Copula-based High Dimensional Cross-market Dependence Modeling

Jia Xu, Wei Wei and Longbing Cao
Advanced Analytics Institute
University of Technology Sydney

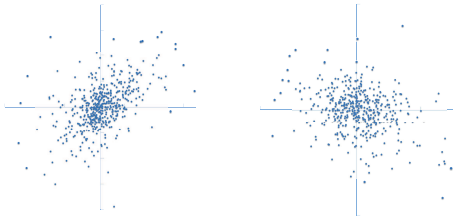
***Abstract*—Dependence across multiple financial markets, such as stock and foreign exchange rate markets, is high-dimensional, contains various relationships, and often presents complicated dependence structures and characteristics such as asymmetrical dependence. Modeling such dependence structures is very challenging. Although copula has been demonstrated to be effective in describing dependence between variables in recent studies, building effective dependence structures to address the above complexities significantly challenges existing copula models. In this paper, we propose a new D vine-based model with a bottom-up strategy to construct high-dimensional dependence structures. The new modeling outcomes are applied to trade 15 stock market indices and 10 currency rates over 16 years as a case study. Extensive experimental results show that this model and its intrinsic design significantly outperform typical models and industry baselines, as shown by the log-likelihood and Vuong test, and Value at Risk - a widely used industrial benchmark. Our model provides interpretable knowledge and profound insights into the high-dimensional dependence structures across data sources.**

I. Introduction

Dependence between financial markets (for short, cross-market dependence) has long been an issue of interest in both academia and industry. Effectively modeling cross-market dependence can contribute to significant consequences including the identification of opportunities for and barriers to international portfolio investment with important implications for portfolio allocation and asset pricing. In financial theory, if financial markets are not integrated, entailing differential investment and consumption opportunity sets across countries, investment barriers will affect investor portfolio choices and company financing decisions. Since exchange rates affect the cost of consumption across countries, as a result, exchange rate risk influences the price of assets to foreign investors.

Dependence between different countries can be easily seen from Figure 1. Figure 1(a) shows the daily returns between the United Kingdom comprehensive index FTSE100 and the United States comprehensive index S&P500, which indicates the strong positive correlation between them. Dependence between the foreign exchange rate GBP against the USD and the United Kingdom comprehensive index FTSE100 is shown in Fig. 1(b), which indicates negative dependence. These examples show that it is essential to realize that exchange rate markets significantly affect asset markets. It means that stock markets and exchange rate markets are dependent.

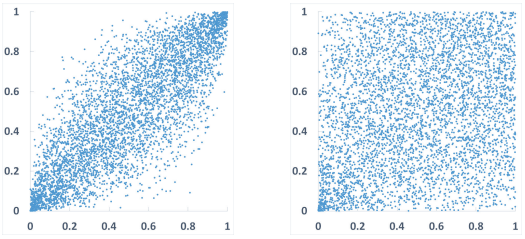
Modeling cross-market dependence involves the development of proper dependence structures. Typically, a low correlation coefficient between two markets implies a good opportunity for an investor to diversify investment and reduce risk. For example, suppose that the return in a domestic market and in a foreign market has a linear correlation coefficient of 0.2. Under the Gaussian assumption, the probability that the return in both markets is in their lowest 5^{th} percentile is less than 0.005. Thus, based on the Gaussian assumption, an investor can significantly reduce the investment risk in the domestic market by hedging in the foreign market. However, it has been widely observed that market crash and financial crisis often happen in different countries approximately around the same time period, even when the correlation between these markets is fairly low. Therefore, in cross-market studies, we have to consider not only the degree of dependence, but also the structure of dependence. The importance of dependence structure is demonstrated in Fig. 2. The two graphs in Fig. 2 present two different dependence structures with the same correlation.



(a) FTSE100 and S&P500 (b) FTSE100 and GBP

Fig. 1. Dependence across Markets

The challenge of modeling cross-market dependence lies in the three major aspects concerning us in this paper. Firstly, as with any complex behavioral and social system, the cross-market dependence structure is often embedded with strong couplings on high dimensionality [1]; the dependence across markets has been demonstrated to be significantly asymmetrical and nonlinear. For example, return in stock markets will have stronger correlation in a bear market downturn than in a bull market. Secondly, financial variables, such as daily return, have been shown to follow non-normal distributions, which means they do not follow the Gaussian assumption. For example, in [2], the empirical distribution of return from developed stock markets tends to display more kurtosis and have a pronounced higher peak than allowed under the normality hypothesis. The return on assets from emerging markets, however, is more volatile, and one can expect that it will be even more difficult to identify its distribution. This means



(a) Gaussian Copula (b) Clayton Copula

Fig. 2. The Scatter Plot of Return with Correlation 0.17

that different markets have different characteristics. Finally, an important issue in modeling cross-market dependence is high dimensional data. The corresponding models have to handle high dimensional financial variables (generally more than 20) that always lead over ten thousand features for a group of time series at one time window. However, it is difficult to deal with high dimensional variables due to the curse of dimensionality.

Dependence across markets has been studied by different communities, including statistics and machine learning. The typical approaches in the statistical community are joint distribution with Gaussian assumption and conditional correlation. The first method has been demonstrated that Gaussian assumption is inappropriate when studying either stock markets or exchange rate markets. The second one is to use conditional correlation to calculate the covariance, which is generally used in empirical studies. As the current correlation depends on previous one, the dependence structure is not flexible. The dependence studies in machine learning community consist of hidden Markov models and graphical probability models. The hidden Markov models, however, could have a large number of hidden states when applying to a high dimensional case, which invariably leads to computational intractability in the algorithms for inferring the hidden states from observations. The graphical probability models, such as Bayesian logic program [3], impose unrealistic assumptions in constructing dependence structures. As a result, they cannot capture the complex and asymmetrical dependence structures with high-dimensional variables [4].

The typical framework for dependence modeling is the copula-based models, such as [5], [6], [7], [8]. A copula-based model is a more convenient tool for studying dependence structures. A copula is a function that connects the marginal distributions to restore the joint distribution and various copula functions representing various dependence structures between variables. In a copula-based model, the primary task is to choose an appropriate copula function and a corresponding estimation procedure. Marginal distributions are treated as nuisance functions. This reorientation has desirable advantages in empirical finance where one of the primary goals is to investigate dependence in order to better understand issues like portfolio allocation and where the marginal distributions of asset return in individual markets may be very complicated and may not easily fit existing parametric models. Existing copula-based models, however, neither apply in high-dimensional cases [5], nor have the 'best' dependence structure to capture the asymmetrical and complex dependence structure across markets [9].

Since the existing copula models do not address all the challenges mentioned above, we here propose a new copula model: a Weighted Partial Vine Copula model (WPVC). WPVC is more powerful, because: (1) A partial D vine tree structure is created to capture the asymmetrical dependence across financial markets to construct complex and asymmetrical dependence structures. The advantage of the partial D vine dependence tree structure is that it can uniquely determine the correlation matrix and be algebraically independent, thus capable of handling the complex and asymmetrical dependence across markets. (2) Various time series models are used to capture the characteristics of different financial markets, which do not impose any Gaussian assumption on data. For example, we use AR(1)-GARCH(1,1) with skewed student t innovation to capture the volatility clustering of stock markets, and use ARMA(1,1)-GARCH(1,1) with inverse normal innovation to capture the fat tail of currency markets. (3) A truncated method replaces weak correlations in dependence structure with conditional independence but does not affect the structure, which thus effectively resolves the high dimensional issues.

The rest of this paper is organized as follows: Section 2 presents the related work. Section 3 introduces the basic concepts and the foundation of copula, the vine copula model and partial correlation. Sections 4 and 5 present WPVC, including D vine dependence structure building, bivariate copula selection, marginal distribution specification, and parameter estimation. The evaluation methods are discussed in Section 6 to verify the performance of high-dimensional financial variables. Section 7 shows the case study results. Section 8 concludes this work.

II. Related Work

Dependence across financial markets has been studied in the past decades. Several typical alternatives are available in multivariate analysis for studying dependence across markets in both statistics and machine learning communities. One approach is to use a joint distribution, typically the multivariate normal distribution. Under the Gaussian assumption, inference is then conducted based on the mean-variance analysis. However, there is increasing evidence indicating that the Gaussian assumption is inappropriate in the real world, as both stock and exchange rate markets face significant non-Gaussian characteristics [10]. Another issue of the joint distribution methods is that they only consider dependence, but ignore the dependence structure. As discussed in Section 1, we have to consider both the dependence degree and dependence structure. For example, given a copula, for the same correlation (dependence degree), it can show different dependence structures.

The second approach that has been used in empirical studies is to compute conditional correlations, such as the Dynamic Conditional Correlation (DCC) model [11]. It has been found that correlations computed with different conditions could differ dramatically. The correlations conditioned on large movements are higher than that conditional on small movements. The reason for this is that even a stationary Gaussian process predicts stronger dependence in volatile periods and weaker dependence in tranquil periods. Hence, the results are sometimes misleading and need to be interpreted carefully. In addition, another drawback of the DCC models is that if restriction on the covariance matrix (dependence structure) is not imposed, then the number of parameters are

huge. If restriction is imposed on the covariance matrix, then the structure is less flexible.

The third approach of modeling dependence is probabilistic graphical models, including the Bayesian logic program [12] and relational dependency networks [3]. These models generally build a graph to represent the conditional dependence structure between random variables. For example, the latent factor models with a dependency structure in the latent space were studied in [13]. A set of probabilistic dependencies were learned in [14] to identify the relationships between the headwords of each phrase. These models handle high-dimensional problems, and have the advantage of learning latent relationships from data. However, as discussed in [4], they tend to force the local quantitative part of the model to take a simple form such as the discretized form of the data, when a multivariate Gaussian or its mixtures cannot capture the data in the real world. This leads to the assumption that is difficult for these models to capture the complex and asymmetrical dependence in the high dimensional case.

Another alternative approach of modeling dependence in machine learning is the hidden Markov model (HMM) and its variants, which represent probabilistic distributions over sequences of observations and Bayesian networks. If HMM is used in the high dimensional case, its generalizations, such as factorial HMMs, tree structured HMMs, and switching state-space models, use richer hidden representations to model more interesting temporal relationships that cannot be captured by simple HMMs. However, having richer hidden state representations invariably leads to computational intractability in the algorithms when inferring hidden states from observations [15].

Finally, copula-based models have been used to directly model dependence, as a copula provides an effective tool for modeling and analyzing dependence structures between random variables. Copula-based dependence models are free of the linear correlation restriction, and allow dependence and correlation to vary over time. In addition to capturing dependence, a copula can build flexible structures to model complex high-dimensional dependency structures. The copula family consists of time-varying copula models [16], stochastic copula models [17], and vine copula models [18]. Time-varying copula models and stochastic copula models are incapable for dealing with high-dimensional variables due to the complexity of dependence structures and the curse of dimensionality. Existing vine-based copula models, such as canonical vine copula [9], has a strong assumption about dependence structures, leading to difficulty in capturing the complex dependences between markets. A copula Bayesian network was proposed in [19] to model high-dimensional continuous distributions. However, this model uses conditional independence to replace the dependence between random variables, which leads to the loss of dependence in structures. A copula-based model was introduced in [8] to measure the dependence between random variables, which is only applied to bivariables. In [6], a copula-based approach learns the copula distribution over the latent variables. This model uses the bivariate copula distribution, which means it can only apply to bivariables.

III. Preliminaries

A. Copula Theory

A copula has been proved to be a powerful tool in modeling the dependence of multivariables. It can capture complicated correlations between variables regardless of whether they are linear or non-linear. In terms of Sklar's theorem [20], every multivariate distribution F with marginal $F_1(x_1), \ldots, F_n(x_n)$ can be expressed with its marginal distribution as variables for a copula function. The definition of copula function is given by:

$$F_1(x_1, x_2, \ldots, x_n) = C(F_1(x_1), F_2(x_2), \ldots, F_n(x_n)) \quad (1)$$

for some appropriate n-dimensional copula C with uniformly distributed marginal $U(0, 1)$ on $[0, 1]$. F is the joint distribution of a random vector $x = [x_1, \ldots, x_n]$, $F_1(x_1), \ldots, F_n(x_n)$ are the marginal distribution of the corresponding variables respectively. If we define F_i^{-1} as the inverse distribution function of a marginal distribution F_i, $u = F_n(x_n)$ the copula from Equation (1) has the expression:

$$C(u_1, u_2, \ldots, u_n) = F(F_1^{-1}(u_1), F_2^{-1}(u_2), \ldots, F_n^{-1}(u_n)) \quad (2)$$

In many cases, it is easier to get a density function rather than a distribution function. We can join the density functions f by using the density function of copula C. If the multivariate distribution F is continuous, with strictly increasing and continuous marginal densities $F_1(x_1), \ldots, F_n(x_n)$, we have:

$$f(x_1, x_2, \ldots, x_n) = c(F_1(x_1), F_2(x_2), \ldots, F_n(x_n)) \prod_{i=1}^{n} f_i(x_i) \quad (3)$$

Here c is the density copula functions and $f_i(x_i)$ is the density function of $F_i(x_i)$. Under the copula theory framework, for any n-dimensional random vector $x = (x_1, \ldots, x_n)$, given any set of marginal distributions $F_1(x_1), \ldots, F_n(x_n)$ and the related copula function C, we can obtain the corresponding joint distribution function $F(x_1, x_2, \ldots, x_n)$ by using Equation (1). On the other hand, from Equation (2), we can infer that copula functions can model dependence from marginal distributions separately. This implies that the choice of a copula function does not depend on the choice of marginal distribution, and the marginal distributions for each variable do not have to be the same. The marginal distribution for each variable can be selected from mixed families. This property gives us a way to measure the joint distribution for multivariates by separately obtaining the marginal distribution and the corresponding copula function.

B. Vine Copula

Joe [21] introduced a method to model multivariate data by a cascade of simple building blocks. The modeling scheme is based on the decomposition of a multivariate density into a cascade of bivariate copulas, applying to the original variables and their conditional and unconditional distribution functions. It allows for different structural behaviors of bivariate variables to be modeled suitably, in particular with regard to their asymmetry, or strength of dependence. Based on the modeling scheme, a D vine structure is introduced by Kurowicka and Cooke [22]. In Equation (3), the multivariate copula functions $c(F_1(x_1), F_2(x_2), \ldots, F_n(x_n))$ can be split into multiple bivariate copulas, which is given as follows:

$$
\begin{aligned}
&c(F_1(x_1), F_2(x_2), \ldots, F_n(x_n)) \\
&= \prod_{k=1}^{n} f_k(x_k) \prod_{j=1}^{n-1} \prod_{i=1}^{n-j} c_{i,i+j|i+1,\ldots,i+j-1} \\
&(F(x_i|x_i, \ldots, x_{i+j-1}), F(x_{i+j}|x_{i+1}, \ldots, x_{i+j-1}))
\end{aligned} \tag{4}
$$

Fig. 3 shows an example of D vine structure of six variables. We can see that D vine belongs to graphical model, thus, we introduce a concept of graphical models, which are very important to understand how we construct the partial D vine structure in the following sections. Let V, T, E, N be denoted as vines, trees, edges and nodes respectively, according to Kurowicka and Cooke [22], a vine V is a nested set of connected trees $V = T_1, \ldots, T_{n-1}$. The edges of a tree T_j are the nodes from tree T_{j+1}, where $j = 1, \ldots, n-2$. The complete union, conditioning and conditioned sets for an edge are given as follows:

Definition1 (Complete Union, Conditioning and Conditioned Sets of an Edge). The complete union of an edge $e_i \in E_i$ is the set $U_{e_i} = \{n_1 \in N_1 \mid \exists e_j \in E_j, j = 1, 2, \ldots, i-1$ with $n_1 \in e_1 \in e_2 \in \ldots \in e_{i-1} \in e_i\} \subset N_1$. For $e_i = \{a, b\} \in E_i, a, b \in N_i, i = 1, 2, \ldots, n-1$, the conditioning set of an edge e_i is $D_{e_i} = U_a \cap U_b$, and the conditioned sets of an edge e_i are $C_{e_i,a} = U_a \setminus D_{e_i}, C_{e_i,b} = U_b \setminus D_{e_i}$ and $C_{e_i} = C_{e_i,a} \cup C_{e_i,b} = U_a \triangle U_b$, where $A \triangle B := (A \setminus B) \cup (B \setminus A)$ denotes the symmetric difference of two sets.

Hence, U_{ei} is a set of all nodes in N_i that are connected by the edges e_i. By definition, $U_{ei}(1) = e_i$. Then, the constraint set is defined:

Definition2 (Constraint Set). The constraint set for V is a set:
$CV = \{(\{C_{e_a}, C_{e_b}\}, D_e) \mid e \in E_i, e = \{a, b\}, i = 1, \ldots, n-1\}$

The edge e can be written as $\{Ce \mid De\}$, or $\{C_{e(a)}, C_{e(b)}|D_e, e = \{a, b\}\}$, where the conditioning set D_e is shown to the right of "|", and the conditioned set C_e to the left. $\{U_a \setminus D_e\}$ is the set which includes all variables in the set U_a, but excludes the variables in the set D_e.

Definition3 The m-child and m-descendant are defined as follows: (M-Child, M-Descendant). Let edge f be a variable of edge k, and f be a m-child of k. If f is reachable from k via the membership relation: $f \in f_1 \in \ldots \in k$, f is m-descendant of k.

Example1Fig. 3 shows a D vine structure with six variables. In Tree T_1, $N_1 = \{A, B, C, D, E, F\}$, $E_1 = \{AB, BC, CD, DE, EF\}$. Then, in Tree T_2, $N_2 = E_1$, and $E_2 = \{\{A, B\}, \{B, C\}; \{B, C\}, \{C, D\}; \{C, D\}, \{D, E\}; \{D, E\}, \{E, F\}\} = \{AC|B, BD|C, CE|F, DF/E\}$. For edge $e = AC|B$ in the tree T_2, the corresponding complete union are $U_a = \{A, B\}$ and $U_b = \{B, C\}$. The conditioning set is $D_e = \{A, B\} \cap \{B, C\} = \{B\}$. The conditioned set is $C_{e(a)} \cup C_{e(b)} = \{A, C\}$, where $C_{e(a)} = \{A, B\} \backslash \{B\} = \{A\}$ and $C_{e(b)} = \{B, C\} \backslash \{B\} = \{C\}$. The corresponding constraint set is $\{\{C_{e,a}, C_{e,b}\}, D_e\} = \{(\{A, C\}, B)\}$.

C. Partial Correlation

Partial correlation is another important concept for constructing the partial D vine structure. Bedford and Cooke [23] proposed a method to use partial correlation to measure and determine the vine structure. The definition of partial correlation is given by:

Definition4 Let $X_1, X_2, \ldots, X_n$ be random variables, the partial correlation of X_1 and X_2 given by $X_3, \ldots, X_n$ is:

$$
\rho_{1,2;3,\ldots,n} = \frac{\rho_{1,2;3,\ldots,n-1} - \rho_{1,n;3,\ldots,n-1} \cdot \rho_{2,n;3,\ldots,n-1}}{\sqrt{1 - \rho^2_{1,n;3,\ldots,n-1}} \cdot \sqrt{1 - \rho^2_{2,n;3,\ldots,n-1}}} \tag{5}
$$

An important property of partial correlation is that for elliptical distributions, partial correlation is equal to the corresponding conditional correlation, which has been proved by [23]. According to this property, we can use partial correlation instead of conditional correlation to measure the correlations on each node. On the other hand, it is easy to find out that the partial correlation ($\rho_{1,2}$) is equal to Kendall's tau ($\tau_{1,2}$). The partial correlation can be easily computed via correlation by iterating Equation (5).

IV. Weighted Partial Vine Copula

As highlighted in the introduction, the Weighted Partial Vine Copula (WPVC) is centered on the partial D vine structure. It is constructed by a large number of bivariate copulas, making it flexible and powerful for modeling the complex dependence structures of high-dimensional financial variables.

A. Partial D Vine Structure Construction

According to Definition 4, it is easy to find out that partial correlation can be calculated by using the original training data directly, without the dependence of the knowledge of assumed structure or bivariate copulas and the corresponding copula parameters. For the vine structure, a good one should own the strongest correlation on the first tree and the weakest correlation on the last tree. By employing partial correlation, we can employ a bottom-up strategy to construct the D vine structure, which ensures that the weakest correlation is always on the last tree and the strongest correlation is on the first tree. By using partial correlation to construct the D vine structure, every tree does not depend on the structure of the previous tree, which is more flexible.

Before building the partial D vine structure, we firstly review several important properties of D vine for vine structure construction (see details in [22]).

(1) There are $(j-1)$ and $(j+1)$ variables in the conditioning sets and constraint sets of an edge of the j^{th} tree respectively;
(2) If two or more nodes have the same constraint sets, they are the same node;
(3) If variable i is a member of the conditioned set of an edge e in a regular vine, then i is a member of the conditioned set of exactly one of the m-child of e, and the conditioning set of an m-child is a subset of De.

According to the above properties, two lemmas and one theorem are introduced to explain the procedure of partial D vine structure construction.

Lemma1 Let $A \subset \{1, \ldots, n\}$ and $x_1, x_2 \notin A, x_1 \neq x_2$ and $y_1, y_2 \notin A$, $N_1 = \{x_1, y_1 \mid A \setminus \{y_1\}\}$ and $N_2 = \{x_2, y_2 \mid A \setminus \{y_2\}\}$ be nodes of tree T_i of regular vine on n variables; then N_1 and N_2 have a common m-child. Moreover if $y_1 \neq y_2$, then this common m-child is $\{y_1, y_2 \mid A \setminus \{y_1, y_2\}\}$.

The proof of Lemma 1 can be referred to the relevant work in [24], which is omitted here.

Lemma2 For a regular vine on n variables, $j = 2, \ldots, n-1$, the edge e in T_j has only two constraint sets of m-children in T_{j-1}, which are indexed by different variables in a conditioned set.

Proof: Suppose there are three identical constraint sets indexed by different variables in a conditioned set. According to Property (2), nodes with the same constraint sets should be the same. Based on Property (3), the variables in the conditioned set is still in the conditioned set of its m-children. This means that the node has three variables in its conditioned set, which violates Property (1). Therefore, one edge has only two constraint sets which are indexed by different variables in a conditioned set. ■

According to the above Lemmas and properties of D vine, we derive a theorem, which is used to build the partial D vine structure.

Theorem1 Given a specific tree T_j in D vine V, all m-children of the nodes in the tree T_{j-1} (the tree above tree T_j) are identified except the two m-children on both sides.

Proof: Suppose there are three nodes A, B, C in tree T_j, and B is in the middle of A and C. The conditioning sets of A, B, C are C_a, C_b, C_c respectively, while the conditioned sets are D_a, D_b, D_c. In terms of Lemma 2, the two constraint sets of the m-children of node B are indexed by the two different variables in C_b. For the two different constraint sets, the two index variables from C_b are confirmed as belonging to the two conditioning sets, respectively, and the other variables of the two conditioning sets are from the conditioning sets of C_a and C_c according to the definition of D vine. Hence, there are two combinations for each conditioning set, and if the two combinations are all accepted for the structure, we can conclude that C_a is a subset of D_b, which implies nodes A and C are not m-children of the same node in terms of the path like structure, which conflicts with the postulated conditions we proposed at first. Therefore, C_a cannot be a subset of D_b, as they have only one mutual element, which is the other variable in the conditioning set of node B's corresponding m-child. A similar conclusion holds for the m-child of nodes B and C. Thus, m-children of node B are identified given the structure of chain A, B and C in tree T_j. For the nodes on two sides of tree T_j, they both have only one identified m-child with the node linked with them, and the other m-children of the two nodes do not have identified constraint sets. ■

Following Theorem 1, we can easily derive a corollary below:

Corollary1 For a D vine V specification from bottom up with n variables, there are $\lceil \frac{n}{2} \rceil$ steps to specify the vine structure and $\frac{n!}{2}$ possible combinations.

Proof: In terms of Theorem 1, for a D vine following a bottom-up specification, two variables are chosen in one node at one time. If the total number of variables is even, there are $\frac{n}{2}$ steps to choose all the variables. If the total number of variables is odd, $\lceil \frac{n}{2} \rceil$ steps are needed to choose all variables. Given a specific tree T_j, if the following tree T_{j+1} is identified, nodes in T_j have been the expected nodes on both sides. Suppose there are N variables in all and M variables have been chosen in the specification process from trees T_{N-1} to T_{j+1}, there are P^2_{N-M} possible permutations when constructing T_j. Particularly, the last tree T_{N-1} only includes two nodes, and there are C^2_N combinations instead. Thus, the total number of possible combinations for our bottom-up D vine is:

$$\begin{aligned} & C_n^2 \cdot P_{n-2}^2 \cdot P_{n-4}^2 \cdots P_2^2 \\ =& \frac{n(n-1)}{2} \cdot (n-2)(n-3) \cdot (n-4)(n-5) \cdots 2 \\ =& \frac{n(n-1)\cdots 2}{2} = \frac{n!}{2} \end{aligned} \tag{6}$$

which means the bottom-up vine construction methodology can cover all possible combinations. ■

The key step in constructing a D vine structure is to determine the nodes on both sides of each tree. To keep the weakest correlation on the bottom and the strongest correlation on the top, we propose a method to identify the appropriate nodes for each candidate combination. In this paper, a tree inverse level is applied to distinguish whether the weakest or strongest correlation is chosen and then denoted as k. Assume i is the level of the tree in a vine structure, for trees under the inverse level ($i \geqslant k$), the appropriate nodes must minimize the value of function $\sum |\rho_{c;d}|$. In addition, if trees are beyond the inverse level ($i < k$), the appropriate nodes must maximize the value of function $\sum ln(1 - \rho^2_{c;d})$.

Below we illustrate the construction of an optimal vine structure. Suppose there is one comprehensive index and six currencies which are denoted by A, B, C, D, E and F, and the tree inverse level k is equal to three. The D vine will consist of 5 trees and 20 nodes in both the D vine structures based on partial correlation and conditional copula. All the trees and nodes are shown in Fig. 3. Each node can be allocated to one bivariate copula or one partial correlation.

For these six variables, there are in total 20 partial correlations. The smallest absolute value of these partial correlations is chosen to be the edge in tree T_5. Suppose the selected partial correlation in T_5 is $\rho_{A,F;B,C,D,E}$, and the conditioned set and conditioning set are $\{A, F\}$ and $\{B, C, D, E\}$ respectively. The next step is to choose nodes for T_5. In terms of Lemma 2, the two variables in the conditioned set are allocated to different constraint sets with the conditioning set $\{B, C, D, E\}$. As a result, the constraint sets of the two nodes are $\{A, B, C, D, E\}$ and $\{F, B, C, D, E\}$. There are four partial correlations for each constraint set. Suppose the combination with the smallest absolute value of these partial correlations is $\rho_{A,E;B,C,D}$ and $\rho_{B,F;C,E,D}$, the nodes of T_5 are $\{\{A, E\}, \{B, C, D\}\}$ and $\{\{B, F\}, \{C, D, E\}\}$. Once T_5 is specified, according to Theorem 1 and Lemma 1, the node in the middle of T_4 is specified as well, which is $\{\{B, E\}, \{C, D\}\}$, and the constraint sets for the nodes on both sides of T_4 are

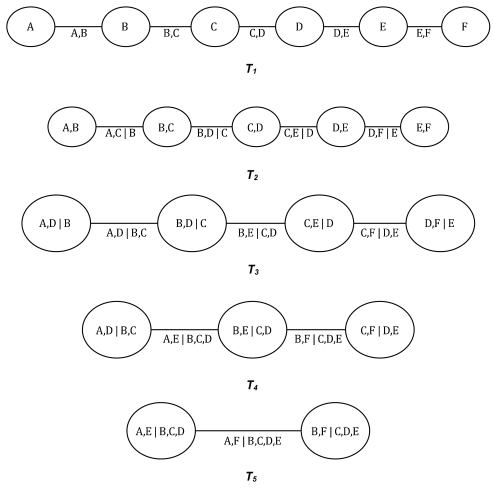


Fig. 3. D vine Trees

$\{A, B, C, D\}$ and $\{C, D, E, F\}$. If the combination with the smallest absolute partial correlation on these two constraint sets are $\rho_{A,D;B,C}$ and $\rho_{C,F;D,E}$, we get the two border nodes $\{\{A, D\}, \{B, C\}\}$ and $\{\{C, F\}, \{D, E\}\}$. For tree T_3, if as assumed the tree broken level is equal to three, we should choose the combination with the biggest absolute value rather than the smallest absolute value of partial correlations for border nodes. Since we only have six variables in this example, once the structures of T_5 to T_3 are specified, the whole D vine structure is specified.

As the parameter k can be chosen from $n-1$ to $\lceil \frac{n}{2} \rceil$, we can infer $\lceil \frac{n}{2} \rceil$ possible D vine structures depending on the different values of tree inverse value k.

After building the $\lceil \frac{n}{2} \rceil$ candidate D vines, the next step is to find the 'Best' D vine among these candidates. Wei [9] presented a method using the determinant of each partial correlation matrix to choose the best vine structure. However, this method only considers the strongest correlations on the top, while ignores the weak correlations at the bottom.

As discussed in Section 2, the 'Best' D vine structure will ensure the strongest correlations in the top tree and the weakest correlations in the bottom tree. If the structure is only selected with the maximum value of determinant of all corresponding partial correlation matrices, the bias will choose the structure with stronger correlations on the top. Giving a weight to each tree can enhance the influence of the trees on the top or at the bottom, and a balanced structure can be selected.

Here a classic model of gas particles in unit volume of altitude in the gravity field in physics is introduced to simulate the weight of each tree in the D vine structure. The distribution equation is given by:

$$n = n_0 e^{-\frac{mgH}{KT}} \tag{7}$$

where m, H are the mass and height of the particle respectively, g is gravitational acceleration, K is the Boltzmann Constant, and T is the current temperature.

Following the instructions in the last section, we assume each level is a unit height and the tree inverse level k is the zero potential energy level. Thus, the weight of each level will increase from level k to level $n-1$ and level one. Since the parameters m, g, K and T are constants in a given environment, Equation (7) can be simplified as follows:

$$W = e^{-m_0 h} \tag{8}$$

where m_0 is the parameter and h is the distance from the level of each tree to the tree inverse level. To restrict the value of weight for each level in interval $[0, 1]$, we standardize the weight:

$$W(h) = \begin{cases} 0.5 \times \frac{e^{-m_0(k-h)}}{\sum_{i=1}^{k} e^{-m_0(k-i)}}, & h \in [1, k]; \\ 0.5 \times \frac{e^{-m_0(h-k)}}{\sum_{i=k+1}^{N-1} e^{-m_0(i-k)}}, & h \in (k, N-1]. \end{cases} \tag{9}$$

where N is the number of variables, k is the tree inverse level, h is the level of a tree and m_0 is a parameter which falls in interval $[0, 1]$.

The 'Best' D vine structure maximizes the value of function $-ln(D)$, where D is the weighted determinant which is calculated by using:

$$D = \prod_{i,j} (1 - W_i \rho^2_{i,j;d(i,j)}) \tag{10}$$

where W_i is the corresponding weight and $d(i, j)$ is the conditioning set excluding variables i and j. The corresponding conditioned set is i and j.

Algorithm 1 D Vine Construction and Selection

Require: Observations of n input variables
1: Calculate all values of partial correlation, and then allocate the smallest absolute value of partial correlation to the node in T_{n-1} (T_{n-1} is the bottom tree).
2: **for** $k = 1, \ldots, n-2$ **do**
3: **for** $i = n-1, \ldots, \lceil \frac{n}{2} \rceil$ **do**
4: **if** $T_i > T_k$ **then**
5: Find variable combinations for nodes on both sides in tree T_i which can minimize the function $|\rho_{c:d}|$, where T_i indicates the ith tree and T_k is tree inverse level tree;
6: **else**
7: Find variable combinations for nodes on both sides in tree T_i which can minimize the function of $\sum ln(1 - \rho^2_{c:d})$
8: **end if**
9: **end for**
10: **end for**
11: There will be $n-2$ D-Vines as $k = 1, \ldots, n-2$. Calculate the function $-ln(D)$ of all of the D-Vines based on partial correlation, and choose the maximum value of the function as the 'Best' D vine. (D is calculated in Equation (10));
12: The 'Best' Weighted Partial D vine based on conditional copula corresponds to the D vine based on partial correlation;
13: **return** The Weighted Partial D vine dependence structure.

B. Dependence Structure Truncation

The number of parameters increases exponentially as the dimension increases. For example, a n-variable partial D vine structure with bivariate t copula (two-parametric copula) has $n(n-1)$ parameters. This may result in a huge computational burden and be time consuming. Hence, it is necessary to reduce the number of parameters by truncating the partial D vine structure. Since conditional independence copula is equal to 1 [5], we consider truncating the partial D vine structure by using

a conditional independence copula to replace those edges that have a low absolute value of partial correlation. This ensures that we retain the important dependence indicated by strong correlations, truncating the useless dependence indicated by weak correlations. In the truncation procedure, we replace those edges for which the absolute value of partial correlations is less than a specified truncation value ρ_{trun} between 0 and 1 (e.g., 0.1), with conditionally independent copula. Then, based on the specified truncation value ρ_{trun}, we can truncate the partial D vine. Once the partial D vine tree structure is identified, the next step is to select bivariate copulas for each edge in all trees.

C. Bivariate Copula Selection

As discussed in Section 3.3, partial correlation is equal to its corresponding conditional correlation for the elliptical family. This means that our partial regular vine tree structure is built based on an elliptical copula family (i.e., Gaussian or student t copulas). However, according to the following theorem, the limitation of partial correlation can be removed by mapping the partial regular vine tree structure to a typical regular vine via conditional correlation.

Theorem2 For any regular vine on n variables, there is one-to-one correspondence between the set of $n \times n$ positive definite correlation matrices and the set of partial correlation specification of the vine.

The proof of Theorem 2 can be referred to in [23], which is omitted here. It shows that there is a one-to-one relationship between the partial regular vine specification and the correlation matrix, which ensures that our WPVC can link to the typical conditional correlation based-regular vine tree structure. A large number of copula family candidates, bivariate copulas rather than the elliptical copula family can be chosen. Hence, the limitation can be removed when selecting bivariate copulas.

D. Marginal Model Specification

To capture the characteristics of stock and exchange rate markets, such as volatility and fat tail, various $ARMA$-$GARCH$ models are used as a marginal distribution in the WPVC model.

We use the $ARMA(1,1) - GARCH(1,1)$ model to capture the characteristics of comprehensive indices in stock markets and the $AR(1)-GARCH(1,1)$ for the exchange rate. Due to the page limitation, we illustrate the work via choosing the $ARMA(1,1)-GARCH(1,1)$ model, which is defined as follows:

$$X_t = C_0 + \delta_1 X_{t-1} + \gamma_1 \varepsilon_{t-1} + \varepsilon_t \tag{11}$$

$$\varepsilon_t = \sigma_t e_t \tag{12}$$

$$\sigma_t^2 = \omega + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \tag{13}$$

where $\omega > 0, \alpha_1, \beta_1 \geq 0, \alpha_1 + \beta_1 < 1$. X_t represents the actual return, ε_t is the residual error and σ_t is volatility of return on day t. Given the value of σ, it is obvious that a small value of σ_{t-1}^2 will result in a small value of σ_t^2 and a large value of σ_{t-1}^2 will result in a large value of σ_t^2.

V. Parameter Estimation

The maximum log-likelihood method is a typical method for estimating the parameters of a copula based on a vine structure. The log-likelihood function is given by:

$$L(\xi : x) = \sum_{j=1}^{n} \{ \sum_{i=1}^{p} ln f_i(x_{i,j}; \phi_i) + ln(c(F_1(x_1, n), \ldots, F_p(x_p, n); \theta)) \} \tag{14}$$

where $\xi = (\phi, \theta)$ is a vector covering all parameters of the marginal distributions $\phi = (\phi_1, \phi_2, \ldots, \phi_p)$ and the copula parameters θ.

Based on the definition of a copula, we can estimate the parameters by decomposing Equation (14) into two parts: marginal distribution log-likelihood function and copula log-likelihood function.

First, we need to estimate the parameters in marginal distributions. The marginal distribution log-likelihood is:

$$L_m(\phi : x) = \sum_{i=1}^{p} \sum_{j=1}^{n} ln f_i(x_{i,j}; \phi_i) \tag{15}$$

Thus, the parameters of the marginal distribution can be estimated by optimizing the marginal log-likelihood $L_m(\phi : x)$

$$\hat{\phi} = \underset{\phi}{\operatorname{argmax}} L_m(\phi : x) \tag{16}$$

Then, we can estimate the parameters in the copula by using the copula log-likelihood function

$$L_c(\theta; u, \phi) = \sum_{i=1}^{p} ln(c(F_1(x_1, n), \ldots, F_p(x_p, n); \theta)) \tag{17}$$

In this step, the parameters are estimated by optimizing the copula log-likelihood $L_c(\theta; u, \phi)$, which is conditional on the estimated parameters ϕ for marginal distributions

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} L_c(\theta; u, \phi) \tag{18}$$

VI. Value at Risk for Evaluating Business Performance

To evaluate the model's business performance, we use the Value at Risk (VaR), which is a probabilistic metric measuring market risk and an industrial golden benchmark for measuring market risk. VaR at the level $(1 - \alpha)$ is defined by:

$$VaR_t(1 - \alpha) = -inf\{c \in \mathbb{R} : P(r_t \leq c | Info_{t-1}) \geq (1 - \alpha)\} \tag{19}$$

where $Info_{t-1}$ represents the past information at time $t - 1$. A good model is expected to produce a high quantity of VaR. Given a set of financial returns, the portfolio return can be defined as:

$$r_{portfolio;t} = \sum_{i=1}^{n} \mu_i r_{i,t} \tag{20}$$

where $r_{i,t}$ is the financial returns at time t for variables $i = 1, \ldots, n$, and μ_i is the weight, where $\sum_{i=1}^{n} \mu_i = 1$.

Suppose the current period is t and we want to calculate the forecasting value of VaR at time $t+1$ using a training data set, the process for computing the value of VaR is given as follows:

(1) Fit the $ARMA(1,1)-GARCH(1,1)$ model and use student t distribution to simulate the residual by using Equation (11). Then the standardized residuals are obtained by:

$$\hat{Z}_{t,j} = \frac{r_{t,j} - \hat{\eta}_j - \hat{\Psi}_j r_{t-1,j} - \hat{\Theta}_j \hat{\sigma}_{t-1,j} \hat{e}_{t-1,j}}{\hat{\sigma}_{t,j}} \quad (21)$$

(2) The ex-ante GARCH variance forecast for $j = 1, \ldots, n$ can be computed by using Equations (12) and (13) as follows:

$$\hat{\sigma}^2_{t+1,j} = \hat{\omega}_j + \hat{\alpha}_j \hat{\varepsilon}^2_{t1,j} + \hat{\beta}_j \hat{\sigma}^2_{t,j} \quad (22)$$

(3) The standardized residuals obtained from $ARMA-GARCH$ are transformed to approximately uniform data $\mathbf{u}_j = u_{1,j}, \ldots, u_{t,j}$ by using the student t cumulative distribution function;

(4) Fit WPVC with approximately uniform data $\mathbf{u}_j$ and estimate copula parameters;

(5) Use the fitted D vine structure with estimated copula parameters to simulate a sample for each financial return variable, i.e.,$v_{t+1,j}$;

(6) Transfer the sample to standard residuals by using the inverse student t cumulative probability distribution functions with parameters obtained in Step (1), and obtain the simulated standardised residuals, i.e., $\hat{Z}_{t+1,j}$;

(7) Calculate the one day ahead forecasting return and variance for each financial variable by using the estimated $ARMA-GARCH$ which is calculated in Step (1), i.e.,

$$\hat{r}_{t+1,j} = \hat{\eta}_j + \hat{\Psi}_j r_{t,j} + \hat{\Theta}_j \hat{\varepsilon}_{t,j} + \hat{\varepsilon}_{t+1,j} \quad (23)$$

VII. Experiments on Multiple Markets

A. Data and Marginal Distribution Specification

To evaluate the performance of this model, we use real-world data, involving 25 financial variables in total. They include (1) eight exchange rates against USD: EUR, GBP, CHF, CAD, AUD, JPY, HKD and SGD ; (2) thirteen major comprehensive indices in the world: ^IXIC, ^GSPC, ^DJI, ^STOXX, ^FTSE, ^GDAXI, ^FCHI, ^AEX, ^BF, ^SSMI, ^N225, ^ST and ^HSI ; (3) three commodity prices: Crude Oil Prices: Brent, Crude Oil Prices: West Texas Intermediate, and the Gold Fixing Price in the London Bullion Market; and (4) one commodity index. These variables are sequentially numbered from $v1$ to $v25$.

The training dataset uses observations from 02/01/1998 to 16/06/2008, with a total of 261 weekly returns (6525 features per time window). These observations from 06/10/2003 to 23/12/2013, in total 470 weekly returns, are used for out-of-sample testing. All the data is downloaded from Yahoo Finance (http://finance.yahoo.com/).

The standardized residuals are transferred to uniform data by using the empirical probability integral transformation. Firstly, we used $ARMA(2,1) - GARCH(1,1)$, $ARMA(1,1) - GARCH(1,1)$ and $AR(1) - GARCH(1,1)$ combined with skewed student t, student t and normal distribution as error type respectively to fit each raw return. Then, we chose the best fitted model by considering the values of Log-likelihood, Akaike information criterion (AIC) and Schwarz criterion(BIC, also named Bayesian information criterion) for each model. Finally, the raw returns of variables $v2, v4, v10, v13, v15, v18$ are fitted with $ARMA(1,1)-GARCH(1,1)$ with skewed student t error distribution, and $v6$s are fitted with $ARMA(1,1)-GARCH(1,1)$ models with student t error distribution. The remain seventeen variables are fitted with $AR(1)-GARCH(1,1)$ with skewed student t error distribution.

TABLE I. Log-likelihood Performance of WPVC with Parameter m_0

m_0	0.1	0.2	0.3	0.4	0.5
LL	2767.226	2767.226	2929.908	2929.908	3031.563
m_0	0.6	0.7	0.8	0.9	1.0
LL	3031.563	3031.563	2929.908	2929.908	2767.226

LL is short for Log-likelihood.

The Ljung-Box (LB) test is used to remove the autocorrelation among these financial returns. In this experiment, the corresponding p values of the LB test are all greater than the significant value 0.05.

B. Weighted Partial Vine Copula Tuning and Truncation Analysis

Typically, the selection of m_0 of the WPVC model is determined by the characteristics of data and domain knowledge. According to the discussion in Section 3.2, m_0 is restricted to interval $[0, 1]$. Table I shows the performance of the Log-likelihood of WPVC with parameter m_0 from 0.1 to 1.0. The high value of the Log-likelihood indicates good performance. According to Table I, the WPVC model with parameter $m_0 = 0.6$ achieves the best performance.

Table II shows the results of the total number of edges against various truncation values (ρ_{trun}) in the training data set. If $\rho_{trun} = 0$, it refers to the non-truncated model, since no partial correlation is less than zero. The training data set is used to examine the performance of non-truncated and truncated models. The results are shown in Fig. 4. These two figures show the estimated time of the truncated model against the non-truncated one, and the corresponding Log-likelihood value. The x axis in Fig. 4 is ρ_{trun} and the dashed line indicates the nontruncated vine as a reference. We can see that the estimated time of WPVC decreases significantly as ρ_{trun} increases. However, the corresponding values of log-likelihood do not show a significant reduction. It can be concluded that the truncated model significantly reduces the estimation time, without losing any important dependencies. Thus, we choose the truncated partial regular vine $\rho_{trun} = 0.05$ in this experiment. $WPVC_{0.05}$ in the following sections indicates a truncated model with truncation value 0.05.

C. In-Sample Performance Analysis

We use the following copula-based models in this experiment to cover a wide spectrum of models when evaluating our new model:

- $WPVC_{0.05}$: the proposed model with truncation value ($\rho_{trun} = 0.05$), built by Algorithm 1;
- $WPVC$: the proposed model without any truncation, built by Algorithm 1;

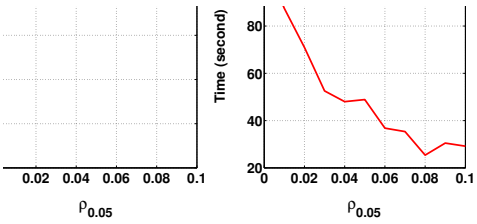


Fig. 4. Performance Analysis of Truncated WPVC

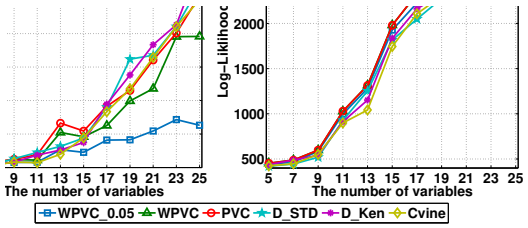


Fig. 5. The Estimate Time and Log-likelihood

TABLE II. THE TOTAL NUMBER OF EDGES AGAINST THE TRUNCATION VALUES

ρ_{trun}	0.01	0.02	0.03	0.04	0.05
Total Number of Edges	269	232	197	180	162
ρ_{trun}	0.06	0.07	0.08	0.09	0.10
Total Number of Edges	145	125	113	101	97

- D_STD: D vine copula-based model, built by [5];
- D_Ken: D vine copula-based model, built by [25];
- $Cvine$: Canonical vine-based model, built by [26];

These bivariate copulas are used as edges, including Gaussian, student t, Clayton, Gumbel and Frank. Table 3 presents the results of the Vuong test with statistical values and the corresponding p-values in the parenthesis. The results are shown without correction and with Akaike and Schwarz corrections respectively. The statistical value of the Vuong test without correction does not indicate which of the $WPVC_{0.05}$ and $WPVC$ models is better. However, the statistical values with Akaike and Schwarz corrections suggest that $WPVC_{0.05}$ is better than $WPVC$, since $WPVC$ is penalized due to its large number of parameters. Thus, this indicates that $WPVC_{0.05}$ is better than $WPVC$, since $WPVC_{0.05}$ uses less parameters to capture the whole dependence structure. Compared to the other three models, it shows that $WPVC_{0.05}$ is the best in all three Vuong tests.

In summary, the in-sample performance tests show that $WPVC_{0.05}$ outperforms the other models.

D. Out-of-Sample Performance Analysis

The out-of-sample performance is evaluated by the Value at Risk (VaR), which a widely used industrial benchmark. The quality of VaR is judged by backtesting. Typically, backtesting methods based on Log-likelihood ratios and a null hypothesis consist of unconditional, independent and conditional coverage tests. VaR can exhibit all the tests at the general significance level (0.05). In this part, we introduce the DCC model [11] as a benchmark, which is a typical model used in the financial industry. In this experiment, we do not use any machine learning models since they do not directly support forecasting VaR.

Table IV presents the statistics of backtesting and the percentage of exceedance failure. A large p-value suggests that the results are accurate and reliable. However, the p-value should at least be greater than 0.05. The percentage of failure for exceedance should be less than or equal to α. The results in Table IV show that $WPVC_{0.05}$ achieves the best performance of all four models. DCC could not pass the backtesting in this experiment, which indicates that this model is not useful for cross-market analysis. The corresponding VaR forecasting results are shown in Fig. 6. The VaR forecasting generated by $WPVC_{0.05}$ accurately predicts the volatility.

TABLE III. VUONG TEST RESULTS

	WPVC	D_STD	D_Ken	Cvine
No Corr	-0.8168	4.8804	3.4409	7.4617
	(-0.4140)	(0.0000)	(-0.0006)	(0.0000)
AIC Corr	0.1925	4.7596	3.3055	7.5042
	(-0.8474)	(0.0000)	(-0.0009)	(0.0000)
BIC Corr	1.9875	4.5447	3.0648	7.5798
	(-0.0469)	(0.0000)	(-0.0022)	(0.0000)

No Corr, AIC Corr or BIC Corr are short for Vuont test without correction and with Akaike and Schwarz corrections respectively. The table shows results of Vuong tests at the 5% level, testing the R1 model ($WPVC_{0.05}$) against the models R2 (including $WPVC$, D_{STD}, D_{Ken} and $Cvine$ models). If the statistical value is greater than 1.96, we favor the model R1. If it is less than -1.96, the model R2 is chosen. If between -1.96 and 1.96, no conclusion is made.

VIII. CONCLUSIONS AND FUTURE WORK

Modeling the dependence across financial markets such as between stock markets and exchange rate markets is very challenging but necessary for typical business including cross-market trading and risk management. While copula-based models have been shown effective in capturing multivariate correlations, existing models have trouble in effectively addressing the complex and asymmetrical dependence and high dimensional issues as well. This work has proposed a weighted partial vine copula model to handle these issues. The model has been demonstrated more workable than typical baselines through an analysis of the complicated structures of portfolios in trading sixteen years of thirteen comprehensive indices, eight currency rates and three commodity price indices from statistical and risk evaluation perspectives. We are further testing the performance of the weighted partial D vine with other kinds of high-dimensional financial time series data.

TABLE IV. BACKTESTING RESULTS OF VALUE AT RISK FOR CURRENCIES

	$1-\alpha$	POF	LR_{UC}	LR_{IC}	LR_{CC}
$WPVC_{0.05}$	99%	5 1.08%	0.032 (0.857)	2.315 (0.128)	2.347 (0.309)
	95%	26 5.64%	0.382 (0.536)	0.188 (0.665)	0.570 (0.752)
	90%	52 11.50%	1.100 (0.294)	1.582 (0.208)	2.683 (0.261)
D_STD	99%	9 1.95%	2.186 (0.139)	2.221 (0.136)	4.408 (0.110)
	95%	27 5.86%	0.451 (0.502)	0.133 (0.715)	0.584 (0.747)
	90%	57 12.36%	1.363 (0.245)	3.533 (0.060)	4.896 (0.086)
D_Ken	99%	10 2.17%	3.376 (0.066)	1.843 (0.175)	5.218 (0.074)
	95%	27 5.86%	0.451 (0.502)	0.133 (0.715)	0.584 (0.747)
	90%	57 12.36%	1.363 (0.243)	3.533 (0.060)	4.896 (0.086)
$Cvine$	99%	11 2.39%	4.770 (0.029)	1.439 (0.230)	6.209 (0.045)
	95%	29 6.29%	0.662 (0.416)	0.042 (0.838)	0.704 (0.703)
	90%	59 12.80%	1.782 (0.182)	4.469 (0.035)	6.251 (0.044)
DCC	99%	103 22.34%	466.082 (0.000)	5.449 (0.021)	471.533 (0.000)
	95%	133 28.85%	276.570 (0.000)	15.257 (0.000)	291.827 (0.000)
	90%	59 32.97%	180.570 (0.000)	15.333 (0.000)	195.903 (0.000)

Here PoF is the percentage of exceedance failure. The first row shows the exceedance number, and the second row gives the corresponding percentage;
LR_{UC}, LR_{IC} and LR_{CC} are short for the likelihood ratio of unconditional, independent and conditional coverage respectively. The first row in each cell shows the statistics value, while the corresponding p-value is given in the corresponding parenthesis. The critical value of LR_{UC} or LR_{IC} is 3.841, while the critical value of LR_{CC} is 5.991.

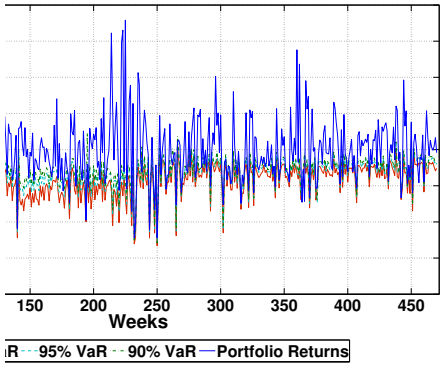


Fig. 6. The VaR Forecast of Portfolio Returns by Using $WPVC_{0.05}$

REFERENCES

[1] L. Cao, "Non-iidness learning in behavioral and social data," *The Computer Journal*, p. bxt084, 2013.

[2] J. Białkowski, *Modelling Returns on Stock Indices For Western and Central European Stock Exchanges: A Markov Switching Approach*. Univ., Department of Economics, 2003.

[3] J. Neville and D. Jensen, "Relational dependency networks," *Journal of Machine Learning Research*, vol. 8, pp. 653–692, 2007.

[4] G. Elidan, "Copulas in machine learning," *Copulae in Mathematical and Quantitative Finance*, p. 39, 2013.

[5] K. Aas, C. Czado, A. Frigessi, and H. Bakken, "Pair-copula constructions of multiple dependence," *Insurance: Mathematics and economics*, vol. 44, no. 2, pp. 182–198, 2009.

[6] B. Letham, W. Sun, and A. Sheopuri, "Latent variable copula inference for bundle pricing from retail transaction data," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 217–225.

[7] A. K. Nikoloulopoulos, H. Joe, and H. Li, "Extreme value properties of multivariate t copulas," *Extremes*, vol. 12, no. 2, pp. 129–148, 2009.

[8] B. Póczos, Z. Ghahramani, and J. Schneider, "Copula-based kernel dependency measures," in *Proceedings of the 29st International Conference on Machine Learning (ICML-12)*, 2012.

[9] W. Wei, X. Fan, J. Li, and L. Cao, "Model the complex dependence structures of financial variables by using canonical vine," in *Proceedings of the 21st ACM international CIKM*. ACM, 2012, pp. 1382–1391.

[10] F. Longin and B. Solnik, "Extreme correlation of international equity markets," *The journal of finance*, vol. 56, no. 2, pp. 649–676, 2001.

[11] R. Engle, "Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models," *Journal of Business & Economic Statistics*, vol. 20, no. 3, pp. 339–350, 2002.

[12] L. De Raedt and K. Kersting, *Probabilistic inductive logic programming*. Springer, 2008.

[13] Y. He, Y. Qi, K. Kavukcuoglu, and H. Park, "Learning the dependency structure of latent factors," in *Advances in Neural Information Processing Systems*, 2012, pp. 2375–2383.

[14] J. Gao and H. Suzuki, "Unsupervised learning of dependency structure for language modeling," in *Proceedings of ACL*, 2003, pp. 521–528.

[15] Z. Ghahramani, "An introduction to hidden markov models and bayesian networks," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 15, no. 01, pp. 9–42, 2001.

[16] A. J. Patton, "Modelling asymmetric exchange rate dependence," *International economic review*, vol. 47, no. 2, pp. 527–556, 2006a.

[17] H. Manner and J. Segers, "Tails of correlation mixtures of elliptical copulas," *Insurance: Mathematics and Economics*, vol. 48, no. 1, pp. 153–160, 2011.

[18] A. Min and C. Czado, "Bayesian inference for multivariate copulas using pair-copula constructions," *Journal of Financial Econometrics*, vol. 8, no. 4, pp. 511–546, 2010.

[19] G. Elidan, "Copula bayesian networks," in *Advances in neural information processing systems*, 2010, pp. 559–567.

[20] M. Sklar, *Fonctions de répartition à n dimensions et leurs marges*. Université Paris 8, 1959.

[21] H. Joe, *Multivariate models and multivariate dependence concepts*. CRC Press, 1997, vol. 73.

[22] D. Kurowicka and R. M. Cooke, *Uncertainty analysis with high dimensional dependence modelling*. John Wiley & Sons, 2006.

[23] T. Bedford and R. M. Cooke, "Vines: A new graphical model for dependent random variables," *Annals of Statistics*, pp. 1031–1068, 2002.

[24] D. Kurowicka and H. Joe, *Dependence Modeling: Vine Copula Handbook*. World Scientific, 2011.

[25] C. Almeida, C. Czado, and H. Manner, "Modeling high dimensional time-varying dependence using d-vine scar models," *arXiv preprint arXiv:1202.2008*, 2012.

[26] J. Dissmann, E. C. Brechmann, C. Czado, and D. Kurowicka, "Selecting and estimating regular vine copulae and application to financial returns," *Computational Statistics & Data Analysis*, vol. 59, pp. 52–69, 2013.