Downgrade to C-}) &= P(t_{Def} < R \leq t_{C-}) = \\
 &= F_t(t_{C-}) - F_t(t_{Def}) = 0.0099 \\
 t_{C-} &= F_t^{-1}(0.0099 + F_t(t_{Def})) = -0.78751139 \\
 &\dots
 \end{aligned} \tag{31}$$

In the same fashion thresholds for all 22 credit ratings can be found for all 3 companies. Note that there is no need to find a threshold for AAA rating, since every log-return above the AA+ threshold will lead to the upgrade of a company to AAA. Asset return thresholds for all companies can be found in Tables 13,14,15. Now in order to perform Monte-Carlo simulations of possible portfolio values, it was needed to define joint behaviour of firms' credit migration. To model that behaviour, appropriate Copula function must be fitted.

3.2 Fitting Copula

Since the margins of Copula functions must be uniform on the unit interval, CDFs of respective t distributions were used to apply Probability integral transform to our log-returns so that they follow uniform distributions on [0,1]. With margins defined, the next step was to construct trivariate distribution by means of pair-copula construction method discussed above. "CDVineCondFit" function of "CD-

VineCopulaConditional” package in R was exploited in order to fit bivariate copula to each pair according to the algorithm explained above by conditioning Nornickel returns. The function is implemented in such a way that it fits all 40 copulas (list of implemented copulas can be found in R documentation), finds maximum likelihood estimators of parameters for each of them, and then chooses the best fit by BIC criteria. Table 4 shows the structure of pair-copula construction.

Tree	Edge	Copula	Par1	Par2	Kendall’s tau	Upper t.d.	Lower t.d.
1	3,1	t	0.62	7.51	0.42	0.19	0.19
	3,2	t	0.63	6.99	0.43	0.21	0.21
2	2,1 3	t	0.59	8.33	0.40	0.15	0.15

Table 4: Joint distribution of 3 companies

It can be seen that there is a positive dependence between companies’ asset returns expressed in terms of Kendall’s tau. t Copula is able to capture Tail dependence. Coefficients of Tail dependence in Table 4 reveal the extent to which companies’ asset returns are tend to jointly take extreme values.

3.3 Estimating VaR

3.3.1. Assigning Credit Ratings

With the purpose to perform Monte-Carlo simulation of possible portfolio values, 3305 random points were simulated from the modelled joint distribution. Figure 7 depicts pairplots of simulated points, where X1, X2 and X3 represent Lukoil, Gazprom and Nornickel respectively. Uniform marginal values of those points were transformed back using inverse CDFs of previously found respective t distributions. Then, those transformed randomly generated points represent possible asset returns of each company in comovement with other firms. For each scenario of possible asset returns appropriate credit rating was assigned to a firm based on which interval of return thresholds it lies. Let us continue with an illustration for Lukoil. Let the simulated asset return of Lukoil be equal to -0.8. From the example displayed in equation (35), it can be seen that -0.8 is in the interval (t_{Def}, t_C) . Therefore, in this particular possible case Lukoil will be assigned a credit rating of “C-”. Following the same procedure, appropriate credit ratings were assigned to all 3 companies.

3.3.2. Mapping Credit Ratings to Bond values

As it was mentioned earlier, in 1 year time horizon the present values of the bonds might change depending on what credit rating an issuer company will be assigned. Now the simulated credit ratings of each company from the previous section were mapped to the appropriate bond values from the Table 8. As an example let us consider the case of a particular realization where in 1 year Lukoil and Gazprom downgraded to “CC”, while Norilsk Nickel upgraded to “A”. In this case, according to the Table 8, the value of our portfolio is $624.9620+829.4743+984.2259 = 2438.6622$ dollars.

3.3.3. Estimating VaR

By the algorithm described above possible portfolio values in 1 year time horizon were calculated for all 3305 simulated scenarios. Figure 2 represents the histogram of possible values of our portfolio.

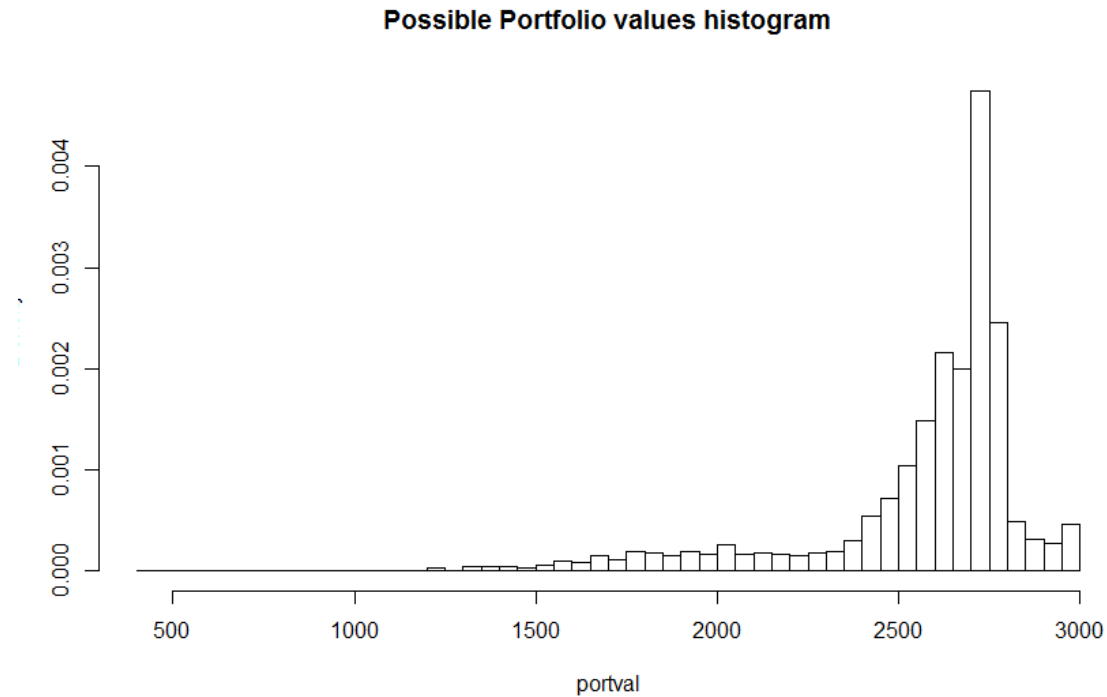


Figure 2: Histogram of possible portfolio values in one year time horizon

From here calculating VaR, which is an α -percentile, is a straightforward procedure, which amounts to ordering values in an ascending order, and choosing αN -th element.

As a result, with 90%, 95% and 99% confidence it can be concluded that the value of the portfolio will not fall below 2057.915, 1798.117 and 1375.011 dollars respectively. Alternatively, all the described procedures were performed following the assumptions of conventional Credit Metrics approach that asset returns follow Normal distribution and their joint behavior is modelled by multivariate Normal distribution. Results are provided in Table 5.

α	VaR using Copula	VaR using normality
10	2057.915	2080.033
5	1798.117	1806.129
1	1375.011	1374.067

Table 5: Values at Risk

Even taking into account the randomness coming from calculating value of a bond in case of default,

VaR of our portfolio calculated by following conventional Credit Metrics approach is more optimistic in comparison with VaR computed using Copula approach. The difference might be insignificant in particular case, but it potentially might cause problems when working with large portfolios.

4. Conclusion

In this project Copula functions were integrated to Credit Metrics methodology of credit risk modelling. Following Asset value approach, dependence structure of joint credit rating movements was modelled using asset returns of issuer companies. As it turned out t-distributions were best to describe asset returns of our companies. By means of pair-copula construction method results in Table 4 were yielded. It turned out that asset returns of Lukoil, Gazprom and Norilsk Nickel possess positive dependence structure. Moreover, there are upper and lower tail dependence between each pair representing degree to which returns might take extreme values jointly. From modelled joint distribution 3305 random points were generated in order to perform Monte Carlo simulation of possible joint asset returns. Those values were exploited to obtain possible values of our portfolio in one year. Then VaR percentile was estimated at different confidence levels.

5. Suggestions for further research

As it was mentioned above, pair-copula construction is only one of the ways of modelling multidimensional joint distribution by using copulas. So another method might be used to model joint distribution of asset returns. Moreover, R implementation of pair-copula construction has 40 copula types to be fitted, and the best fit was chosen by Goodness-of-fit test. Thus, increasing number of implemented copulas might result in a better fit. Concerning input parameters, real forward-zero rates can be used to make a complete analysis of the portfolio. Additionally, a test can be performed to figure out whether the differences of VaR's found in Table 5 are statistically significant.

6. Reference List

1. McNeil, A., Frey, R. and Embrechts, P. (2005). Quantitative Risk Management: Concepts, Techniques and Tools. [online] Available at: <https://www.researchgate.net/publication/301776007>
2. Nelsen, R. (1999). An introduction to copulas. New York: Springer.
3. Credit Metrics – Technical Document. J.P Morgan , New York, April 1997
4. Spratings.com. (2018). 2017 Annual Global Corporate Default Study and Rating Transitions. [online] Available at: <https://www.spratings.com/documents/20184/774196/2017+Annual+Global+Corporate+Default+Study/a4cffa07-e7ca-4054-9e5d-b52a627d8639>.
5. Embrechts, P., McNeil, A. and Straumann, D. (1999). Correlation and dependence in risk management: properties and pitfalls. <http://www.risknet.de/uploads/txbxlibrary/Embrechts-Correlations-1999-ETH-Paper.pdf>
6. Aas, K., Czado, C., Frigessi, A., and Bakken, H. (2009). Pair-copula constructions of multiple dependence. *Insurance: Mathematics And Economics*, 44(2), 182-198. doi: 10.1016/j.insmatheco.2007.02.001
7. Li, D. (1999). On Default Correlation: A Copula Function Approach. *SSRN Electronic Journal*. doi: 10.2139/ssrn.187289
8. Meucci, A. (2011). A Short, Comprehensive, Practical Guide to Copulas. *SSRN Electronic Journal*. doi: 10.2139/ssrn.1847864

7. Appendix

% от общего числа	AAA	AA+	AA	AA-	A+	A	A-	BBB+	BBB	BBB-	BB+	BB	BB-	B+	B	B-	CC+	CC	CC-	C+	C	C-	D
AAA	92.09%	7.42%	0.24%	0.22%	0.01%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
AA+	57.38%	36.51%	2.52%	2.47%	0.14%	0.10%	0.08%	0.20%	0.05%	0.02%	0.01%	0.01%	0.01%	0.01%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%	0.01%	0.07%
AA	28.60%	28.77%	31.42%	4.37%	0.41%	0.68%	0.98%	2.25%	0.67%	0.07%	0.13%	0.09%	0.08%	0.22%	0.00%	0.00%	0.29%	0.09%	0.00%	0.00%	0.01%	0.16%	0.62%
AA-	8.09%	11.94%	22.09%	36.92%	4.51%	3.42%	2.20%	3.68%	1.25%	0.50%	0.28%	0.15%	0.18%	0.34%	0.00%	0.00%	0.38%	0.18%	0.00%	0.01%	0.02%	0.20%	2.55%
A+	1.70%	3.07%	6.12%	17.25%	45.48%	13.13%	4.65%	3.29%	1.80%	0.36%	0.33%	0.16%	0.18%	0.41%	0.00%	0.00%	0.32%	0.18%	0.00%	0.00%	0.03%	0.15%	1.29%
A	0.85%	1.49%	1.62%	5.69%	17.97%	44.64%	8.24%	6.46%	3.75%	1.02%	0.73%	0.32%	0.56%	1.46%	0.00%	0.01%	0.75%	0.67%	0.00%	0.01%	0.08%	0.35%	3.33%
A-	0.18%	0.44%	0.39%	1.73%	6.74%	27.25%	23.79%	11.70%	7.72%	0.78%	2.10%	0.80%	0.69%	2.01%	0.01%	0.01%	2.53%	0.97%	0.00%	0.01%	0.18%	0.78%	9.18%
BBB+	0.08%	0.21%	0.25%	1.05%	4.27%	12.62%	14.55%	17.20%	11.28%	0.84%	2.37%	1.13%	1.29%	5.30%	0.01%	0.01%	7.01%	2.53%	0.00%	0.01%	0.16%	3.41%	14.22%
BBB	0.01%	0.04%	0.04%	0.21%	1.16%	3.48%	5.73%	13.24%	25.29%	2.43%	4.49%	1.14%	1.98%	10.81%	0.01%	0.02%	3.93%	7.26%	0.00%	0.01%	0.25%	0.99%	17.26%
BBB-	0.01%	0.02%	0.02%	0.13%	0.74%	2.22%	3.90%	10.22%	21.22%	7.22%	4.67%	0.82%	3.21%	4.89%	0.00%	0.02%	1.98%	7.29%	0.00%	0.02%	0.25%	0.64%	30.29%
BB+	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	6.23%	0.00%	0.00%	0.00%	0.00%	0.03%	0.00%	0.00%	0.00%	0.00%	0.37%	0.00%	93.37%
BB	0.00%	0.01%	0.01%	0.07%	0.43%	1.30%	2.44%	7.00%	16.48%	1.26%	2.42%	1.39%	1.03%	4.65%	0.00%	0.01%	51.15%	3.04%	0.00%	0.01%	0.13%	0.38%	6.67%
BB-	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	6.94%	0.00%	0.00%	0.00%	46.53%	0.00%	0.00%	0.00%	0.00%	0.00%	46.53%
B+	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
B	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%
B-	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%
CC+	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
CC	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%
CC-	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%
C+	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%
C	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%
C-	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%
D	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%

Figure 3: Credit Rating Migration Likelihood Table by National Rating Agency of Russia

Seniority	Recovery Rate (%)	st.dev
Senior Secured	62.3	15.4
Senior Unsecured	47.9	12.7
Subordinate	28	6.68

Table 6: Average Corporate Debt Recovery Rates Measured by Ultimate Recoveries, 1987-2017

Category	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6	Year 7
AAA	5.54	5.23	5.60	4.76	5.61	4.48	5.06
AA+	6.23	5.65	5.93	6.22	5.67	5.93	4.78
AA	5.81	5.88	5.77	6.16	6.38	6.15	5.82
AA-	6.40	6.48	6.49	6.34	6.84	5.94	6.30
A+	6.94	7.00	6.94	6.85	7.80	6.29	6.44
A	7.07	6.81	7.53	7.40	7.75	7.06	7.20
A-	7.16	8.08	6.95	7.61	6.94	7.36	7.31
BBB+	8.21	7.54	7.34	7.85	8.08	8.09	7.43
BBB	8.21	8.58	7.97	7.62	8.38	7.95	8.82
BBB-	7.36	7.92	8.30	8.26	8.35	8.46	8.53
BB+	9.41	9.72	9.04	9.30	8.76	8.78	9.29
BB	10.0	9.32	10.3	10.5	10.4	9.56	10.1
BB-	10.6	10.7	11.3	10.9	10.6	11.1	11.4
B+	12.1	10.7	11.5	11.9	11.5	12.2	11.7
B	12.9	12.5	11.6	12.4	11.8	12.5	12.7
B-	13.3	12.9	12.8	13.3	13.5	13.1	13.5
CC+	13.3	14.2	13.3	14.0	14.3	13.5	13.6
CC	13.9	14.2	15.2	14.8	13.5	15.1	13.4
CC-	14.6	13.8	15.1	14.6	14.9	14.2	14.6
C+	15.6	15.8	14.9	15.2	15.5	15.1	15.5
C	15.9	15.6	15.1	14.9	14.9	15.8	15.3
C-	15.2	14.5	14.9	14.8	15.1	14.7	15.2

Table 7: Forward rates generated from beta distribution by changing parameters

Year-end rating	LUKOIL bond	GAZPROM bond	NORNICKEL bond
AAA	950.0290	969.5942	1014.0616
AA+	923.8489	963.2285	1007.5380
AA	910.2512	962.5074	1006.7746
AA-	893.7346	951.5238	995.5367
A+	872.9929	943.5162	987.3345
A	857.3122	940.4325	984.2259
A-	853.1292	934.4570	977.9915
BBB+	840.2922	935.7844	979.4084
BBB	820.9605	922.3582	965.6354
BBB-	818.2014	925.2208	968.6281
BB+	785.9697	905.2442	948.1071
BB	755.5898	899.1159	941.9301
BB-	724.1725	881.6000	923.9692
B+	706.4897	879.8096	922.1473
B	685.5049	865.8387	907.7371
B-	657.5539	853.2991	894.9415
CC+	639.5347	841.9683	883.2936
CC	624.9620	829.4743	870.5918
CC-	621.2737	832.4938	873.7039
C+	600.9724	820.6981	861.4985
C	602.7211	820.5169	861.3401
C-	613.4543	829.4793	870.5647

Table 8: Possible present values of Bonds

Company	JB	p-value	Decision
LUKOIL	64765	<2.2e-16	Rej H_0
GAZPROM	12461	<2.2e-16	Rej H_0
NORNICKEL	98789	<2.2e-16	Rej H_0

Table 9: Jarque-Bera Test for Normality

Company	χ^2	df	p-value	Decision
LUKOIL	3108.1	20	<2.2e-16	Rej H_0
GAZPROM	5014.9	20	<2.2e-16	Rej H_0
NORNICKEL	2869.7	20	<2.2e-16	Rej H_0

Table 10: Ljung-Box Test for Serial Independence before ARIMA-GARCH model

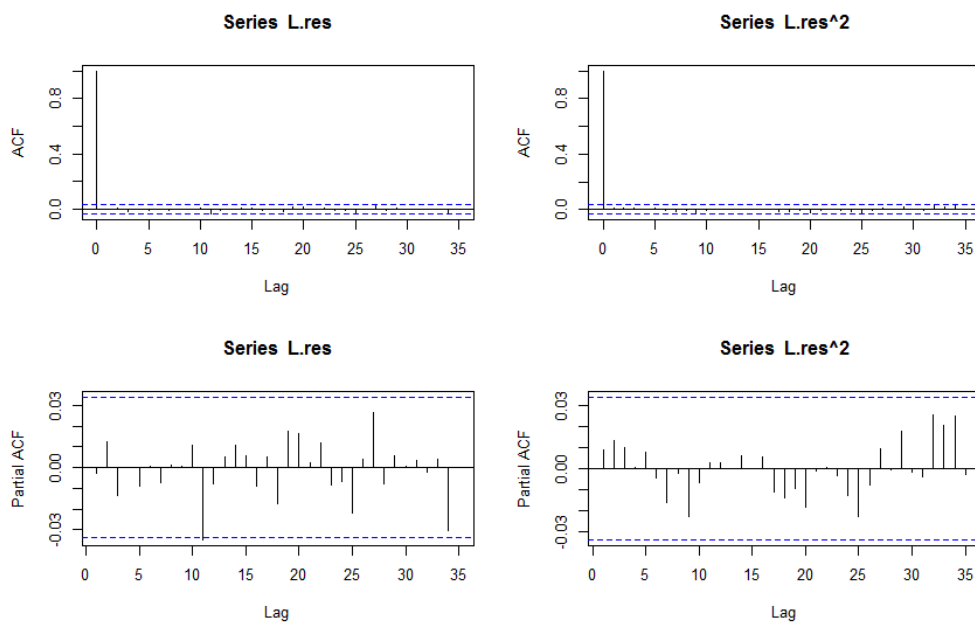


Figure 4: ACF and PACF of residuals and squared residuals of Lukoil log-returns after ARIMA(4,1,3)-GARCH(1,1)

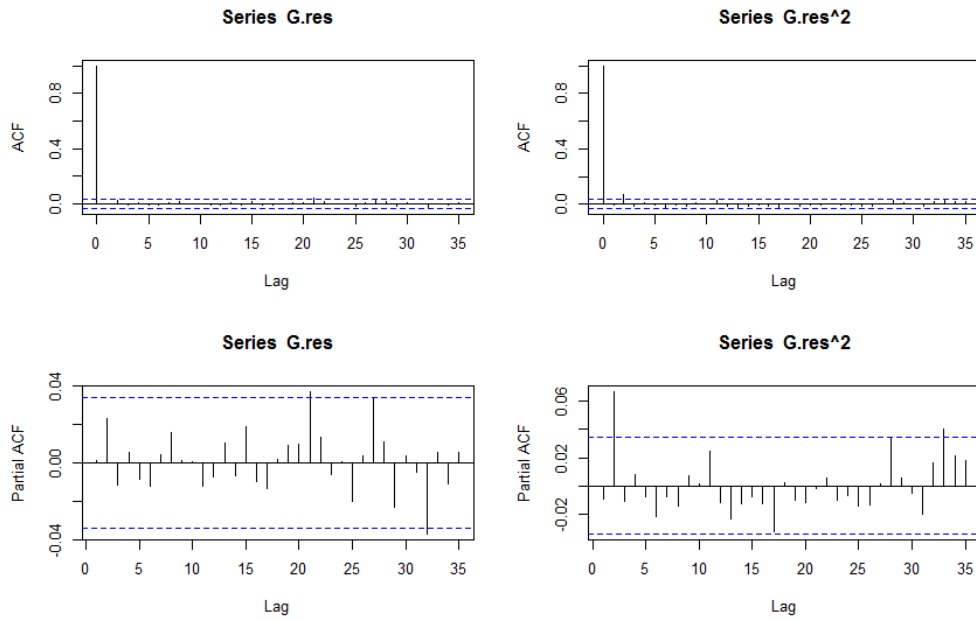


Figure 5: ACF and PACF of residuals and squared residuals of Gazprom log-returns after ARIMA(3,0,4)-GARCH(1,1)

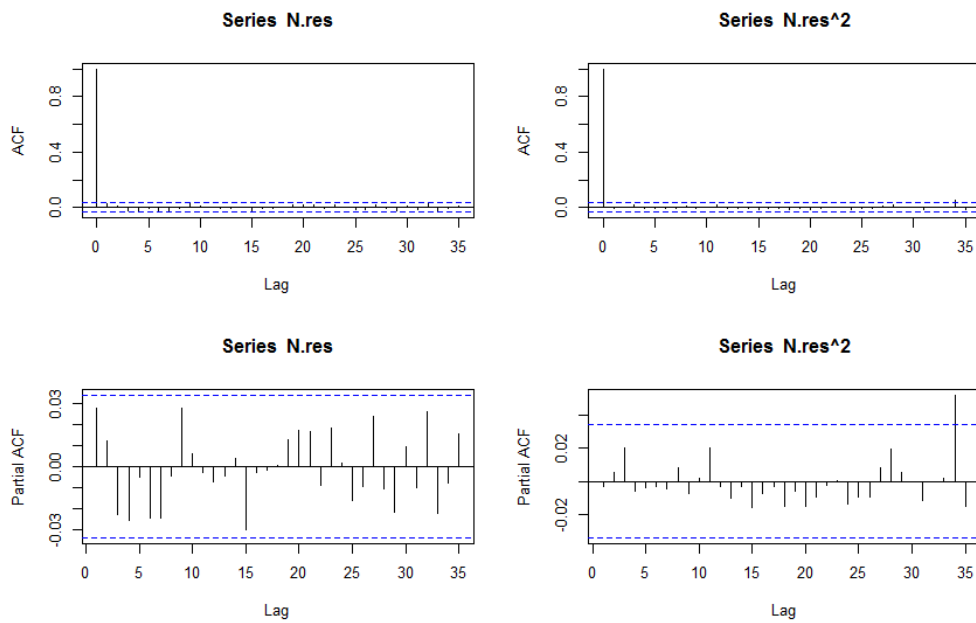


Figure 6: ACF and PACF of residuals and squared residuals of Nornickel log-returns after ARIMA(4,1,3)-GARCH(1,1)

Company	mean	st.dev	AIC	BIC
LUKOIL	0.002459044	1.000851347	9388.809	9401.015
GAZPROM	-0.01254965	1.00089028	9389.066	9401.272
NORNICKEL	0.01349338	1.00410348	9410.252	9422.459

Table 11: Parameters of Normal Distribution fit

Company	df	mean	st.dev	AIC	BIC
LUKOIL	6.53046942	0.01954184	0.83277494	9222.032	9240.342
GAZPROM	7.653713749	-0.008275583	0.861420330	9288.704	9307.014
NORNICKEL	6.57588223	0.03295066	0.82908809	9185.557	9203.867

Table 12: Parameters of Student t Distribution fit

AA+	6.23057956	BBB	0.64293344	B-	-0.44339645	D	-0.82389251
AA	4.71334745	BBB-	0.03472303	CC+	-0.44391275		
AA-	4.23347944	BB+	-0.01798963	CC	-0.54965555		
A+	3.34881242	BB	-0.11603182	CC-	-0.77821076		
A	2.33668737	BB-	-0.14120916	C+	-0.77821076		
A-	1.62154684	B+	-0.18535843	C	-0.77856662		
BBB+	1.16756519	B	-0.44313838	C-	-0.78751139		

Table 13: Asset return thresholds of LUKOIL

AA+	4.81171054	BBB	0.85366114	B-	-0.22289852	D	-0.47180171
AA	4.43095332	BBB-	0.25034147	CC+	-0.22336057		
AA-	4.19054120	BB+	0.08293131	CC	-0.26947527		
A+	3.48696049	BB	-0.02143760	CC-	-0.44815200		
A	2.54964412	BB-	-0.03973709	C+	-0.44815200		
A-	1.86220113	B+	-0.11161573	C	-0.44866783		
BBB+	1.40504026	B	-0.22289852	C-	-0.45513039		

Table 14: Asset return thresholds of GAZPROM

AA+	5.126972080	BBB	0.871458386	B-	-0.174787089	D	-0.416270428
AA	4.685954894	BBB-	0.283310372	CC+	-0.175234632		
AA-	4.410596120	BB+	0.121206467	CC	-0.219911776		
A+	3.619359286	BB	0.020215301	CC-	-0.393283229		
A	2.601748684	BB-	0.002508811	C+	-0.393283229		
A-	1.883635857	B+	-0.067047597	C	-0.393784496		
BBB+	1.419206221	B	-0.174787089	C-	-0.400065019		

Table 15: Asset return thresholds of NORNICKEL

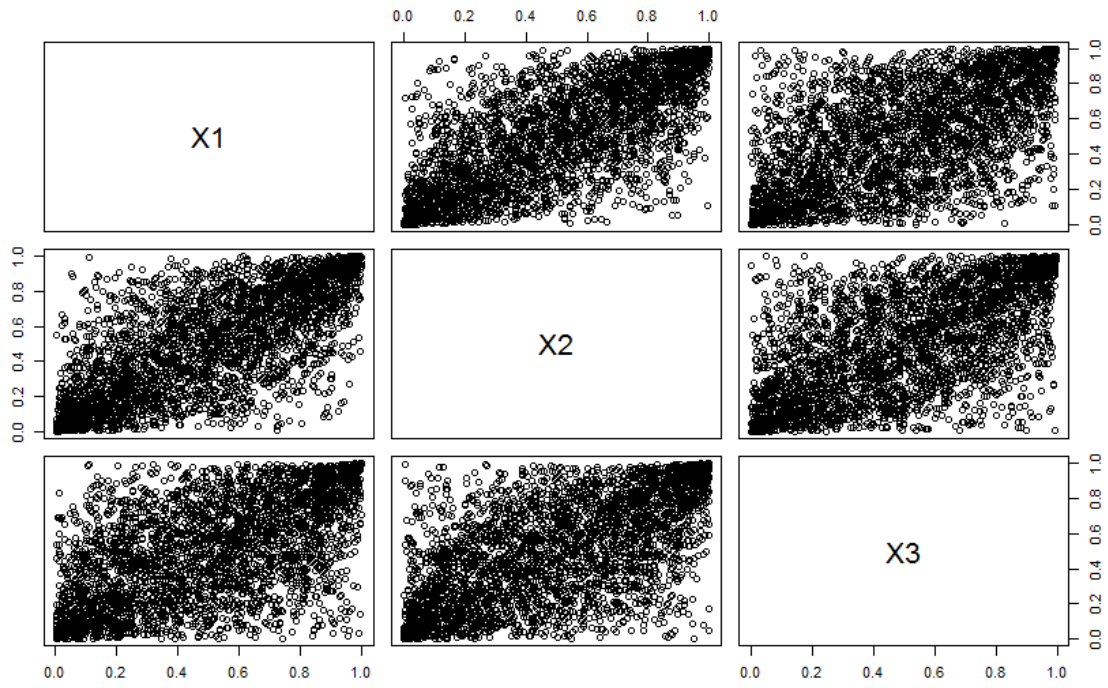


Figure 7: Simulated points from Joint distribution

R Script

```
data<- read.table("Returns2.txt",header=TRUE, sep=",")

mean(data$LUKOIL)
mean(data$GAZPROM)
mean(data$NORNICKEL)
n<- length(data$LUKOIL)
lretsL <- diff(log(data$LUKOIL), lag=1)
head(lretsL)
summary(lretsL)
lretsG <- diff(log(data$GAZPROM), lag=1)
summary(lretsG)
lretsN <- diff(log(data$NORNICKEL), lag=1)
summary(lretsN)

sd(lretsL)
sd(lretsG)
sd(lretsN)

library(e1071)

skewness(lretsL)
skewness(lretsG)
skewness(lretsN)
kurtosis(lretsL)
kurtosis(lretsG)
kurtosis(lretsN)

cor(lretsL , lretsG , method = "pearson")
cor(lretsL , lretsN , method = "pearson")
cor(lretsG , lretsN , method = "pearson")

par(mfrow=c(3 , 1))
plot.ts(lretsL)
plot.ts(lretsG)
plot.ts(lretsN)

par(mfrow=c(1 , 3))
hist(lretsL , nclass=40, freq=FALSE, main='Return_histogram')
curve(dnorm(x, mean=mean(lretsL), sd=sd(lretsL)), from= -0.3, to=0.2, add=TRUE,
col="red")
hist(lretsG , nclass=40, freq=FALSE, main='Return_histogram')
curve(dnorm(x, mean=mean(lretsG), sd=sd(lretsG)), from= -0.3, to=0.2, add=TRUE,
```

```

col="red")
hist(lretsN , nclass=40, freq=FALSE, main='Return_histogram')
curve(dnorm(x, mean=mean(lretsN), sd=sd(lretsN)), from= -0.3, to=0.2, add=TRUE,
col="red")
plot(density(lretsL), main= 'Return_empirical_distribution')
curve(dnorm(x, mean=mean(lretsL), sd=sd(lretsL)), from= -0.3, to=0.2, add=TRUE,
col="red")

#Jarque-Bera test for normality
library(normtest)
jb.norm.test(lretsL , nrepl=2000)
jb.norm.test(lretsG , nrepl=2000)
jb.norm.test(lretsN , nrepl=2000)

sqLretsL=lretsL*lretsL
sqLretsG=lretsG*lretsG
sqLretsN=lretsN*lretsN

#Ljung-Box test for serial independence
library(LSTS)\
Box.Ljung.Test(sqLretsL , lag = 24, main = NULL)
Box.Ljung.Test(sqLretsG , lag = 24, main = NULL)
Box.Ljung.Test(sqLretsN , lag = 24, main = NULL)
ts.diag(sqLretsL)
Box.test(sqLretsL , type="Ljung", lag=20, fitdf=0)
Box.test(sqLretsG , type="Ljung", lag=20, fitdf=0)
Box.test(sqLretsN , type="Ljung", lag=20, fitdf=0)

#ARIMA-GARCH model fitting
library(aTSA)
library(forecast)
library(rugarch)

L <- as.numeric(lretsL)
L <- L[!is.na(L)]

G <- as.numeric(lretsG)
G <- G[!is.na(G)]

N <- as.numeric(lretsN)
N <- N[!is.na(N)]

Lfinal.aic <- Inf
Lfinal.order <- c(0,0,0)
Gfinal.aic <- Inf

```



```

Gfinal.order <- c(0,0,0)
Nfinal.aic <- Inf
Nfinal.order <- c(0,0,0)

for (p in 1:4) for (d in 0:2) for (q in 1:4) {
  Lcurrent.aic <- AIC(arima(L, order=c(p, d, q), optim.control = list(maxit = 1000)))
  Gcurrent.aic <- AIC(arima(G, order=c(p, d, q), optim.control = list(maxit = 1000)))
  Ncurrent.aic <- AIC(arima(N, order=c(p, d, q), optim.control = list(maxit = 1000)))
  if (Lcurrent.aic < Lfinal.aic) {
    Lfinal.aic <- Lcurrent.aic
    Lfinal.order <- c(p, d, q)
    Lfinal.arima <- arima(L, order=Lfinal.order)
  }
  if (Gcurrent.aic < Gfinal.aic) {
    Gfinal.aic <- Gcurrent.aic
    Gfinal.order <- c(p, d, q)
    Gfinal.arima <- arima(G, order=Gfinal.order)
  }
  if (Ncurrent.aic < Nfinal.aic) {
    Nfinal.aic <- Ncurrent.aic
    Nfinal.order <- c(p, d, q)
    Nfinal.arima <- arima(N, order=Nfinal.order)
  }
}
}
Lfinal.order
Gfinal.order
Nfinal.order

par(mfrow=c(2, 2))
acf(resid(Lfinal.arima))
acf(resid(Lfinal.arima)^2)
pacf(resid(Lfinal.arima))
pacf(resid(Lfinal.arima)^2)

acf(resid(Gfinal.arima))
acf(resid(Gfinal.arima)^2)
pacf(resid(Gfinal.arima))
pacf(resid(Gfinal.arima)^2)

acf(resid(Nfinal.arima))
acf(resid(Nfinal.arima)^2)
pacf(resid(Nfinal.arima))
pacf(resid(Nfinal.arima)^2)

library(tseries)

```

```
L.garch <- garch(L ,trace=F)
L.res <- L.garch$res[-1]
summary(L.garch)
```

```
G.garch <- garch(G, trace=F)
G.res <- G.garch$res[-1]
summary(G.garch)
```

```
N.garch <- garch(N, trace=F)
N.res <- N.garch$res[-1]
summary(N.garch)
```

```
acf(L.res)
acf(L.res^2)
pacf(L.res)
pacf(L.res^2)
```

```
acf(G.res)
acf(G.res^2)
pacf(G.res)
pacf(G.res^2)
```

```
acf(N.res)
acf(N.res^2)
pacf(N.res)
pacf(N.res^2)
```

```
Box.test(L.res , lag=20, type="Ljung-Box")
Box.test(G.res , lag=20, type="Ljung-Box")
Box.test(N.res , lag=20, type="Ljung-Box")
```

```
#Fitting distributions to residuals
```

```
library(fitdistrplus)
```

```
library(metRology)
```

```
L.res <- L.res[!is.na(L.res)]
Lnorm <- fitdist(L.res , dist = "norm")
summary(Lnorm)
```

```
Lt <- fitdist(L.res , dist = "t.scaled", start = list(df = 3, mean=mean(L.res),
sd=sd(L.res)))
summary(Lt)
plot(Lt)
```

```
G.res <- G.res[!is.na(G.res)]
Gnorm <- fitdist(G.res , dist = "norm")
```

```
summary(Gnorm)
```

```
Gt <- fitdist(G.res, dist = "t.scaled", start=list(df = 3, mean=mean(G.res),  
sd=sd(G.res)))
```

```
summary(Gt)
```

```
plot(Gt)
```

```
N.res <- N.res[!is.na(N.res)]
```

```
Nnorm <- fitdist(N.res, dist = "norm")
```

```
summary(Nnorm)
```

```
Nt <- fitdist(N.res, dist = "t.scaled", start=list(df = 3, mean=mean(N.res),  
sd=sd(N.res)))
```

```
summary(Nt)
```

```
plot(Nt)
```

```
#Importing Credit Migration matrix and calculating thresholds of asset returns  
data2<-read.table("transition.txt",header=TRUE, sep=",")
```

```
thr<-function(d,m,s,vector){
```

```
  Def=qt.scaled(vector[1],df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  Cm= qt.scaled(vector[2]+vector[1],df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE)
```

```
  C= qt.scaled(vector[3]+pt.scaled(Cm,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  Cp=qt.scaled(vector[4]+pt.scaled(C,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  CCm=qt.scaled(vector[5]+pt.scaled(Cp,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  CC=qt.scaled(vector[6]+pt.scaled(CCm,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  CCp=qt.scaled(vector[7]+pt.scaled(CC,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  Bm=qt.scaled(vector[8]+pt.scaled(CCp,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  B=qt.scaled(vector[9]+pt.scaled(Bm,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  Bp=qt.scaled(vector[10]+pt.scaled(B,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  BBm=qt.scaled(vector[11]+pt.scaled(Bp,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  BB=qt.scaled(vector[12]+pt.scaled(BBm,df=d,mean = m, sd = s,lower.tail = TRUE,  
  log.p = FALSE),df=d,mean = m, sd = s,lower.tail = TRUE, log.p = FALSE)
```

```
  BBp=qt.scaled(vector[13]+pt.scaled(BB,df=d,mean = m, sd = s,lower.tail = TRUE,
```

```

log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
BBBm=qt.scaled(vector[14]+pt.scaled(BBp, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
BBB=qt.scaled(vector[15]+pt.scaled(BBBm, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
BBBp=qt.scaled(vector[16]+pt.scaled(BBB, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
Am=qt.scaled(vector[17]+pt.scaled(BBBp, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
A=qt.scaled(vector[18]+pt.scaled(Am, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
Ap=qt.scaled(vector[19]+pt.scaled(A, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
AAm=qt.scaled(vector[20]+pt.scaled(Ap, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
AA=qt.scaled(vector[21]+pt.scaled(AAm, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
AAp=qt.scaled(vector[22]+pt.scaled(AA, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
AAA=qt.scaled(vector[23]+pt.scaled(AAp, df=d, mean = m, sd = s, lower.tail = TRUE,
log.p = FALSE), df=d, mean = m, sd = s, lower.tail = TRUE, log.p = FALSE)
t<-c(Def, Cm, C, Cp, CCm, CC, CCp, Bm, B, Bp, BBm, BB, BBp, BBBm, BBB, BBBp, Am, A, Ap, AAm, AA, AAp, AAA)
}
thresholdsL <- thr(6.53046942, 0.01954184, 0.83277494, data2$BBB)
thresholdsG <- thr(7.653713749, -0.008275583, 0.861420330, data2$BBBm)
thresholdsN <- thr(6.57588223, 0.03295066, 0.82908809, data2$BBBm)

```

```

thresholdsL
thresholdsG
thresholdsN

```

```

thrnorm<-function(m, s, vector){
  Def = qnorm(vector[1], mean = m, sd = s)
  Cm = qnorm(vector[2]+vector[1], mean = m, sd = s)
  C = qnorm(vector[3]+pnorm(Cm, mean = m, sd=s), mean = m, sd = s)
  Cp = qnorm(vector[4]+pnorm(C, mean = m, sd=s), mean = m, sd = s)
  CCm = qnorm(vector[5]+pnorm(Cp, mean = m, sd=s), mean = m, sd = s)
  CC = qnorm(vector[6]+pnorm(CCm, mean = m, sd=s), mean = m, sd = s)
  CCp = qnorm(vector[7]+pnorm(CC, mean = m, sd=s), mean = m, sd = s)
  Bm = qnorm(vector[8]+pnorm(CCp, mean = m, sd=s), mean = m, sd = s)
  B = qnorm(vector[9]+pnorm(Bm, mean = m, sd=s), mean = m, sd = s)
  Bp = qnorm(vector[10]+pnorm(B, mean = m, sd=s), mean = m, sd = s)
  BBm = qnorm(vector[11]+pnorm(Bp, mean = m, sd=s), mean = m, sd = s)
  BB = qnorm(vector[12]+pnorm(BBm, mean = m, sd=s), mean = m, sd = s)
  BBp = qnorm(vector[13]+pnorm(BB, mean = m, sd=s), mean = m, sd = s)
  BBBm = qnorm(vector[14]+pnorm(BBp, mean = m, sd=s), mean = m, sd = s)
}

```

```

BBB = qnorm(vector[15]+pnorm(BBBm, mean = m, sd=s), mean = m, sd = s)
BBBp = qnorm(vector[16]+pnorm(BBB, mean = m, sd=s), mean = m, sd = s)
Am = qnorm(vector[17]+pnorm(BBBp, mean = m, sd=s), mean = m, sd = s)
A = qnorm(vector[18]+pnorm(Am, mean = m, sd=s), mean = m, sd = s)
Ap = qnorm(vector[19]+pnorm(A, mean = m, sd=s), mean = m, sd = s)
AAm = qnorm(vector[20]+pnorm(Ap, mean = m, sd=s), mean = m, sd = s)
AA = qnorm(vector[21]+pnorm(AAm, mean = m, sd=s), mean = m, sd = s)
AAp = qnorm(vector[22]+pnorm(AA, mean = m, sd=s), mean = m, sd = s)
n<-c(Def, Cm, C, Cp, CCm, CC, CCp, Bm, B, Bp, BBm, BB, BBp, BBBm, BBB, BBBp, Am, A, Ap, AAm, AA, AAp)
}
nthresholdsL <- thnorm(0.002459044, 1.000851347, data2$BBB)
nthresholdsG <- thnorm(-0.01254965, 1.00089028, data2$BBBm)
nthresholdsN <- thnorm(0.01349338, 1.00410348, data2$BBBm)

nthresholdsL
nthresholdsG
nthresholdsN

#Obtaining uniform distributions by means of t distributions found earlier
uL<-pt.scaled(L.res, df=6.53046942, mean = 0.01954184, sd = 0.83277494, ncp=0,
lower.tail = TRUE, log.p = FALSE)
uG<-pt.scaled(G.res, df=7.653713749, mean = -0.008275583, sd = 0.861420330, ncp=0,
lower.tail = TRUE, log.p = FALSE)
uN<-pt.scaled(N.res, df=6.57588223, mean = 0.03295066, sd = 0.82908809, ncp=0,
lower.tail = TRUE, log.p = FALSE)

#Pair-copula construction. For reproducibility randomly generated values from joint
distribution were saved in file simulatedvalues
library(copula)
library(VineCopula)
library(CDVineCopulaConditional)
library(scatterplot3d)

length(uL)
length(uG)
length(uN)

margins <- cbind(uL, uG, uN)
RVM <- CDVineCondFit(margins, Nx=1, treecrit="BIC", type="1", selectioncrit="BIC")
summary(RVM)
#d=dim(RVM$Matrix)[1]
#cond1 <- margins[,RVM$Matrix[(d+1)-1,(d+1)-1]]

#Sim <- CDVineCondSim(RVM, cond1)
#Sim <- data.frame(Sim)
#overplot(Sim, margins)

```

```

Sim <- read.table("Simulatedvalues.txt",header=TRUE, sep=",")
plot(Sim)
par(mfrow = c(1, 1))
scatterplot3d(Sim)

#Transforming margins back to t distributions. Based on asset return value assigning
a company the credit rating based on thresholds
tL<-qt.scaled(Sim$X1, df=6.53046942, mean = 0.01954184, sd = 0.83277494, ncp=0,
lower.tail = TRUE, log.p = FALSE)
tG<-qt.scaled(Sim$X2, df=7.653713749, mean = -0.008275583, sd = 0.861420330, ncp=0,
lower.tail = TRUE, log.p = FALSE)
tN<-qt.scaled(Sim$X3, df=6.57588223, mean = 0.03295066, sd = 0.82908809, ncp=0,
lower.tail = TRUE, log.p = FALSE)
tL
temp<-c()
temp
credrat <- function(values , thresh){

for (i in 1:length(values)) {
  if (values[i]<=thresh[1]){
    temp[i]="Default"
  }
  if((values[i]>thresh[1]) & (values[i]<=thresh[2])){
    temp[i]="C-"
  }
  if((values[i]>thresh[2]) & (values[i]<=thresh[3])){
    temp[i]="C"
  }
  if((values[i]>thresh[3]) & (values[i]<=thresh[4])){
    temp[i]="C+"
  }
  if((values[i]>thresh[4]) & (values[i]<=thresh[5])){
    temp[i]="CC-"
  }
  if((values[i]>thresh[5]) & (values[i]<=thresh[6])){
    temp[i]="CC"
  }
  if((values[i]>thresh[6]) & (values[i]<=thresh[7])){
    temp[i]="CC+"
  }
  if((values[i]>thresh[7]) & (values[i]<=thresh[8])){
    temp[i]="B-"
  }
}
}

```

```

if(( values [ i ] > thresh [ 8 ] ) & ( values [ i ] <= thresh [ 9 ] )){
    temp [ i ] = "B"
}
if(( values [ i ] > thresh [ 9 ] ) & ( values [ i ] <= thresh [ 10 ] )){
    temp [ i ] = "B+"
}
if(( values [ i ] > thresh [ 10 ] ) & ( values [ i ] <= thresh [ 11 ] )){
    temp [ i ] = "BB-"
}
if(( values [ i ] > thresh [ 11 ] ) & ( values [ i ] <= thresh [ 12 ] )){
    temp [ i ] = "BB"
}
if(( values [ i ] > thresh [ 12 ] ) & ( values [ i ] <= thresh [ 13 ] )){
    temp [ i ] = "BB+"
}
if(( values [ i ] > thresh [ 13 ] ) & ( values [ i ] <= thresh [ 14 ] )){
    temp [ i ] = "BBB-"
}
if(( values [ i ] > thresh [ 14 ] ) & ( values [ i ] <= thresh [ 15 ] )){
    temp [ i ] = "BBB"
}
if(( values [ i ] > thresh [ 15 ] ) & ( values [ i ] <= thresh [ 16 ] )){
    temp [ i ] = "BBB+"
}
if(( values [ i ] > thresh [ 16 ] ) & ( values [ i ] <= thresh [ 17 ] )){
    temp [ i ] = "A-"
}
if(( values [ i ] > thresh [ 17 ] ) & ( values [ i ] <= thresh [ 18 ] )){
    temp [ i ] = "A"
}
if(( values [ i ] > thresh [ 18 ] ) & ( values [ i ] <= thresh [ 19 ] )){
    temp [ i ] = "A+"
}
if(( values [ i ] > thresh [ 19 ] ) & ( values [ i ] <= thresh [ 20 ] )){
    temp [ i ] = "AA-"
}
if(( values [ i ] > thresh [ 20 ] ) & ( values [ i ] <= thresh [ 21 ] )){
    temp [ i ] = "AA"
}
if(( values [ i ] > thresh [ 21 ] ) & ( values [ i ] <= thresh [ 22 ] )){
    temp [ i ] = "AA+"
}
if( values [ i ] > thresh [ 22 ] ){
    temp [ i ] = "AAA"
}
}

```

```

temp
}
ratingsL<- credrat(tL, thresholdsL)
ratingsG<- credrat(tG, thresholdsG)
ratingsN<- credrat(tN, thresholdsN)

ratings <- cbind(ratingsL, ratingsG, ratingsN)
ratings

#For reproducibility forward rates were written down in the file forwards.
fwd<-read.table("forwards.txt",header=TRUE, sep=",")
#function LUKOIL calculates present value of Lukoil bond
LUKOIL <- function(f, c){
}
PVL<- LUKOIL(1000,47.5)
#Function GN calculates present values of Gazprom and Nornickel bonds, since they
both have 3 years till maturity
GN <- function(f, c){
  PV <- c()
  PV[1]=(c+c/(1+fwd$AAA[2])+(c+f)/((1+fwd$AAA[3])*(1+fwd$AAA[2]))) / (1+0.0736)
  PV[2]=(c+c/(1+fwd$AAp[2])+(c+f)/((1+fwd$AAp[3])*(1+fwd$AAp[2]))) / (1+0.0736)
  PV[3]=(c+c/(1+fwd$AA[2])+(c+f)/((1+fwd$AA[3])*(1+fwd$AA[2]))) / (1+0.0736)
  PV[4]=(c+c/(1+fwd$AAm[2])+(c+f)/((1+fwd$AAm[3])*(1+fwd$AAm[2]))) / (1+0.0736)
  PV[5]=(c+c/(1+fwd$Ap[2])+(c+f)/((1+fwd$Ap[3])*(1+fwd$Ap[2]))) / (1+0.0736)
  PV[6]=(c+c/(1+fwd$A[2])+(c+f)/((1+fwd$A[3])*(1+fwd$A[2]))) / (1+0.0736)
  PV[7]=(c+c/(1+fwd$Am[2])+(c+f)/((1+fwd$Am[3])*(1+fwd$Am[2]))) / (1+0.0736)
  PV[8]=(c+c/(1+fwd$BBBp[2])+(c+f)/((1+fwd$BBBp[3])*(1+fwd$BBBp[2]))) / (1+0.0736)
  PV[9]=(c+c/(1+fwd$BBB[2])+(c+f)/((1+fwd$BBB[3])*(1+fwd$BBB[2]))) / (1+0.0736)
  PV[10]=(c+c/(1+fwd$BBBm[2])+(c+f)/((1+fwd$BBBm[3])*(1+fwd$BBBm[2]))) / (1+0.0736)
  PV[11]=(c+c/(1+fwd$BBp[2])+(c+f)/((1+fwd$BBp[3])*(1+fwd$BBp[2]))) / (1+0.0736)
  PV[12]=(c+c/(1+fwd$BB[2])+(c+f)/((1+fwd$BB[3])*(1+fwd$BB[2]))) / (1+0.0736)
  PV[13]=(c+c/(1+fwd$BBm[2])+(c+f)/((1+fwd$BBm[3])*(1+fwd$BBm[2]))) / (1+0.0736)
  PV[14]=(c+c/(1+fwd$Bp[2])+(c+f)/((1+fwd$Bp[3])*(1+fwd$Bp[2]))) / (1+0.0736)
  PV[15]=(c+c/(1+fwd$B[2])+(c+f)/((1+fwd$B[3])*(1+fwd$B[2]))) / (1+0.0736)
  PV[16]=(c+c/(1+fwd$Bm[2])+(c+f)/((1+fwd$Bm[3])*(1+fwd$Bm[2]))) / (1+0.0736)
  PV[17]=(c+c/(1+fwd$CCp[2])+(c+f)/((1+fwd$CCp[3])*(1+fwd$CCp[2]))) / (1+0.0736)
  PV[18]=(c+c/(1+fwd$CC[2])+(c+f)/((1+fwd$CC[3])*(1+fwd$CC[2]))) / (1+0.0736)
  PV[19]=(c+c/(1+fwd$CCm[2])+(c+f)/((1+fwd$CCm[3])*(1+fwd$CCm[2]))) / (1+0.0736)
  PV[20]=(c+c/(1+fwd$Cp[2])+(c+f)/((1+fwd$Cp[3])*(1+fwd$Cp[2]))) / (1+0.0736)
  PV[21]=(c+c/(1+fwd$C[2])+(c+f)/((1+fwd$C[3])*(1+fwd$C[2]))) / (1+0.0736)
  PV[22]=(c+c/(1+fwd$Cm[2])+(c+f)/((1+fwd$Cm[3])*(1+fwd$Cm[2]))) / (1+0.0736)
  PV
}
PVG<- GN(1000, 49.5)
PVG
PVN<-GN(1000,66.25)

```


PVN

```
PV<-cbind(PVL,PVG,PVN)
```

PV

```
#write.table(PV, file = "Revaluation.txt", sep = ",")
PV<-read.table("Revaluation.txt",header=TRUE, sep=",")
PVL<-PV$PVL
PVG<-PV$PVG
PVN<-PV$PVN
PV
```

```
#valueL<-c()
#valueG<-c()
#valueN<-c()
#ratingsL[1]
#for(i in 1:length(ratingsL)){
  if(ratingsL[i]=="AAA"){
    valueL[i]=PVL[1]
  }
  if(ratingsL[i]=="AA+"){
    valueL[i]=PVL[2]
  }
  if(ratingsL[i]=="AA"){
    valueL[i]=PVL[3]
  }
  if(ratingsL[i]=="AA-"){
    valueL[i]=PVL[4]
  }
  if(ratingsL[i]=="A+"){
    valueL[i]=PVL[5]
  }
  if(ratingsL[i]=="A"){
    valueL[i]=PVL[6]
  }
  if(ratingsL[i]=="A-"){
    valueL[i]=PVL[7]
  }
  if(ratingsL[i]=="BBB+"){
    valueL[i]=PVL[8]
  }
  if(ratingsL[i]=="BBB"){
    valueL[i]=PVL[9]
  }
  if(ratingsL[i]=="BBB-"){
    valueL[i]=PVL[10]
  }
}
```

```

}
if (ratingsL [ i ] == "BB+") {
    valueL [ i ] = PVL [ 11 ]
}
if (ratingsL [ i ] == "BB") {
    valueL [ i ] = PVL [ 12 ]
}
if (ratingsL [ i ] == "BB-") {
    valueL [ i ] = PVL [ 13 ]
}
if (ratingsL [ i ] == "B+") {
    valueL [ i ] = PVL [ 14 ]
}
if (ratingsL [ i ] == "B") {
    valueL [ i ] = PVL [ 15 ]
}
if (ratingsL [ i ] == "B-") {
    valueL [ i ] = PVL [ 16 ]
}
if (ratingsL [ i ] == "CC+") {
    valueL [ i ] = PVL [ 17 ]
}
if (ratingsL [ i ] == "CC") {
    valueL [ i ] = PVL [ 18 ]
}
if (ratingsL [ i ] == "CC-") {
    valueL [ i ] = PVL [ 19 ]
}
if (ratingsL [ i ] == "C+") {
    valueL [ i ] = PVL [ 20 ]
}
if (ratingsL [ i ] == "C") {
    valueL [ i ] = PVL [ 21 ]
}
if (ratingsL [ i ] == "C-") {
    valueL [ i ] = PVL [ 22 ]
}
if (ratingsL [ i ] == "Default") {
    valueL [ i ] = 1000 * rbeta ( 1 , 0.479 , 0.127 )
}
if (ratingsG [ i ] == "AAA") {
    valueG [ i ] = PVG [ 1 ]
}
if (ratingsG [ i ] == "AA+") {
    valueG [ i ] = PVG [ 2 ]
}
}

```

```

if (ratingsG [ i ] == "AA") {
    valueG [ i ] = PVG [ 3 ]
}
if (ratingsG [ i ] == "AA-") {
    valueG [ i ] = PVG [ 4 ]
}
if (ratingsG [ i ] == "A+") {
    valueG [ i ] = PVG [ 5 ]
}
if (ratingsG [ i ] == "A") {
    valueG [ i ] = PVG [ 6 ]
}
if (ratingsG [ i ] == "A-") {
    valueG [ i ] = PVG [ 7 ]
}
if (ratingsG [ i ] == "BBB+") {
    valueG [ i ] = PVG [ 8 ]
}
if (ratingsG [ i ] == "BBB") {
    valueG [ i ] = PVG [ 9 ]
}
if (ratingsG [ i ] == "BBB-") {
    valueG [ i ] = PVG [ 10 ]
}
if (ratingsG [ i ] == "BB+") {
    valueG [ i ] = PVG [ 11 ]
}
if (ratingsG [ i ] == "BB") {
    valueG [ i ] = PVG [ 12 ]
}
if (ratingsG [ i ] == "BB-") {
    valueG [ i ] = PVG [ 13 ]
}
if (ratingsG [ i ] == "B+") {
    valueG [ i ] = PVG [ 14 ]
}
if (ratingsG [ i ] == "B") {
    valueG [ i ] = PVG [ 15 ]
}
if (ratingsG [ i ] == "B-") {
    valueG [ i ] = PVG [ 16 ]
}
if (ratingsG [ i ] == "CC+") {
    valueG [ i ] = PVG [ 17 ]
}
if (ratingsG [ i ] == "CC") {

```

```

    valueG [ i ] = PVG [ 18 ]
}
if ( ratingsG [ i ] == "CC-") {
    valueG [ i ] = PVG [ 19 ]
}
if ( ratingsG [ i ] == "C+") {
    valueG [ i ] = PVG [ 20 ]
}
if ( ratingsG [ i ] == "C") {
    valueG [ i ] = PVG [ 21 ]
}
if ( ratingsG [ i ] == "C-") {
    valueG [ i ] = PVG [ 22 ]
}
if ( ratingsG [ i ] == "Default") {
    valueG [ i ] = 1000 * rbeta ( 1 , 0.479 , 0.127 )
}
if ( ratingsN [ i ] == "AAA") {
    valueN [ i ] = PVN [ 1 ]
}
if ( ratingsN [ i ] == "AA+") {
    valueN [ i ] = PVN [ 2 ]
}
if ( ratingsN [ i ] == "AA") {
    valueN [ i ] = PVN [ 3 ]
}
if ( ratingsN [ i ] == "AA-") {
    valueN [ i ] = PVN [ 4 ]
}
if ( ratingsN [ i ] == "A+") {
    valueN [ i ] = PVN [ 5 ]
}
if ( ratingsN [ i ] == "A") {
    valueN [ i ] = PVN [ 6 ]
}
if ( ratingsN [ i ] == "A-") {
    valueN [ i ] = PVN [ 7 ]
}
if ( ratingsN [ i ] == "BBB+") {
    valueN [ i ] = PVN [ 8 ]
}
if ( ratingsN [ i ] == "BBB") {
    valueN [ i ] = PVN [ 9 ]
}
if ( ratingsN [ i ] == "BBB-") {
    valueN [ i ] = PVN [ 10 ]
}

```

```

}
if (ratingsN[i]==”BB+”){
  valueN[i]=PVN[11]
}
if (ratingsN[i]==”BB”){
  valueN[i]=PVN[12]
}
if (ratingsN[i]==”BB-”){
  valueN[i]=PVN[13]
}
if (ratingsN[i]==”B+”){
  valueN[i]=PVN[14]
}
if (ratingsN[i]==”B”){
  valueN[i]=PVN[15]
}
if (ratingsN[i]==”B-”){
  valueN[i]=PVN[16]
}
if (ratingsN[i]==”CC+”){
  valueN[i]=PVN[17]
}
if (ratingsN[i]==”CC”){
  valueN[i]=PVN[18]
}
if (ratingsN[i]==”CC-”){
  valueN[i]=PVN[19]
}
if (ratingsN[i]==”C+”){
  valueN[i]=PVN[20]
}
if (ratingsN[i]==”C”){
  valueN[i]=PVN[21]
}
if (ratingsN[i]==”C-”){
  valueN[i]=PVN[22]
}
if (ratingsN[i]==”Default”){
  valueN[i]=1000*rbeta(1,0.479, 0.127)
}
}
#values<-cbind(valueL, valueG, valueN)
#values
#write.table(values, file = ”values.txt”, sep = ”,”)
values <- read.table(”values.txt”,header=TRUE, sep=”,”)
valueL<-values$valueL

```

```

valueG<-values$valueG
valueN<-values$valueN

#Portfolio values are calculated
portval<-c()
for(i in 1:length(valueL)){
  portval[i]=valueL[i]+valueG[i]+valueN[i]
}
portval
hist(portval, nclass=40, freq=FALSE, main='Possible Portfolio values histogram')

#VaR percentile is calculated

sortedportval<-sort(portval, decreasing = FALSE)
sortedportval

VaR95=sortedportval[(1-0.95)*length(sortedportval)]
VaR99=sortedportval[(1-0.99)*length(sortedportval)]
VaR90=sortedportval[(1-0.9)*length(sortedportval)]
VaR95
VaR99
VaR90

```