Title of your report

Dependence Modelling in Ultra High Dimensions with Vine Copulas

Research institution

Department of Mathematics

Principal Investigator

Claudia Czado

Researchers

Dominik Müller, Thomas Nagler

Project partners

-

LRZ project ID of the projects you report in this article

pr23fo

Introduction

Vine copulas [1] are highly versatile statistical models for the dependence between several quantities. They express how multiple variables are related to each other, for example: stocks in a stock market, weather stations covering a country, or gene expressions in a genetic dataset.

In general, describing the joint behavior of variables in arbitrary dimensions is a highly complicated task. Vine copula models tackle this problem by decomposing the dependence structure into several two-dimensional building blocks. This leads to a graphical model consisting of several nested trees, where the first tree encodes unconditional pairwise dependencies, and subsequent trees encode conditional pairwise dependencies. An example on four variables is shown in Figure 1. Yet, this reformulation of the problem comes with a price tag: the number of (conditional) pairs in the model grows quadratically in the number of variables.

In high dimensional applications (e.g., 1,000 variables), the model needs to be simplified to keep it computationally tractable. In the end, only the important dependences should be included to obtain a sparse model. The main difficulty is to find the right tradeoff between model fit and complexity. Additionally, an efficient implementation of the model is essential.

Results and Methods

Advances in statistical methodology

The established standard for fitting vine copula models is the algorithm of Dißmann et al. [2], a greedy algorithm that aims to capture most of the dependence in the first couple of trees. In high dimensions, the resulting models become overly complex and computationally infeasible. A popular trick to obtain sparser model is *truncation*, i.e., only modeling dependencies in the first few trees, while assuming conditional independence for the remainder. But even truncated models can be unnecessarily complex when there are many independent pairs in the data.

Our research group developed several techniques to exploit sparsity patterns in the data. Müller [3] builds on techniques from sparse graphical models to first obtain a simple proxy model, which is then translated into a vine copula. A more aggressive approach proposed by Müller [3] allows to cluster the variables first and then estimate isolated models on each cluster, which heavily reduces the computational burden. An alternative route was taken by Nagler et al. [4], who developed a method that only includes dependencies whose strength passes a certain threshold. The optimal threshold can be selected automatically with almost no computational overhead.

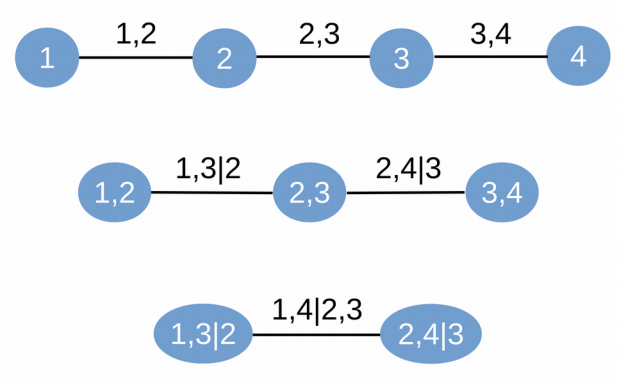

Figure 1: Illustration of a vine on four variables.

Advances in computation

The standard implementation of vine copula models, the R package *VineCopula*, has been actively developed and maintained from our group. However, it was initially designed for problems of small or moderate dimensions and has severe limitations in high dimensional applications. One issue is that the R interpreter is single-threaded by design and multiple instances have to be run for concurrent computations. Hence, the R implementation cannot fully exploit the fact that fitting a vine copula is a sequence of embarrassingly parallel tasks. Another issue is that the standard algorithms for vine copula models are of quadratic complexity in both memory and time.

To empower applications in high dimensions, we developed a pure C++ implementation, *vinecopulib* [5], that has interfaces to both R and Python. It implements

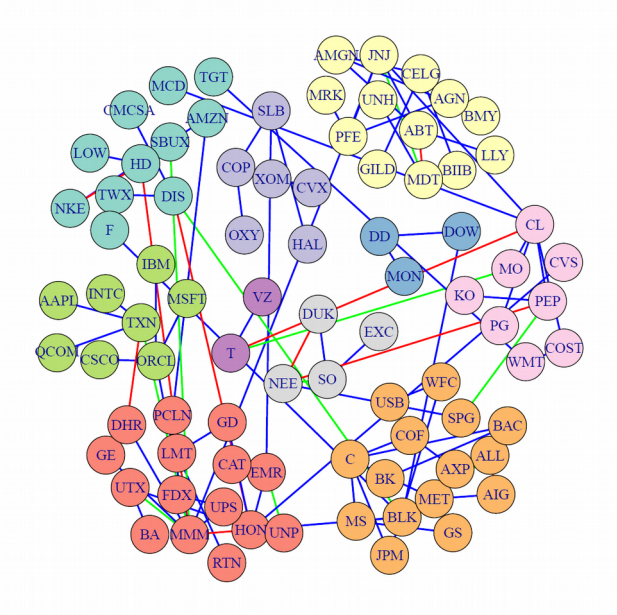

Figure 2: A high dimensional network of the stocks which are comprised in the S&P100 index. The picture shows how the algorithms of [2] detect connections between stocks of similar industry sectors, which are encoded by the colors. Picture taken from [2].

optimized data structures and algorithms that remove many redundancies and scale linearly with the dimension if the model is truncated.

Numerical evaluation on simulated data

All newly developed methods were benchmarked extensively against the standard method of Dißmann et al. [2] via simulations. To assess their performance, we fitted baseline models on several data sets from various disciplines and used it as ground truth. One example we used are 85 stocks of the S&P100 stocks index, see Figure 2. We then simulated from the baseline model and fitted all competitor models to the synthetic data. The new methods were found to fit almost equally well to the data while at the same time being of much lower complexity.

All this, however, requires extensible computation time. For all implementations, the statistical software R and several individually developed libraries were used. Most of the algorithms and simulation run in parallel, typically using 28 – 32 cores per job. For the simulations we carried out in 85 dimensions, the standard fitting routine took about 5 hours. With the newly developed methods, the fitting time was reduced to approximately 30 minutes. Other benchmarks went over 2,000 dimensions and took almost two days for a single fit of a sparse model. Fitting the standard method was only feasible until around 400 dimensions, i.e., 400 stocks or similar.

Application to financial risk management

Several of the new methods were successfully applied to risk management problems in financial markets. We fitted models to portfolios of hundreds of stocks and evaluated how well the one-day ahead risk is predicted. This procedure is repeated for every trading day over several years, thus requiring many computing hours. Figure 3

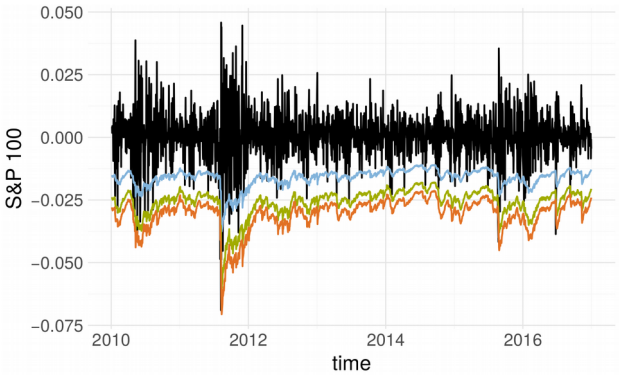

Figure 3: Time series of returns of a portfolio of stocks from the S&P 100 index (black) along with one-day-ahead predictions of the Value-at-Risk for different risk levels (colors).

shows the realized returns of a portfolio of S&P 100 stocks along with predicted levels of the risk.

Summary

This research project developed, evaluated, and applied statistical dependence models for hundreds or thousands of variables. It used approximately 340,000 CPU hours and generated thousands of files totaling almost 500 GB of data. Without the CoolMUC2, exploration, development and validation of the new methods would not have been possible.

On-going Research / Outlook

The project required extensive computational capabilities for running simulations, fitting the models, as well as storing the models and data sets. The algorithms we use run many tasks in parallel so that computation time is inverse proportional to the number of threads. One node of the CoolMUC2 allows for only 28 threads. Hence, an implementation that runs on multiple nodes will be necessary to scale the models up to problems of higher magnitudes.

References and Links

[1] http://www.vine-copula.org/

[2] Jeffrey Dißmann, Eike C. Brechmann, Claudia Czado, Dorota Kurowicka. 2013. Selecting and estimating regular vine copulae and application to financial returns. COMPUT STAT DATA AN 59 (March 2013), 52-69. DOI: https://doi.org/10.1016/j.csda.2012.08.010

[3] Dominik T. Müller. 2017. Selection of Sparse Vine Copulas in Ultra High Dimensions. PhD Dissertation. Technical University of Munich. http://mediatum.ub.tum.de/node?id=1382835

[4] Thomas Nagler, Christian Bumann, and Claudia Czado. 2018. Model selection in sparse high-dimensional vine copula models with application to portfolio risk. arXiv:1801.09739. Retrieved from https://arxiv.org/abs/1801.09739

[5] http://www.vinecopulib.org/