Value at Risk calculation using sparse vine copula models

Time series seminar

2021711914 김동영

2021.11.15

Contents

Introduction

- Recently, asset price change rapidly in the global market

  ⇒ The importance of how to manage market risk

- VaR (Value at Risk) is one of the extreme value theory (EVT)

  ⇒ statistical measure of the portfolios of assets or financial entities

- VaR is a method for estimating the **maximum loss** of asset price

What is the VaR? (not VAR)

Value at Risk (VaR)

- $r_t = \log(P_t) - \log(P_{t-1})$
- $R_t := \sum_{i=1}^{d} w_i r_{t,i}$
- Focus and interest on *Loss* rather than profit
- VaR is a simple quantile calculation with distribution assumption

$$P(R_t \leq VaR) = 1 - \alpha$$

- **However, a problem is to estimate the multivariate distribution**

Copula function (Skalr's theorem)

- Every multivariate cumulative joint distribution function $H(\cdot)$

$$
\begin{aligned}
& H(x_1, \ldots, x_d) = Pr(X_1 \leq x_1, \ldots X_d \leq x_d) \\
\iff \ & H(x_1, \ldots, x_d) = C(F_1(x_1), \ldots, F_d(x_d)) \\
\implies \ & h(x_1, \ldots, x_d) = c(F_1(x_1), \ldots, F_d(x_d)) \cdot f_1(x_1) \cdots f_d(x_d)
\end{aligned}
$$

- Separately specifying the marginal distributions and the copula
- It can be used to **generate random variables** considering **dependency**

copula simulation

- z = MASS::mvrnorm(2000, rep(0,3), Sigma=sigma)

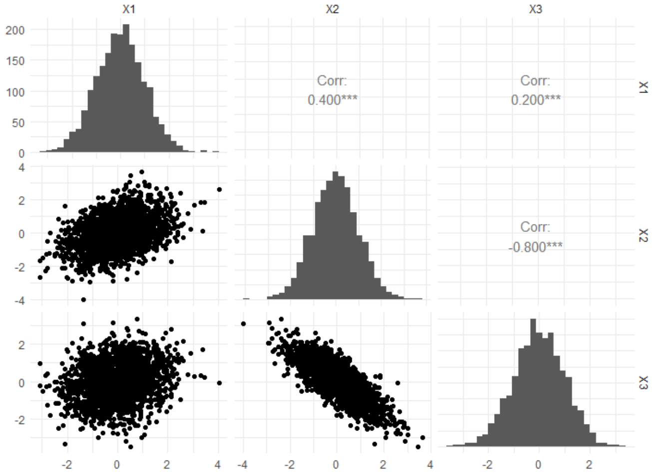


Figure 1: Multivariate normal distribution pairs plot

copula simulation

- u = apply(z, c(1,2), pnorm)

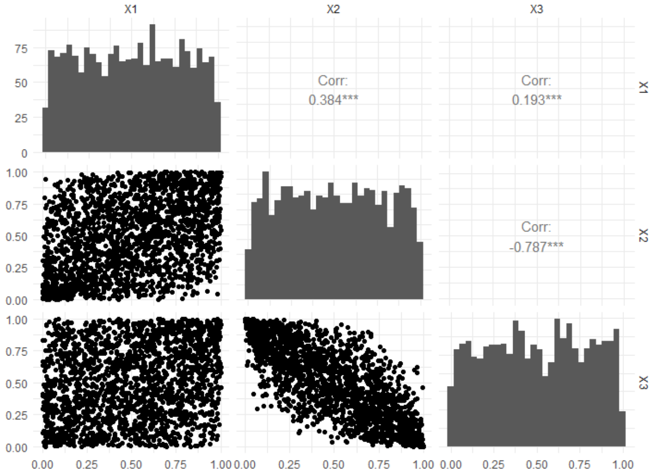


Figure 2: Normal cdf transformation for MVN data pairs plot

copula simulation

- qgamma(X1, shape=2, scale=1), qbeta(X2, 0.5,0.5), qt(X3, df=5)

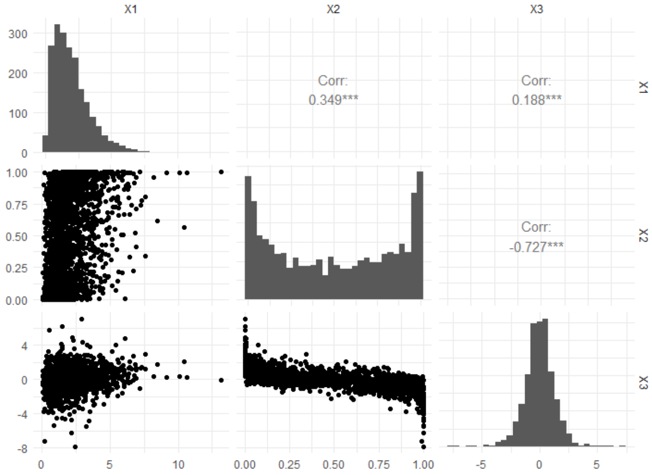


Figure 3: Inverse marginal distribution transformation pairs plot

Pair Copula Constructions (PCC)

- High dimensions of three or more can be decomposed into bivariate copula and marginal density

- *When* $d = 3$

$$
\begin{aligned}
h(x_1, x_2, x_3) &= f_{3|1,2}(x_3|x_1, x_2) \cdot f_{2|1}(x_2|x_1) \cdot f_1(x_1) \\
&= f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \\
&\quad \times c_{12}(F_1(x_1), F_2(x_2)) \cdot c_{23}(F_2(x_2), F_3(x_3)) \\
&\quad \times c_{13|2}(F_{1|2}(x_1|x_2), F_{3|2}(x_3|x_2))
\end{aligned}
$$

- 3-dim $\rightarrow$ bi-copula & marginal density

Vine Copula

- Financial data with heavy tail and skewness (stylized facts)

  $\Rightarrow$ Problem with multivariate copula ex. MVN, MVt

- These Distributions do not explain this property (underfit)

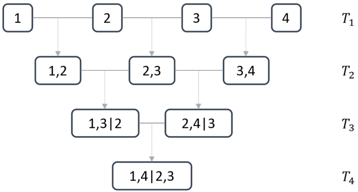
  $\Rightarrow$ Connecting bi-copula to form a more flexible function

- Vine copula determines the **unique solution** of PCC considering **dependency**

  $\Rightarrow$ **How to represent a multivariate density as Vine Copula**

Vine Copula

- Tree graph can be represented by this formula and matrix



$$
\begin{aligned}
C(u_1, u_2, u_3, u_4) = \; & C_{12}(u_1, u_2) C_{23}(u_2, u_3) C_{34}(u_3 u_4) \\
& \times C_{13|2}\big(u_{1|2}, u_{3|2}\big) C_{24|3}\big(u_{2|3}, u_{4|3}\big) \\
& \times C_{14|23}(u_{1|23}, u_{4|23})
\end{aligned}
$$

$$
R = \begin{pmatrix} 4 & & & \\ 1 & 3 & & \\ 2 & 1 & 2 & \\ 3 & 2 & 1 & 1 \end{pmatrix} \; or \; \begin{pmatrix} 1 & & & \\ 4 & 2 & & \\ 3 & 4 & 3 & \\ 2 & 3 & 4 & 4 \end{pmatrix}
$$

Figure 4: Three way of representation of vine copula

Vine Copula

- Vine copula can apply a different copula to each bivariate distribution
- Which copula is desirable?
  - close to the multivariate joint distribution
  - Order of sequence and each bi-copula to describe dependency best

However,

- The number of possible combinations : $\frac{d(d-1)}{2}$
- *As d increase*, it is hard to find an optimal solution

$\Rightarrow$ Using Lasso penalty

Sparse Vine Copula

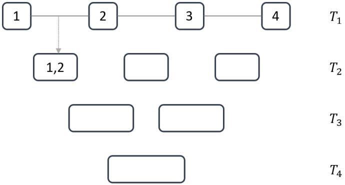
- Difficult to find a solution if $d \geq 20$
- Vine Copula with Lasso penalty (Muller and Czado(2019))

  $\Rightarrow$ simplifying calculations in high dimension situation
- For applying lasso for vine copula, we formulate structure equation model

$$
\begin{aligned}
X_1 &= \psi_1 \epsilon_1 \\
X_2 &= \beta_{2,1} X_1 + \psi_2 \epsilon_2 \\
X_3 &= \beta_{3,1} X_1 + \beta 3, 2 X_2 +_3 \epsilon_3 \\
X_4 &= \beta_{4,1} X_1 + \beta_{4,2} X_2 + \beta 4, 3 X_3 +_4 \epsilon_4
\end{aligned}
$$

Sparse Vine Copula

- Represent vine copula as tree and matrix



$$\begin{pmatrix} 4 & & & \\ ? & 3 & & \\ ? & ? & 2 & \\ ? & ? & 1 & 1 \end{pmatrix}$$

- Regularization path

$$argmin_{\varphi}\left(\frac{1}{2n}\sum_{i=1}^{n}(x_{i,j} - \sum_{l\in\mathcal{H}\setminus\mathcal{B}} \varphi_{j,l}x_{i,l})^2 + \sum_{l\in\mathcal{H}\setminus\mathcal{B}\setminus\mathcal{W}} \lambda_{j,l}|\varphi_{j,l}|\right)$$

- Notation
  - $\mathcal{H}$: set of potential regressors
  - $\mathcal{B}$: set of blacklist regressors
  - $\mathcal{W}$: set of whitelist regressors

Simulation

- Comparison of VaR calculated by regular and sparse vine copula
- Use out of sample forecasting with MSPE

$$MSPE = E[(g(x_i) - \hat{g}(x_i))^2]$$

- Point of simulation
  1. Generate random number from MVt to get return data
  2. Set the covariance matrix of MVt to correlation of the real stock price
  3. Two simulation setting
     - Original data set (non-sparse)
     - Data with a correlation of less than 0.6 set to 0 (sparse)

Simulation

- Comparison of VaR calculated by regular and sparse vine copula (1)

Table 1: $n = 1000$

	$d = 10$	$d = 30$	$d = 50$
$df = 5$			
R-vine	114.01	1365.09	2562.34
L-vine	114.79	445.38	1528.99
$df = 15$			
R-vine	31.96	98.16	602.89
L-vine	32.41	86.39	623.76

Table 2: $n = 500$

	$d = 10$	$d = 30$	$d = 50$
$df = 5$			
R-vine	179.76	947.50	2983.20
L-vine	178.51	490.48	1919.12
$df = 15$			
R-vine	38.83	231.44	781.66
L-vine	38.25	222.79	673.58

Table 3: $n = 300$

	$d = 10$	$d = 30$	$d = 50$
$df = 5$			
R-vine	177.97	1538.32	3057.82
L-vine	183.07	865.78	2195.18
$df = 15$			
R-vine	51.18	312.58	786.81
L-vine	55.10	307.85	754.01

Figure 5: Simulation setting 1 (original)

Simulation

- Comparison of VaR calculated by regular and sparse vine copula (2)

Table 4: $n = 1000$

	$d = 10$	$d = 30$	$d = 50$
$df = 5$			
R-vine	21.81	85.94	363.11
L-vine	21.22	67.32	211.29
$df = 15$			
R-vine	8.64	26.83	75.09
L-vine	9.45	20.25	83.27

Table 5: $n = 500$

	$d = 10$	$d = 30$	$d = 50$
$df = 5$			
R-vine	29.46	73.06	453.45
L-vine	28.71	61.45	297.17
$df = 15$			
R-vine	9.1	24.26	85.42
L-vine	9.38	23.46	92.88

Table 6: $n = 300$

	$d = 10$	$d = 30$	$d = 50$
$df = 5$			
R-vine	33.64	116.67	456.55
L-vine	33.85	99.59	260.47
$df = 15$			
R-vine	11.96	30.45	103.76
L-vine	11.73	27.09	92.07

Figure 6: Simulation setting 2 (sparse)

Real data study

- Comparison through real stock data
- 25 KOSPI stocks ex) 삼성전자, 네이버, SK하이닉스 etc.
- data period
  - train : 2016/01/01 $\sim$ 2020/02/20
  - test : 2020/02/21 $\sim$ 2020/12/30
- Log return : $r_t = log(P_t) - log(P_{t-1})$

  $\Rightarrow$ Differencing and variance stabilizing can make data stationary.

Real data study

- model : GARCH(1,1)

$$\sigma_t^2 = \alpha_0 + \alpha_1 r_{t-1}^2 + \beta_1 \sigma_{t-1}^2$$

1. using residual for data

$$\hat{\epsilon}_t = \frac{r_t}{\hat{\sigma}_t}$$

2. Copula data between 0 and 1 are converted by t-dist cdf and residual

3. Fitting (sparse) vine copula model using copula data

4. VaR calculation using the inverse of t-distribution

   by generating 10,000 samples from fitted copula model

Real data study

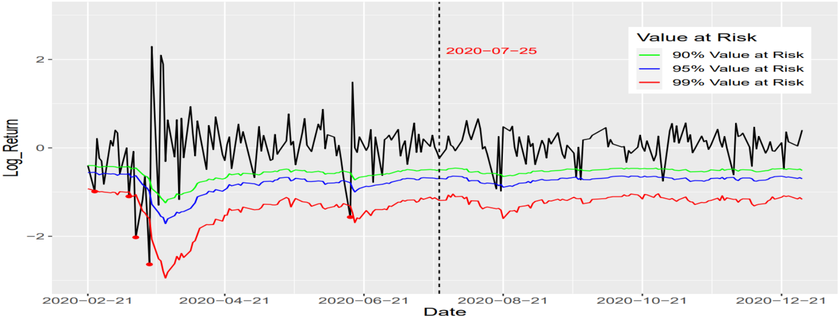


Figure 7: (non sparse) vine copula Value at risk plot

Real data study

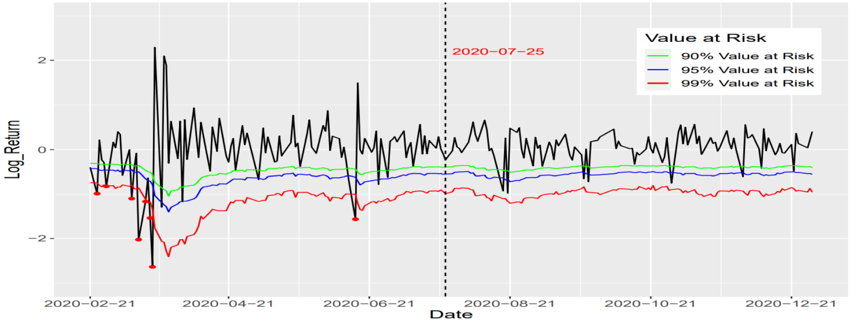


Figure 8: sparse vine copula Value at risk plot

Real data study

- Kupiec test (based on likelihood ratio test)
- H_0 : the observed violation rate = expected violation rate $(1-\alpha)$
- Under H_0, $L \sim \chi^2(1)$

	VaR_α	Expected exceedances	Actual exceedances	H0 "Correct Exceedances"
	From 2020-02-21 to 2020-07-24 (107 days)			
R-vine	$VaR_{0.1}$	10	16	p=0.1
	$VaR_{0.05}$	5	8	p=0.27
	$VaR_{0.01}$	1	5	p=0.0055
	From 2020-07-25 to 2020-12-30 (107 days)			
	$VaR_{0.1}$	10	7	p=0.2
	$VaR_{0.05}$	5	4	p=0.53
	$VaR_{0.01}$	1	4	p=0.028

	VaR_α	Expected exceedances	Actual exceedances	H0 "Correct Exceedances"
	From 2020-02-21 to 2020-07-24 (107 days)			
L-vine	$VaR_{0.1}$	10	18	p=0.03
	$VaR_{0.05}$	5	13	p=0.003
	$VaR_{0.01}$	1	8	p=1.4*10-5
	From 2020-07-25 to 2020-12-30 (107 days)			
	$VaR_{0.1}$	10	8	p=0.36
	$VaR_{0.05}$	5	6	p=0.77
	$VaR_{0.01}$	1	6	p=8*10-4

Figure 9: Kupiec test result

Summary

- Pros
  - Penalty term enables VaR calculation in situation where the number of assets is large ($p \approx 100, 1000$)
  - Sparse vine copula can be used not only in Value at Risk different situations requiring multivariate distribution estimation
- Cons
  - In real data study, there is no point that Sparse vine copula is better
  - No explanation for the actual correlation of stock price in simulation
  - Sparse vine copula is said to be faster empirically, but there are no figures like elapsed time or ratio

Thank you