

# **Model selection for copulas and model pruning for neural networks**

**Vinnie Ko**

Dissertation presented for the degree of  
Philosophiae Doctor (PhD)



Department of Mathematics  
University of Oslo  
May 2019



# Forord

Et eventyr, det er ordet som passer best for å beskrive de tre årene jeg har tilbragt som doktorgradsstipendiat i statistikk på Matematisk institutt ved Universitetet i Oslo.

“Hvem vil du ha som veileder?”

Det var noe av det første jeg ble spurt, da jeg signerte kontrakten. Det viste seg at jeg hadde en ekstrem luksusposisjon ettersom jeg fikk en PhD stilling uten fast emne og veileder.

Etter samtaler med professorene i seksjonen, valgte jeg Nils Lid Hjort som min veileder, ikke bare fordi han er svært dyktig i sitt fag, men også fordi han har forståelse for aktivitetene jeg holdt på med utenfor statistikk. Noen av disse aktivitetene kunne virke poengløse ved første øyekast og blir begripelig bare når de ses bakover i tid. Takk Nils for tilliten du alltid hadde til meg.

Nils tok meg også imot med åpne armer i sin forskningsgruppe FocuStat. Jeg vil takke alle deltakerne av FocuStat for alle våre aktiviteter blant annet lange møter, middager, konferanser, basketballkamp og litterær kveld. Dere gjorde tiden min i Oslo til den mest norske opplevelsen.

Jeg var heldig nok å ha Ingrid Hobæk Haff som biveileder. Jeg har opplevd samarbeidet vårt både behagelig og effektivt, men det jeg vil takke deg aller mest for er at du var tålmodig med mitt temperament og håndterte det på en ærlig og direkte måte. Jeg var virkelig glad, da du kom innom kontoret mitt for å snakke om det i begynnelsen. (‘Konfliktsky’ var et av ordene som dukket opp ofte i hodet mitt, da jeg var i Norge.)

Undervisning var uten tvil en av yndlingsdelene mine i mitt PhD liv. Da jeg var gruppelærer for GLM-kurset, var Ørnulf Borgan snill og hjelpsom nok å lese mitt løsningsforslag hver uke og gi omfattende tilbakemelding, selv når han var i utlandet.

Jeg vil også takke alle kollegaer i statistikk seksjonen for et trygt og hyggelig miljø.

I det siste året av min PhD var jeg på forskningsopphold på Datalogisk institut ved Københavns Universitet. Jeg vil takke Fabian Gieseke for invitasjonen og for hans forståelse da jeg var opptatt med revisjon av mine papers fra Oslo og skriving av avhandling.

Sist, men absolutt ikke minst, vil jeg takke Jolien for at hun forstod mitt begjær etter eventyr og utstod to års langdistanseforhold, og at hun til slutt flyttet til et nytt land etter meg.

Vinnie Ko, Oslo, mai 2019

# List of papers and manuscripts

## Paper I

Ko, V. & Hjort, N. L. (2019). Model robust inference with two-stage maximum likelihood estimation for copulas. *Journal of Multivariate Analysis*, 171: 362–381.

## Paper II

Ko, V. & Hjort, N. L. (2019). Copula information criterion for model selection with two-stage maximum likelihood estimation. *Econometrics and Statistics*, in press.

## Paper III

Ko, V., Hjort, N. L. & Hobæk Haff, I. (2019). Focused information criteria for copulas. *Scandinavian Journal of Statistics*, in press.

## Paper IV

Ko, V., Oehmcke, S. & Gieseke, F. (2019). Magnitude and Uncertainty Pruning Criterion for Neural Networks. *Submitted for publication*.

# Contents

|                                                                                |           |
|--------------------------------------------------------------------------------|-----------|
| <b>Forord</b>                                                                  | <b>i</b>  |
| <b>List of papers</b>                                                          | <b>ii</b> |
| <b>1 Introduction</b>                                                          | <b>1</b>  |
| <b>2 Copulas</b>                                                               | <b>3</b>  |
| 2.1 Estimation . . . . .                                                       | 7         |
| 2.1.1 Maximum likelihood . . . . .                                             | 8         |
| 2.1.2 Two-stage maximum likelihood . . . . .                                   | 8         |
| 2.1.3 Pseudo maximum likelihood . . . . .                                      | 9         |
| 2.1.4 Example . . . . .                                                        | 10        |
| <b>3 Model selection</b>                                                       | <b>12</b> |
| 3.1 KL divergence based model selection criteria . . . . .                     | 12        |
| 3.1.1 True model assumption . . . . .                                          | 13        |
| 3.2 Focused information criterion (FIC) . . . . .                              | 13        |
| <b>4 Neural networks</b>                                                       | <b>16</b> |
| 4.1 Convolution layer . . . . .                                                | 16        |
| 4.2 Pooling . . . . .                                                          | 17        |
| 4.3 Fully connected layer . . . . .                                            | 17        |
| 4.4 Activation function . . . . .                                              | 18        |
| 4.5 Universal approximation theorem . . . . .                                  | 18        |
| 4.6 Stochastic gradient descent . . . . .                                      | 19        |
| 4.7 Pruning . . . . .                                                          | 20        |
| <b>5 Summary of papers</b>                                                     | <b>23</b> |
| 5.1 Paper I . . . . .                                                          | 23        |
| 5.2 Paper II . . . . .                                                         | 24        |
| 5.3 Paper III . . . . .                                                        | 25        |
| 5.4 Paper IV . . . . .                                                         | 26        |
| <b>6 Discussion</b>                                                            | <b>28</b> |
| 6.1 Analytical tool for analyzing the impact of true mode assumption . . . . . | 28        |
| 6.2 CIC or $AIC_{2ML}$ ? . . . .                                               | 29        |
| 6.3 Two-stage density estimators via a parametric start . . . . .              | 30        |
| 6.4 Utilizing uncertainty information from training . . . . .                  | 31        |
| <b>References</b>                                                              | <b>32</b> |
| <b>Papers I-IV with supplementary material</b>                                 | <b>39</b> |



# 1 Introduction

Thanks to the technological developments in the last decades, there have been huge amount of data available across many different fields and industries. This led to increase in demand for statistical and machine learning models and the number of such models is higher than ever at the moment. In the wild jungle of models, it is a challenging task to evaluate each candidate model properly and find the most suitable one for the given data and purpose. This thesis explores two topics related to choosing the most proper model.

The first main topic is *model selection*. Model selection is at the core of statistics, since there are almost always multiple candidate models that can be used for the given data. In general sense, model selection also involves all the modifications that can be made to a model. e.g. Do we need all the variables? Is it better to increase or decrease the number of parameters through reparameterizing?

In Paper II and III, model selection criteria for copula models are developed. Copula is a multivariate probability distribution that is often used for modeling the joint behavior of variables and it is regularly used in finance and actuarial science (Embrechts, 2009). To use copula in practice, one has to choose a univariate distribution for each variable (or use a nonparametric equivalent) and a copula that binds them together. This means that the number of candidate models grows very quickly. Imagine that we want to model 4-dimensional data with a copula and that we have 4 candidate copulas and 3 candidate univariate distributions for each variable. Then, there are  $324 (= 4 \times 3^4)$  possible models. It is thus crucial to have a tool for systemically evaluating and selecting the most appropriate model.

Specifically, we look into the case when the parameters of the copula model are estimated in a step-wise fashion (i.e. First, estimate the parameters of each univariate distribution and then estimate the copula parameters). What is often done in practice is that one performs piece-wise model comparison for each part separately. This approach, obviously, cannot compare the models as a whole. In Paper II and III, we developed model selection criteria that allow more systemic comparison of copula models that are estimated in a step-wise fashion.

A general theme that is present in this thesis is *model robust*, in the sense that we do not make the assumption that the candidate parametric model is correctly specified and captures the true model that generated data. This is often called as ‘true model assumption’ and is one of the key assumptions behind many classical results in statistics. The model selection criteria developed in Paper II and III are model robust since they do not make this true model assumption. In Paper I, which is an important stepping stone to Paper II and III, this true model assumption is studied.

When there are a large number of candidate models, it often gives better predictive performance if the outputs from multiple models are combined into one, instead of using the output from a single model. This is called model averaging (Claeskens & Hjort, 2008, Chapter 7). The model selection criteria developed in this thesis can be used for identifying the best performing models

and combining them (e.g. through a weighted sum, where the weights are functions of model selection criteria).

The second main topic of this thesis is *model pruning* for neural networks. Neural network is a class of model in machine learning that achieved state-of-the-art performance in wide range of tasks like computer vision and has gained big popularity in recent years. One of the issues of modern neural networks is that they contain easily millions and sometimes billions of parameters. Denil et al. (2013) showed that there is significant redundancy in the parameterization of many deep neural network models. This overparameterization not only requires unnecessary computational resources, but can also lead to overfitting of the models. In Paper 4, inspired by the Wald test statistic, we develop a novel pruning criterion that can evaluate the parameters of the network for pruning.

The further structure of this thesis is as follows. Sections 2–4 give a gentle introduction to the key concepts that play an important role in this thesis. Section 5 contains summaries of four papers constituting this thesis. Finally, Section 6 offers possible improvements and future works.

## 2 Copulas

When one has univariate data and looks for distributions that can fit into the data, there is a long list of distributions where one can pick from. However, when it comes to multivariate data where each variable has different marginal distributions, there are not many choices. For example, if one has 3 dimensional data where variable 1 follows gamma distribution, variable 2 follows Weibull distribution and variable 3 follows Pareto distribution, it is not clear how their joint distribution can be modeled. With copula one can pick univariate distributions that fits each marginal distribution best and easily ‘bind them together’. This allows very flexible modeling of multivariate distribution. Thanks to this advantage, copula got its popularity in the early 2000s and is now often used in many high-dimensional statistical applications.

So, what is copula exactly? Let  $(Y_1, \dots, Y_d)$  be a  $d$ -variate random variable vector originating from a joint density  $g(y_1, \dots, y_d)$ .  $G(y_1, \dots, y_d)$  indicate the cumulative distribution function corresponding to  $g$ .

Then, we can perform transformation of variables on each univariate variable with their own marginal cdfs  $U_j = G_j(y_j)$ . This is called probability integral transform (PIT) and it is well know that the resulting variables  $U_j$  follow uniform distribution. The cdf after PIT, is called *copula* and Sklar’s theorem (Sklar, 1959) implies that, under certain regularity conditions, there always exists copula  $C$ ,

$$G(y_1, \dots, y_d) = C(G_1(y_1), \dots, G_d(y_d)). \quad (2.1)$$

When  $G(y_1, \dots, y_d)$  is continuous,  $C(\cdot)$  is unique. If we assume that each  $G_j(y_j)$  is absolutely continuous and strictly increasing,  $C$  can be differentiated and (2.1) becomes

$$g(y_1, \dots, y_d) = c(G_1(y_1), \dots, G_d(y_d)) \prod_{j=1}^d g_j(y_j), \quad (2.2)$$

where  $c(u_1, \dots, u_d) = \partial^d C(u_1, \dots, u_d) / \partial u_1 \dots \partial u_d$  and  $g_j(y_j) = \partial G_j(y_j) / \partial y_j$ ; see Nelsen (2006).

The idea behind copula is that, by using PIT, we split a multivariate distribution into (a) individual behavior of the variables (univariate margins) and (b) remaining joint structure (i.e. dependency) between the variables. With copula, one can easily and flexibly obtain a multivariate distribution that has different univariate margins. Another advantage is that, unlike Kendall’s  $\tau$  and Spearman’s  $\rho_S$  which capture dependency in a single number, copula is a distribution that captures multivariate dependencies. In fact, both Kendall’s  $\tau$  and Spearman’s  $\rho_S$  can be written as functions that only depend on the copula.

**Definition 2.0.1.** Kendall's  $\tau$  for the continuous random variable vector  $(Y_1, Y_2)$  is defined as

$$\tau(Y_1, Y_2) = P[(Y_1 - Y'_1)(Y_2 - Y'_2) > 0] - P[(Y_1 - Y'_1)(Y_2 - Y'_2) < 0]$$

where  $(Y'_1, Y'_2)$  is independent of  $(Y_1, Y_2)$ , but has the same joint distribution as  $(Y_1, Y_2)$ . In short, Kendall's  $\tau$  is the difference between the probabilities of concordance and discordance.

**Theorem 2.0.1.** Let  $(Y_1, Y_2)$  be a continuous random variable vector whose copula is  $C$ , the Kendall's  $\tau$  is then given by

$$\tau(Y_1, Y_2) = 4 \int_0^1 \int_0^1 C(u_1, u_2) dC(u_1, u_2) - 1.$$

*Proof.* Since  $(Y_1, Y_2)$  is continuous,

$$P[(Y_1 - Y'_1)(Y_2 - Y'_2) < 0] = 1 - P[(Y_1 - Y'_1)(Y_2 - Y'_2) > 0].$$

Further,  $(Y_1 - Y'_1)(Y_2 - Y'_2) > 0$  if and only if  $Y_1 > Y'_1$  and  $Y_2 > Y'_2$  or  $Y_1 < Y'_1$  and  $Y_2 < Y'_2$ , and these two events clearly have the same probability. Thus,

$$\begin{aligned} P[(Y_1 - Y'_1)(Y_2 - Y'_2) > 0] &= P[Y_1 > Y'_1, Y_2 > Y'_2] + P[Y_1 < Y'_1, Y_2 < Y'_2] \\ &= 2P[Y_1 < Y'_1, Y_2 < Y'_2]. \end{aligned}$$

Finally, we have

$$\begin{aligned} \tau(Y_1, Y_2) &= P[(Y_1 - Y'_1)(Y_2 - Y'_2) > 0] - P[(Y_1 - Y'_1)(Y_2 - Y'_2) < 0] \\ &= 2P[(Y_1 - Y'_1)(Y_2 - Y'_2) > 0] - 1 \\ &= 4P[Y_1 < Y'_1, Y_2 < Y'_2] - 1 \\ &= 4 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(Y_1 < y_1, Y_2 < y_2) dG(y_1, y_2) - 1 \\ &= 4 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(y_1, y_2) dG(y_1, y_2) - 1 \\ &= 4 \int_0^1 \int_0^1 C(G_1(y_1), G_2(y_2)) dC(G_1(y_1), G_2(y_2)) - 1 \\ &= 4 \int_0^1 \int_0^1 C(u_1, u_2) dC(u_1, u_2) - 1. \end{aligned}$$

□

**Definition 2.0.2.** Spearman's  $\rho_S$  for the random variable vector  $(Y_1, Y_2)$  is defined as

$$\rho_S(Y_1, Y_2) = 3(P[(Y_1 - Y'_1)(Y_2 - Y''_2) > 0] - P[(Y_1 - Y'_1)(Y_2 - Y''_2) < 0])$$

where  $(Y_1, Y_2)$ ,  $(Y'_1, Y'_2)$  and  $(Y''_1, Y''_2)$  are independent pairs from the same joint distribution.

**Theorem 2.0.2.** Let  $(Y_1, Y_2)$  be a random variable vector whose copula is  $C$ , the Spearman's

$\rho_S$  is then given by

$$\rho_S(Y_1, Y_2) = 12 \int_0^1 \int_0^1 C(u_1, u_2) du_1 du_2 - 3.$$

*Proof.*

$$\begin{aligned} \rho_S(Y_1, Y_2) &= \text{cor}(G_1(Y_1), G_2(Y_2)) \\ &= \frac{E[Y_1 Y_2] - E[Y_1]E[Y_2]}{\sqrt{\text{Var}(Y_1)\text{Var}(Y_2)}} \\ &= \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} y_1 y_2 dF(y_1, y_2) - \left(\frac{1}{2}\right)^2}{\sqrt{\left(\frac{1}{12}\right)^2}} \\ &= 12 \int_0^1 \int_0^1 u_1 u_2 dC(u_1, u_2) - 3 \\ &= 12 \int_0^1 \int_0^1 C(u_1, u_2) du_1 du_2 - 3. \end{aligned}$$

□

Some well known copulas and their dependency patterns are visualized in Figure 2.1. Parametric copula families contain the parameter vector  $\theta$  and are steered by it. Figure 2.2 visualizes dependency pattern of Gaussian copula with different parameter values.

Further, all copulas satisfy following properties:

- $0 \leq \min(u_1, \dots, u_d) \leq C(u_1, \dots, u_d) \leq \max(u_1 + \dots + u_d - d + 1, 0) \leq 1$
- $Y_1, \dots, Y_d$  are independent if and only if  $C(u_1, \dots, u_d) = \prod_{j=1}^d u_j$
- If at least one  $u_j = 0$ ,  $C(u_1, \dots, u_d) = 0$
- $C(1, \dots, 1, u_j, 1, \dots, 1) = u_j$

For more details on copula see Nelsen (2006) and Joe (2014).

Copula is often used in statistical applications where modeling the joint distribution of variables is important, such as finance and insurance. For example, if one models the joint distribution between the stock price of company A and company B, copula model can be used to obtain the probability of stock price of company A being lower than certain value, given that the stock price of company B is lower than certain value.

There are also a number of extensions that make copula useful in many other settings such as regression. One of the important extensions of copula is conditional copula, which allows regression between its variables. Patton (2002) extends the existing theories of copula to the conditional copula setting. For more details about conditional copula regression, see Patton (2006), Acar et al. (2013) and Palaro & Hotta (2006). Another important extension is pair-copula constructions (PCC). PCC allows flexible modeling of high-dimensional data by decomposing the copula (of dimension higher than 3) into a set of bivariate copulas. For details of PCC, see Aas & Berg (2009), Aas et al. (2009), Hobæk Haff (2013) and Hobæk Haff et al. (2016).

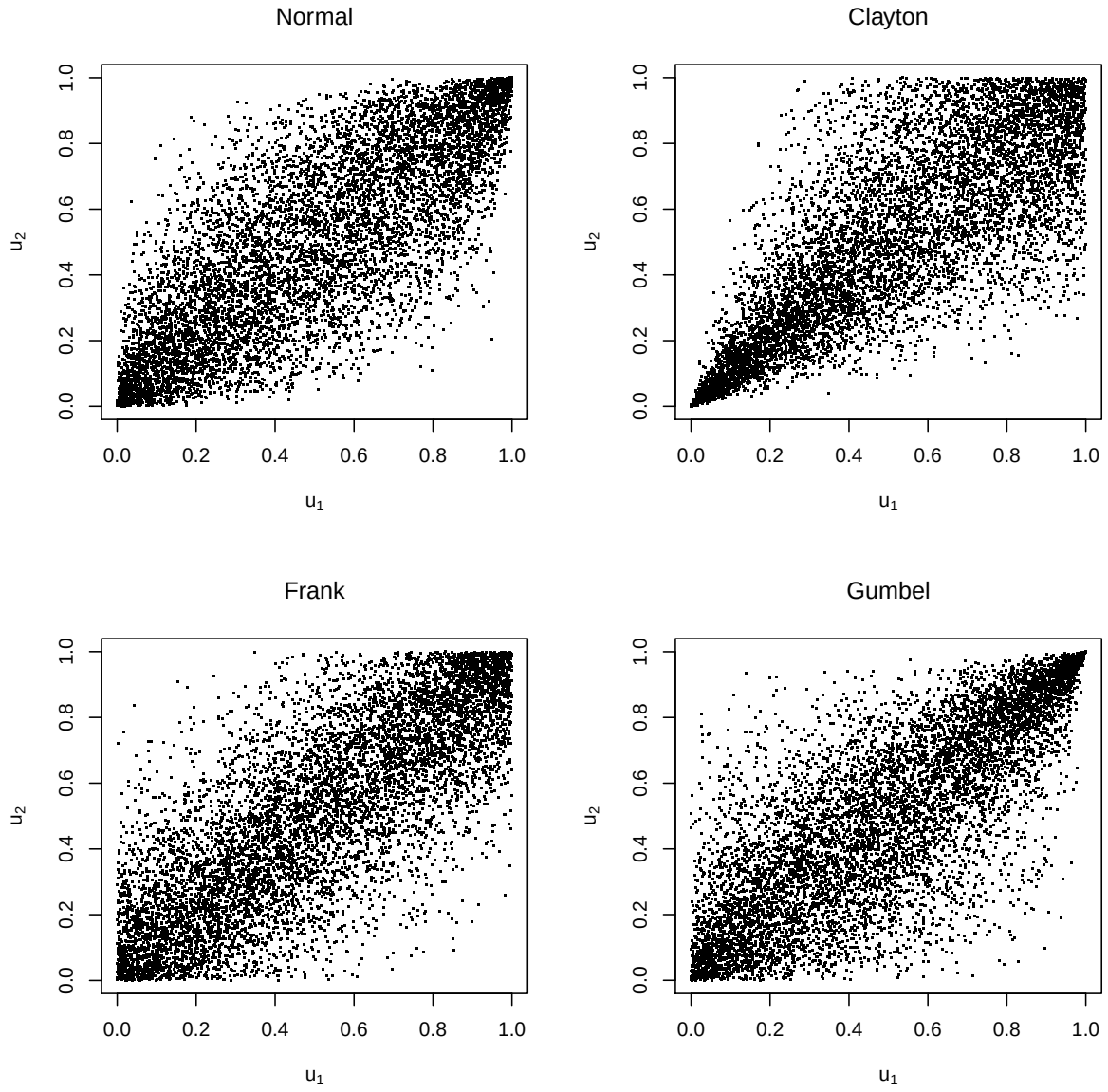


**Figure 2.1:** Dependency structure of four different copulas. The parameter of each copula is chosen such that the simulated data from each copula have Pearson's  $\rho_S$  of about 0.7.

## 2.1. Estimation

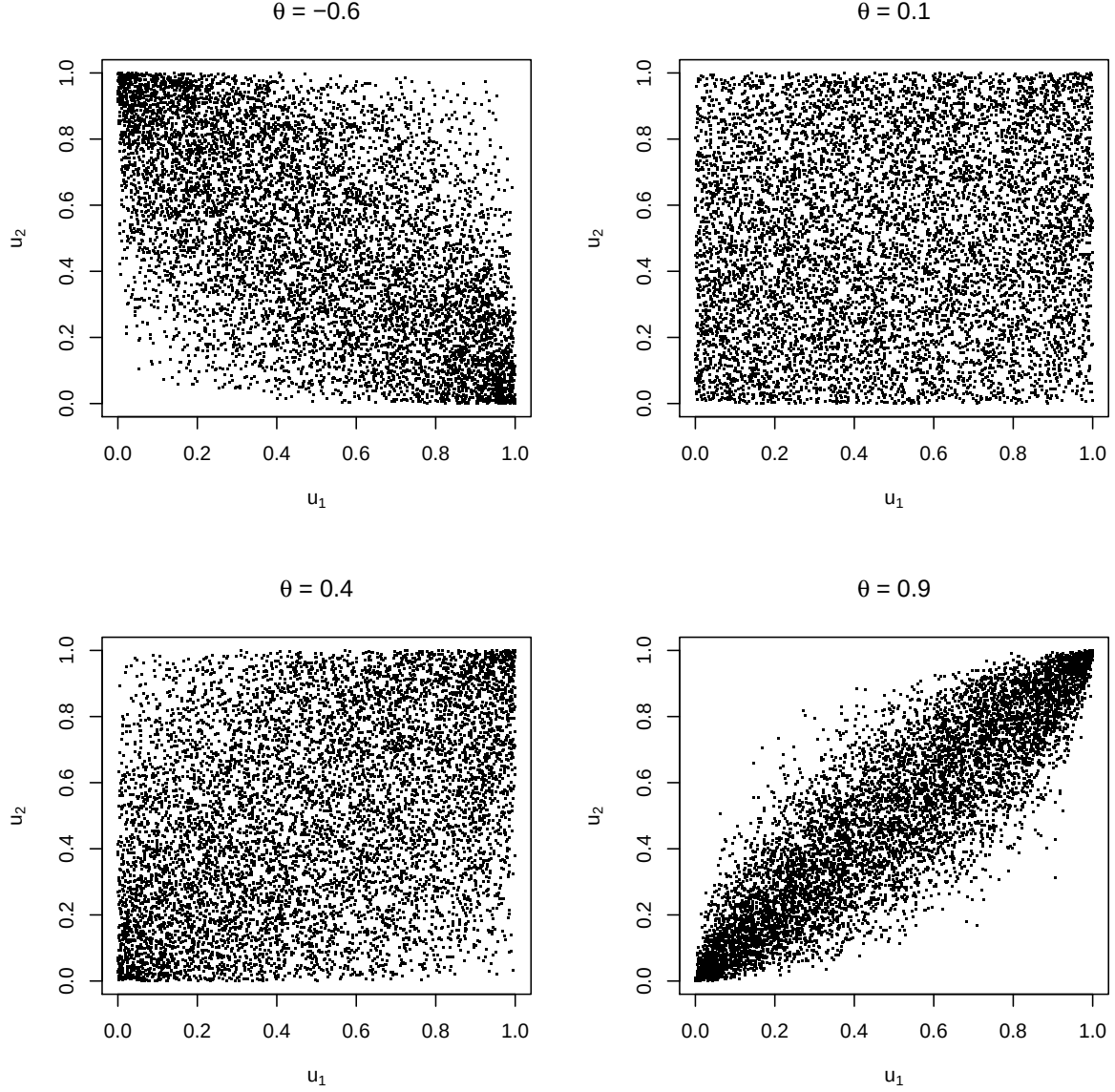


**Figure 2.2:** Dependency structure of Gaussian copula with different parameter values.

## 2.1 Estimation

Let  $y_1 = (y_{1,1}, \dots, y_{1,d})^T, \dots, y_n = (y_{n,1}, \dots, y_{n,d})^T$  be mutually independent observations of  $(Y_1, \dots, Y_d)$ , which is a random variable vector originating from a joint density  $g(y_1, \dots, y_d)$ . Given this data, we wish to find the best fitting copula. The nonparametric way of doing this is using the empirical copula defined as

$$C(u_1, \dots, u_d) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(\hat{G}_1(y_{i,1}) \leq u_1, \dots, \hat{G}_d(y_{i,d}) \leq u_d),$$

where  $\hat{G}_j(y_{i,j}) = \frac{1}{n+1} \sum_{k=1}^n \mathbf{1}(y_{k,j} \leq y_{i,j})$  is pseudo observation, which is simply the data transformed with their own empirical margins.

Another option used often in practice, is using a parametric copula and estimating its parameter vector  $\theta$ . The parametric estimation of  $\theta$  can be divided into two approaches. The first category of approaches is using a maximum-likelihood (ML) based estimation. The second category of approaches is a method of moments type estimator, for example by using the relationship between Kendall's  $\tau$  and  $\theta$ . In this thesis, we focus on the ML based estimation techniques.

### 2.1.1 Maximum likelihood

Consider  $F(y, \eta)$ , our choice of parametric approximation of  $G(y)$ , with  $\eta$  the parameter vector, belonging to some connected subset of the appropriate Euclidean space. Let  $f(y, \eta)$  indicate a density corresponding to  $F(y, \eta)$ .  $F_j(y_j, \alpha_j)$  indicate the  $j$ -th marginal distribution of  $F(y, \eta)$ , where  $\alpha_j$  is the parameter vector of the  $j$ -th margin. Similarly,  $f_j(y_j, \alpha_j)$  is marginal density of  $F_j(y_j, \alpha_j)$ . By using (2.1), we have

$$F(y_1, \dots, y_d, \eta) = C_F(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta), \quad (2.3)$$

with

$$\eta = (\alpha^T, \theta^T)^T = (\alpha_1^T, \dots, \alpha_d^T, \theta^T)^T.$$

Similarly, by using (2.2), we have

$$f(y_1, \dots, y_d, \eta) = c_F(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta) \prod_{j=1}^d f_j(y_j, \alpha_j). \quad (2.4)$$

Then, the ML estimate of  $\hat{\eta}^{\text{ML}}$  can be obtained by maximizing

$$\ell(\eta) = \sum_{i=1}^n [\log f_1(y_{i,1}, \alpha_1) + \dots + \log f_d(y_{i,d}, \alpha_d) + \log c_F(F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta)]. \quad (2.5)$$

For the properties of the ML estimator in a general setting, see for example Le Cam (1990), Casella & Berger (2002), Shao (2003) and Claeskens & Hjort (2008, Chapter 2–3).

Note that we estimate the entire parameter vector  $\eta$  (which consists of  $\alpha$  and  $\theta$ ) with a single ML step, unlike the two-step approach that we will introduce in the next section.

### 2.1.2 Two-stage maximum likelihood

As the dimension  $d$  of the copula gets higher, the ML estimator becomes computationally more demanding and not even feasible when the number of parameters is high (Joe, 1997). To avoid this problem, the two-stage ML is introduced by Shih & Louis (1995). This two-stage ML method is also often referred to as inference function for margins (IFM), a term coined by Xu (1996). The asymptotic behavior and efficiency of this method have been studied both theoretically and numerically in Andersen (2005), Joe (1997), Joe (2005), Kim et al. (2007), Shih & Louis (1995) and Xu (1996).

When parametric families for the copula and the margins are chosen, the two-stage ML estima-

## 2.1. Estimation

tor works as follows.

*Stage 1:* For each  $j \in \{1, \dots, d\}$ , obtain  $\hat{\alpha}_j^{2\text{ML}}$ , the ML estimate of  $\alpha_j$ , by maximizing

$$\ell_{f_j} = \sum_{i=1}^n \log f_j(y_{i,j}, \alpha_j),$$

with respect to  $\alpha_j$ .

*Stage 2:* Plugging in  $\hat{\alpha}_1^{2\text{ML}}, \dots, \hat{\alpha}_d^{2\text{ML}}$  from Stage 1, into (2.5) gives

$$\begin{aligned} \ell(\hat{\alpha}^{2\text{ML}}, \theta) &= \sum_{i=1}^n [\log f_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}) + \dots + \log f_d(y_{i,d}, \hat{\alpha}_d^{2\text{ML}}) \\ &\quad + \log c_F(F_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}), \dots, F_d(y_{i,d}, \hat{\alpha}_d^{2\text{ML}}), \theta)]. \end{aligned}$$

Then,  $\theta$  is estimated by maximizing  $\ell(\hat{\alpha}^{2\text{ML}}, \theta)$  with respect to  $\theta$ , yielding

$$\hat{\theta}^{2\text{ML}} = \arg \max_{\theta} \ell(\hat{\alpha}^{2\text{ML}}, \theta) = \arg \max_{\theta} \ell_{c_F}(\hat{\alpha}^{2\text{ML}}, \theta),$$

where

$$\ell_{c_F}(\alpha, \theta) = \sum_{i=1}^n \log c_F(F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta).$$

The two-stage ML estimate  $\hat{\eta}^{2\text{ML}} = ((\hat{\alpha}^{2\text{ML}})^T, (\hat{\theta}^{2\text{ML}})^T)^T = ((\hat{\alpha}_1^{2\text{ML}})^T, \dots, (\hat{\alpha}_d^{2\text{ML}})^T, (\hat{\theta}^{2\text{ML}})^T)^T$  obtained in this way satisfies

$$\left( \frac{\partial}{\partial \hat{\alpha}_1^{2\text{ML}}} \ell_{f_1}, \dots, \frac{\partial}{\partial \hat{\alpha}_d^{2\text{ML}}} \ell_{f_d}, \frac{\partial}{\partial \hat{\theta}^{2\text{ML}}} \ell_{c_F} \right) = (0, \dots, 0).$$

### 2.1.3 Pseudo maximum likelihood

One of the drawbacks of (two-stage) ML estimation is that the estimation of the copula parameter  $\theta$  is affected by the choice of parametric distribution for the margins. Even when the copula is correctly specified, possible misspecifications of the margins can lead to suboptimal values of  $\hat{\theta}^{\text{ML}}$  or  $\hat{\theta}^{2\text{ML}}$ . Genest et al. (1995) proposed the pseudo maximum likelihood (PML) estimator that overcomes this shortcoming. In this semi-parametric estimation scheme, one first uses the pseudo-observations  $\hat{G}_j(y_{i,j}) = \frac{1}{n+1} \sum_{k=1}^n \mathbf{1}(y_{k,j} \leq y_{i,j})$ , which are nonparametric probability integral transforms of the data. Subsequently, one plugs these into the copula log-likelihood and maximizes it with respect to  $\theta$  and obtain  $\hat{\theta}^{\text{PML}}$ . For details concerning the PML estimator, see Tsukahara (2005), Chen & Fan (2005) and Kim et al. (2007).

Once  $\theta$  is estimated, one can keep the nonparametric margins. However, parametric margins are often used instead. In this case, one chooses parametric forms for each  $f_j(y_{i,j}, \alpha_j)$  and estimate  $\alpha_j$  for each margin separately by ML, just like in the stage 1 of the two-stage ML estimation (Hobæk Haff et al., 2015; Bevacqua et al., 2017). We abbreviate this ‘PML combined with parametric margins’ by ‘PMLpm’.

### 2.1.4 Example

To give a better idea on how the estimations schemes discussed in Sections 2.1.1–2.1.3 work, we illustrate a bivariate case with a subset of the daily precipitation data from the Norwegian Meteorological Institute. The subset we used consist of daily measurements of precipitation in millimeters at two meteorological stations in Lørenskog and Nannestad, Norway from January 1, 1990 to December 31, 2006. The subset contains only the days when there was precipitation in both cities. The original dataset was used previously in Aas & Berg (2009) and Hobæk Haff (2013). Figure 2.3 shows the scatter plot of the dataset.

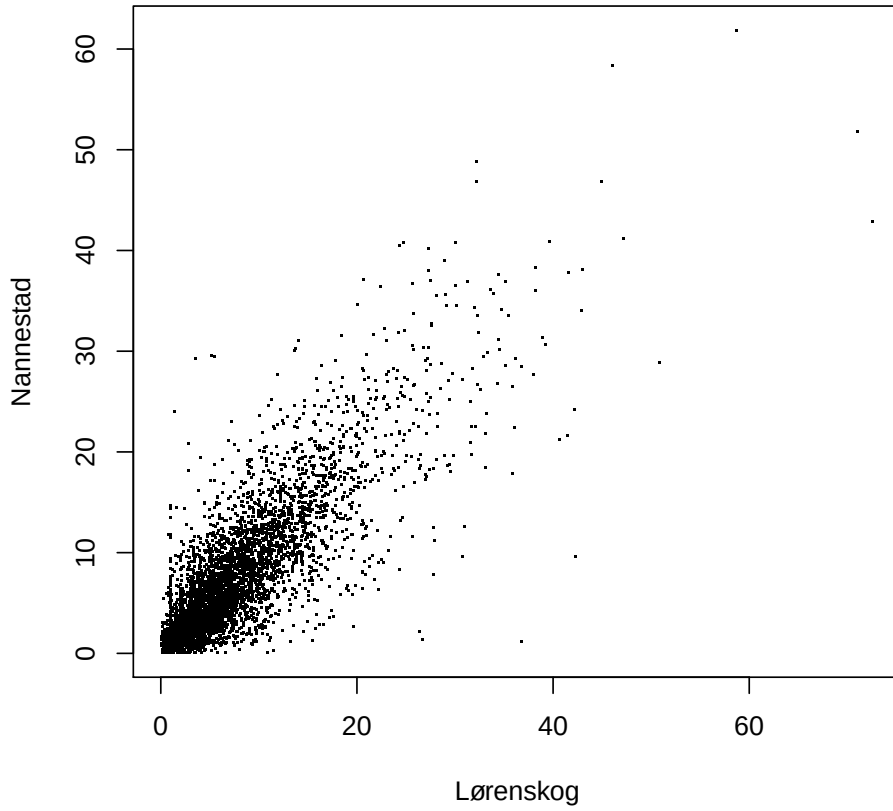


**Figure 2.3:** Visualization of the precipitation data measured in meteorological stations in Lørenskog and Nannestad. There are total 5536 observations.

Imagine that we decided to use Gamma distribution to model the precipitation in Lørenskog and Weibull distribution to model the precipitation in Nannestad. We will use Gumbel copula to model the joint distribution of the two.

The log-likelihood in this case is

$$\ell(\eta) = \sum_{i=1}^n [\log f_1(y_{i,1}, \alpha_1) + \log f_2(y_{i,2}, \alpha_2) + \log c_F(F_1(y_{i,1}, \alpha_1), F_2(y_{i,2}, \alpha_2), \theta)].$$

## 2.1. Estimation

where

$$\eta = (\alpha^T, \theta^T)^T = (\alpha_1^T, \alpha_2^T, \theta^T)^T.$$

### Maximum likelihood

We estimate the entire parameter vector  $\eta$  at once by maximizing  $\ell(\eta)$  with respect to  $\eta$ . The obtained ML estimate is  $\hat{\eta}^{\text{ML}} = (1.28, 0.18, 1.08, 7.30, 2.65)^T$ .

### Two-stage maximum likelihood

*Stage 1:*

Estimate  $\alpha_1$  by maximizing

$$\ell_{f_1} = \sum_{i=1}^n \log f_1(y_{i,1}, \alpha_1),$$

with respect to  $\alpha_1$ . The obtained estimate is  $\hat{\alpha}_1^{2\text{ML}} = (1.35, 0.20)^T$ .

Estimate  $\alpha_2$  by maximizing

$$\ell_{f_2} = \sum_{i=1}^n \log f_1(y_{i,2}, \alpha_2),$$

with respect to  $\alpha_2$ . The obtained estimate is  $\hat{\alpha}_2^{2\text{ML}} = (1.09, 7.13)^T$ .

*Stage 2:*

Plugging in  $\hat{\alpha}_1^{2\text{ML}}$  and  $\hat{\alpha}_2^{2\text{ML}}$  that we obtained in Stage 1, into  $\ell(\eta)$  and removing the parts that do not contain  $\theta$  gives

$$\ell_c(\theta) = \sum_{i=1}^n [\log c_F(F_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}), F_2(y_{i,2}, \hat{\alpha}_2^{2\text{ML}}), \theta)].$$

Now, estimate  $\theta$  by maximizing  $\ell_c(\theta)$  with respect to  $\theta$ . The obtained two-stage ML estimate is  $\hat{\theta}^{2\text{ML}} = 2.56$ .

Finally, we can merge  $\hat{\alpha}_1^{2\text{ML}}$ ,  $\hat{\alpha}_2^{2\text{ML}}$  and  $\hat{\theta}^{2\text{ML}}$  into one vector and obtain the two-stage ML estimate  $\hat{\eta}^{2\text{ML}} = (1.35, 0.20, 1.09, 7.13, 2.56)^T$ .

### Pseudo maximum likelihood

Perform PIT on the marginal variables nonparametrically with  $\hat{G}_j(y_{i,j}) = \frac{1}{n+1} \sum_{k=1}^n \mathbf{1}(y_{k,j} \leq y_{i,j})$  and plug this into  $\ell_c(\theta)$  and obtain

$$\ell_c^{\text{PML}}(\theta) = \sum_{i=1}^n \left[ \log c_F \left( \hat{G}_1(y_{i,1}), \hat{G}_2(y_{i,2}), \theta \right) \right].$$

Now, estimate  $\theta$  by maximizing  $\ell_c^{\text{PML}}(\theta)$  with respect to  $\theta$ . The obtained PML estimate is  $\hat{\theta}^{\text{PML}} = 2.59$ .

### 3 Model selection

When data are given, we usually have multiple candidates for a model. In many cases, it is not clear which of the models is the ‘best’ one and there is need for a systemic method for evaluating and selecting the model that is most appropriate for the situation. For example, Figure 3.1 shows a histogram of given data. The three lines indicate density curves of three different probability distributions (log-normal, gamma and Weibull) fitted to the data. From visual inspection, it is not clear which of the three suits the data best.

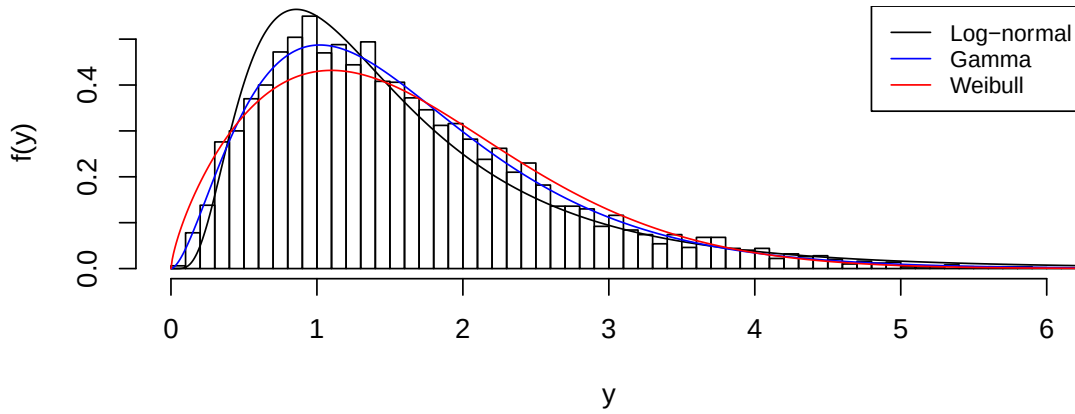


**Figure 3.1:** Visualization of three candidate distributions for modeling the given data.

There are many approaches for evaluating models such as goodness of fit tests and significance testing. We focus on model selection using model selection criteria.

Model selection criteria assign a score to each candidate model according to certain criteria. Then, the model with the lowest or highest score, depending on how the model selection criterion is defined, is chosen as the ‘best’ model. Note that model selection criteria are used to compare candidate models and they have to be interpreted in relative sense. The ‘best’ candidate model according to a model selection criterion from a given list of models, is not necessarily a ‘good’ model. It depends on the quality of the models on the list.

#### 3.1 KL divergence based model selection criteria

Consider the situation that we have data generated from an unknown true density model  $g(y)$  and we have some parametric candidate models  $f(y, \eta)$ . The Kullback–Leibler (KL) divergence (Kullback & Leibler, 1951) measures the extent to which one probability distribution diverges

### 3.2. Focused information criterion (FIC)

from another. The KL divergence from  $g$  to  $f$  is defined as

$$\text{KL}\{g, f(\cdot, \eta)\} = \int g(y) \log \frac{g(y)}{f(y, \eta)} dy.$$

Many of the well known model selection criteria are aiming for the model that minimizes the KL divergence. Important to note is that the true density  $g$  is the same for all models, because all the candidate models are used for modeling the same data. Thus, minimizing the KL divergence is equivalent to maximizing  $\int g(y) \log f(y, \eta) dy$ . Based on this observation, one could argue that the maximized likelihood value of each model, computed with given data, can be used as a model selection criterion. It turns out, however, that the maximized likelihood is a biased estimator of  $\int g(y) \log f(y, \eta) dy$ .

Identifying and subtracting the bias and scaling the bias-corrected likelihood with a certain constant, gives Takeuchi Information Criterion (TIC) (Takeuchi, 1976; Claeskens & Hjort, 2008)

$$\text{TIC} = 2\ell(\hat{\eta}^{\text{ML}}) - 2\hat{p}_{\eta, \text{TIC}}^*, \quad (3.1)$$

where  $\hat{p}_{\eta, \text{TIC}}^* = \text{tr}(\hat{\mathcal{I}}_{\eta}^{-1} \hat{K}_{\eta})$ . See Takeuchi (1976) for the exact definition of the quantities. Note that the bias correction term  $\hat{p}_{\eta, \text{TIC}}^*$  assumes that the parameters of the model are estimated with the ML estimator. Using any other estimator, including the two-stage ML and the PML estimator, invalidates the formula of TIC.

#### 3.1.1 True model assumption

Many of the theories and tools developed for the ML estimator are based on so-called ‘true model assumption’. This assumption means that the candidate model is correctly specified. In other words,  $g(y) = f(y, \theta_0)$  for a certain parameter value  $\theta_0$ . Under the true model assumption, we have  $\mathcal{I}_{\eta} = K_{\eta}$  and the bias correction term of TIC simplifies to  $p$ , which is the number of parameters in the model.

The simplified version of TIC under the true model assumption is called AIC (Akaike, 1974) and defined as

$$\text{AIC} = 2\ell(\hat{\eta}^{\text{ML}}) - 2p. \quad (3.2)$$

AIC is probably the most famous and often used model selection criterion in statistics. Due to the popularity of AIC, TIC is also often called as model robust AIC.

### 3.2 Focused information criterion (FIC)

TIC, AIC and also other model selection criteria such as BIC (Schwarz et al., 1978) and DIC (Spiegelhalter et al., 2002) measure how close a candidate model is to the true model in terms of a certain divergence measure. However, in many cases, the model itself is not the goal, but rather a stepping stone to estimate a specific quantity, such as the mean or the probability of certain event. In this setting, one is often not interested in how close the candidate model is

to the true model, but how close the estimated quantity is to the true quantity. The main idea behind the FIC, first introduced by Claeskens & Hjort (2003) and later modified by Jullum & Hjort (2017, 2018) is that one wants to select a model that is good for a specific task. This specific task is quantified as estimating a focus parameter, say  $T(G)$ , a functional of the data generating distribution. The functional needs to have a clear statistical interpretation across candidate models. Let  $F$  be the distribution function of the candidate model.  $T(\hat{F})$  is then the focus parameter computed with the estimated candidate model.

The FIC, aims for the model that minimizes the risk function of the focus parameter with quadratic loss function. It is well known that

$$R(T(G), T(\hat{F})) = E_G \left[ (T(G) - T(\hat{F}))^2 \right] = \text{MSE}(T(\hat{F})) = \text{Var}_G \left( T(\hat{F}) \right) + \text{bias}^2 \left( T(\hat{F}), T(G) \right).$$

The FIC, in its general form, is defined as the estimated version of MSE

$$\text{FIC} = \widehat{\text{MSE}}(\hat{T}_{\text{cnd}}) = \widehat{\text{Var}}_G \left( \hat{T}_{\text{cnd}} \right) + \widehat{\text{bias}}^2 \left( \hat{T}_{\text{cnd}}, T_{\text{true}} \right), \quad (3.3)$$

where  $\hat{T}_{\text{cnd}} = T(\hat{F})$  and  $T_{\text{true}} = T(G)$ .

There are two different versions of FIC. The ‘original’ FIC from (Claeskens & Hjort, 2003, 2008) and the ‘new’ FIC from Jullum & Hjort (2017). The difference between the two is in the the bias term  $\widehat{\text{bias}}^2$ . The ‘original’ FIC is designed for locally misspecified models. In this setting, one assumes that there exists an unbiased ‘wide’ parametric model and all the candidate models are considered as nested models of this ‘wide’ model. This means that each candidate model can be obtained from the ‘wide’ model by fixing some of its parameters to some proper ‘null values’. The obvious disadvantage of this approach is that we need to find such a wide model and that every candidate model should be a nested version of this wide model.

The ‘new’ FIC is designed to overcome this issue. Instead of using the parametric ‘wide’ model, it assumes that there exists an asymptotically unbiased model that does not need to contain the candidate models as its special cases. This means that this model can also be a nonparametric or semi-parametric model. Jullum & Hjort (2017) use nonparametric model as such a ‘broad’ model and in this thesis, we stick to this canonical choice.

Once we have a nonparametric model that is asymptotically unbiased, the  $\widehat{\text{bias}}^2$  term for each candidate model can in principle be estimated as the squared difference between  $\hat{T}_{\text{np}}$  and  $\hat{T}_{\text{cnd}}$ , where  $\hat{T}_{\text{np}}$  is the focus parameter estimated with the nonparametric model. However, in general,  $\widehat{\text{bias}}^2 \neq \widehat{\text{bias}}^2$  and it turns out that this naive estimator tends to overshoot its estimand with  $\text{Var} \left( \hat{T}_{\text{cnd}} - \hat{T}_{\text{np}} \right)$  (Jullum & Hjort, 2017). Compensating for the overestimation gives

$$\widehat{\text{bias}}^2 \left( \hat{T}_{\text{cnd}}, T_{\text{true}} \right) = \left( \hat{T}_{\text{cnd}} - \hat{T}_{\text{np}} \right)^2 - \widehat{\text{Var}}_G \left( \hat{T}_{\text{cnd}} \right) - \widehat{\text{Var}}_G \left( \hat{T}_{\text{np}} \right) + 2 \widehat{\text{Cov}}_G \left( \hat{T}_{\text{np}}, \hat{T}_{\text{cnd}} \right)$$

as the estimate of the squared bias. Yet, in practice, it can happen that this estimate is a negative

### 3.2. Focused information criterion (FIC)

value. Truncating for the possible negative cases gives

$$\widehat{\text{bias}}^2(\hat{T}_{\text{cnd}}, T_{\text{true}}) = \max \left\{ 0, (\hat{T}_{\text{cnd}} - \hat{T}_{\text{np}})^2 - \widehat{\text{Var}}_G(\hat{T}_{\text{cnd}}) - \widehat{\text{Var}}_G(\hat{T}_{\text{np}}) + 2\widehat{\text{Cov}}_G(\hat{T}_{\text{np}}, \hat{T}_{\text{cnd}}) \right\}.$$

With this bias estimate, the ‘new’ FIC is defined as

$$\begin{aligned} \text{FIC} &= \widehat{\text{Var}}_G(\hat{T}_{\text{cnd}}) + \widehat{\text{bias}}^2(\hat{T}_{\text{cnd}}, T_{\text{true}}) \\ &= \begin{cases} \widehat{\text{Var}}_G(\hat{T}_{\text{cnd}}) + \max \left\{ 0, (\hat{T}_{\text{cnd}} - \hat{T}_{\text{np}})^2 - \widehat{\text{Var}}_G(\hat{T}_{\text{cnd}}) \right. \\ \quad \left. - \widehat{\text{Var}}_G(\hat{T}_{\text{np}}) + 2\widehat{\text{Cov}}_G(\hat{T}_{\text{np}}, \hat{T}_{\text{cnd}}) \right\} & \text{if the model is (semi)parametric} \\ \widehat{\text{Var}}_G(\hat{T}_{\text{cnd}}) & \text{if the model is nonparametric.} \end{cases} \end{aligned} \tag{3.4}$$

Note that this derivation of the FIC is done without specifying the estimation scheme. As long as the variance term  $\widehat{\text{Var}}_G(\hat{T}_{\text{cnd}})$  and the bias term  $\widehat{\text{bias}}^2(\hat{T}_{\text{cnd}}, T_{\text{true}})$  can be obtained, the FIC can be defined for any model and any estimation scheme.

## 4 Neural networks

In recent years, neural networks, also known under the popular name ‘deep learning’, have achieved state-of-the-art performance in wide range of tasks like computer vision and natural language processing and have gained big popularity (Goodfellow et al., 2016). From statisticians perspective, neural networks can be seen as a multilayer non-linear regression, where its parameters are estimated with an iterative algorithm called stochastic gradient descent (SGD). We briefly explain the key building blocks of neural networks.

### 4.1 Convolution layer

Consider a neural network that takes explanatory variables  $\mathbf{X}$  as input. Often, the first operation applied to  $\mathbf{X}$  is *convolution*. There are multiple versions and variations of convolution in the context of neural network. Here, we focus on so-called 2D convolution, which is a canonical option for image data.

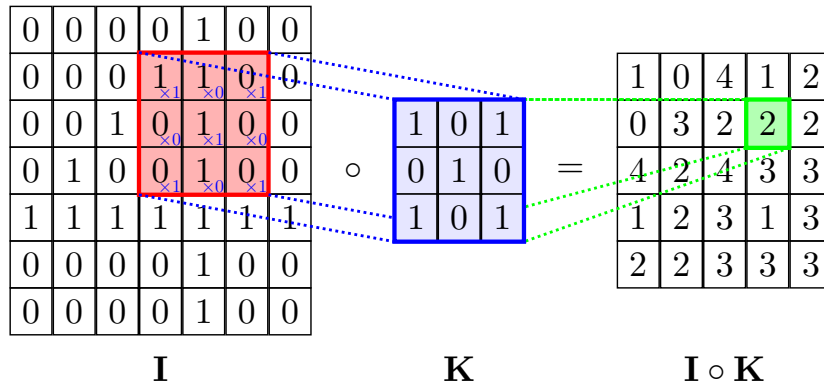


**Figure 4.1:** Visualization of the convolution operation described in Section 4.1 with stride=1, depth=1 and zero-padding=0.  $\mathbf{I}$  denotes an input to the layer and  $\mathbf{K}$  indicates the kernel.  $\mathbf{I} \circ \mathbf{K}$  indicates the resulting feature map resulted by the convolution operation with given  $\mathbf{I}$  and  $\mathbf{K}$ . This figure is generated by adapting the code from Veličković (2016).

A convolution layer has a kernel (a.k.a. filter), which is a set of parameters to be estimated/trained/learned. Figure 4.1 shows how a convolution layer with a filter  $\mathbf{K}$  of size  $3 \times 3$  processes a two-dimensional input  $\mathbf{I}$  of size  $7 \times 7$ . We begin at the upper-left region of  $\mathbf{I}$  and compute the Hadamard product between the filter  $\mathbf{K}$  and  $3 \times 3$  regions of  $\mathbf{I}$  by sliding (more precisely, convolving) the filter  $\mathbf{K}$  across the width and the height of the input. As we slide the filter over the input, we obtain a two-dimensional activation map (i.e. output of a layer) of size  $5 \times 5$ . Intuitively speaking, the parameter values of the kernel will be ‘learned’ during the training process with SGD such that the resulting activation map of the convolution layer is so-called ‘extracted feature’, which is a meaningful input to the next layer. Typically, multiple kernels are used simultaneously on a given input and a set of feature maps will be passed to the next layer.

## 4.2. Pooling

The convolution operation also can be specified further with *stride* (i.e. how much we ‘slide’ the kernel at a time), *zero-padding* (i.e. adding zeros around the border of input to prevent that the information on edges are lost), *depth* (i.e. number of kernels) and more. For more details on convolution, see for example Dumoulin & Visin (2016).

## 4.2 Pooling

A typical modern modern neural network has easily over multiple millions of parameters and running a neural network is both computationally and memory intensive, even with the use of Graphics Processing Unit (GPU). To reduce the number of parameters in the model, it is common to decrease the size of feature maps from a convolution layer by using *pooling*, which is a form of non-linear down-sampling. The most commonly used form of pooling is *max pooling*. It partitions its input into a set of sub-regions and computes maximum for each sub-region. Figure 4.2 shows max pooling with filter size  $2 \times 2$  applied with a stride of 2. It returns the max value of every 4 numbers in each  $2 \times 2$  sub-region. This reduces the size of the feature map by a factor of 4. *Average pooling* is similar to max pooling, but it takes average of the numbers in each sub-region. It used to be used often, but fallen out of favor since max pooling often works better in practice.

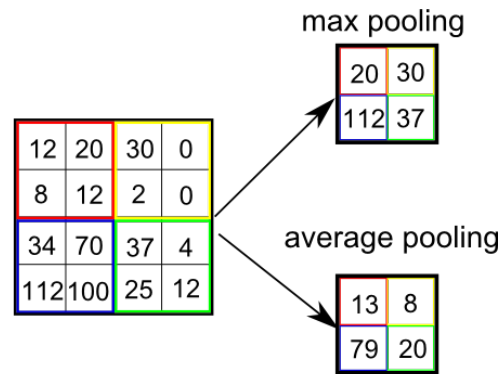


**Figure 4.2:** Visualization of the max pooling and average pooling described in Section 4.2 with the filter size  $2 \times 2$  and stride of 2. This figure is generated by adapting the code from Singhal (2017).

## 4.3 Fully connected layer

*Fully connected layer* can be seen as a regression through multiple layers of latent variables. Figure 4.3 visualizes a fully connected layer with a 5 inputs, a single hidden layer with 3 ‘neurons’ and a output layer with 1 neuron. Each neuron is computed as a linear combination of its inputs obtained by multiplying them with their ‘weight’ parameters and adding an intercept term. (This intercept term is, confusingly enough, called ‘bias’ in the field of neural network.) Typically, the fully connected layer comes right after a convolution layer and takes the ‘flattened’ (i.e. converted into a vector) feature map from the convolution layer as its input. In modern neural networks, the largest part of the parameters are in fully connected layers. For example, consider a fully connected layer that is placed after a convolution layer returning a feature map of size  $32 \times 32 \times 3$ . The fully connected layer then gets the flattened version of this feature

map which is a vector of size 3072. If there is a single hidden layer with 1000 neurons and an output layer with 10 neurons, there are total 3,083,010 ( $= 3072 \times 1000 + 1000 + 1000 \times 10 + 10$ ) parameters. Considering that modern neural networks usually contain multiple hidden layers, the number of parameters in fully connected layers is much higher in practice. For instance, VGG16 (Simonyan & Zisserman, 2014) architecture has total 138 million parameters and 89% of them are from the fully connected layers.

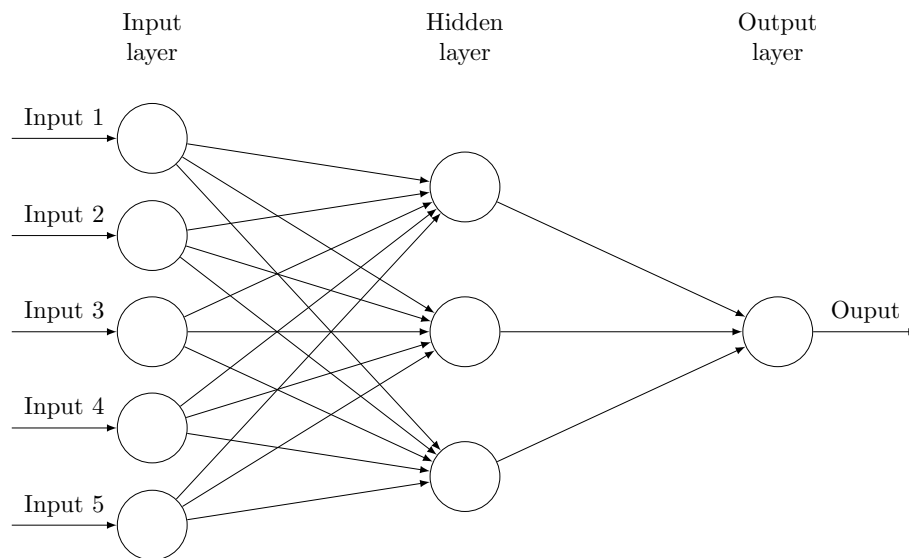


**Figure 4.3:** Visualization of a fully connected layer with one hidden layer. This figure is generated by adapting the code from Wibrow (2014).

## 4.4 Activation function

*Activation function* is utilized to insert non-linearity in neural networks. Its role is highly similar to that of the inverse link function in GLM. Typically, activation function  $h(\cdot)$  is inserted between layers. For example, it is often applied to the output of each hidden layer within a fully connected layer and also after each convolution layer.

Historically, sigmoid function  $h(x) = 1/(1+e^{-x})$  (a.k.a. logistic function) and  $h(x) = \tanh(x)$  were most common choices of activation function. However, in recent years, the canonical choice has moved to Rectified Linear Unit (ReLU), which is defined as  $h(x) = \max(0, x)$ . ReLU tends to accelerate the convergence of SGD, compared to the sigmoid and tanh functions (e.g. improvement with a factor of 6 in Krizhevsky et al. (2012)). Another advantage is that ReLU is much simpler operation than the sigmoid and tanh function and thus faster in computation.

## 4.5 Universal approximation theorem

The fully connected layer described in Section 4.3 might seem unnatural and redundant to whom are used to linear model, generalized linear model, generalized additive model and other regres-

## 4.6. Stochastic gradient descent

sion models that parameterize the relationship between the dependent and independent variables without a multilayer structure. However, it turns out that the multilayer structure is highly suitable for estimating an arbitrary non-linear continuous function. Cybenko (1989) proved that a fully connected layer containing a single hidden layer and finite number of neurons, combined with the sigmoid activation function is so-called *universal estimator*, which means that it can estimate any continuous function to any desired precision. This is called universal approximation theorem and later proved by Hornik (1991) for more general activation functions.

## 4.6 Stochastic gradient descent

Once the parametric form of a network is set with the building blocks from the previous sections, for example the one displayed in Figure 4.4, its parameter values are to be estimated/trained/learned via *gradient descent* (GD). GD is an iterative refinement algorithm for finding parameter values that minimizes the given loss function (e.g. negative log-likelihood, cross entropy or mean squared error). Before the algorithm starts, the parameters of the model are initialized to some initial values. Then, at each iteration, the derivatives of the loss function with respect to its parameters are computed. This gradient, obtained for each parameter, is multiplied with a constant called ‘learning rate’ and subtracted from the parameter values. The intuition behind GD is that the gradients, which can be interpreted as slopes in a  $n$ -dimensional plane, indicates in which direction and speed we have to ‘descend’ to arrive at a (local) minimum point.

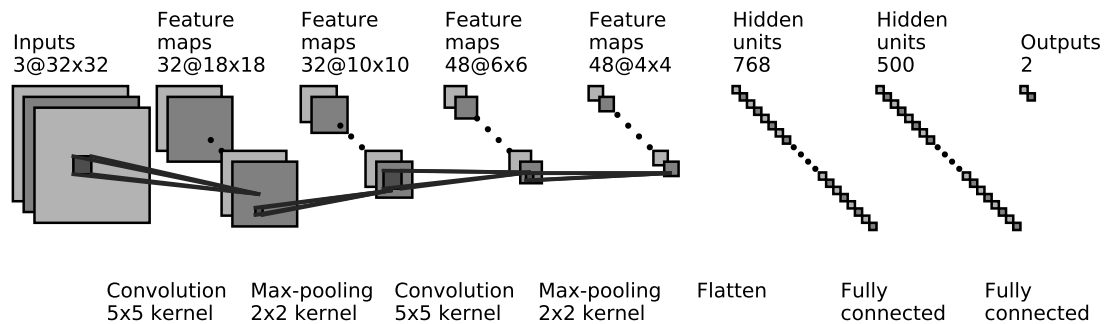


**Figure 4.4:** An example of a feedforward neural network with building blocks from Section 4.1 to 4.4. This figure is generated by adapting the code from Ding (2018).

Computing the value of loss function with given data is called *forward pass* and computing gradients w.r.t. each parameter is called *backward pass*. Obtaining gradients by using numerical differentiation is computationally heavy and can be even infeasible for intricate functions. One of the core components of deep learning software like TensorFlow (Abadi et al., 2016) and PyTorch (Paszke et al., 2017) is so called ‘automatic differentiation’, which keeps track of the operations performed during the forward pass and obtains the gradients analytically during the backward pass by using the chain rule. This makes backward pass efficient and feasible even for deep networks with large number of layers.

There are several variations of GD. When the loss function is computed with all training data

points to compute gradients (and update the parameters), it is called *gradient descent*. This is the most ideal version of the algorithm since it utilizes all the data points. However, this is often infeasible in practice due to high memory requirement when the number of data points is high (large image datasets can easily contain millions of examples). *Stochastic gradient descent* (SGD) was introduced to overcome this problem. At each iteration, the loss function (and thus its gradient and also the parameter update) is computed with a single data point. As one can imagine, SGD can lead to sub-optimal results. *Mini-batch gradient descent* (MGD) is a compromise between GD and SGD. The training data are divided into ‘mini-batches’ and at each iteration, the loss function is computed by using one mini-batch. At the moment, MGD is the canonical choice for training a neural network. Note that many people use the term SGD even when they are referring to MGD.

In practice, the ‘vanilla’ MGD is rarely used. Instead, one of the modified versions of MGD is used for better results. Adam (Kingma & Ba, 2015) and RMSprop (Tieleman & Hinton, 2012) are some of the popular choices.

## 4.7 Pruning

Since LeNet-5 from LeCun et al. (1998), which is one of the first models that showed the potential of neural networks, people started using bigger and deeper models with many layers and neurons. Although large architectures can lead to better prediction performance, they require significant computational and memory resources. Denil et al. (2013) showed that there is significant redundancy in the parameterization of many deep neural network models. This overparameterization not only requires unnecessary computational resources, but can also lead to an overfitting.

In the field of computer science, various approaches have been suggested to reduce the computational footprint of large neural network models. One approach is quantization, which aims to reduce the numerical precision required to represent parameters (or neurons). For instance, Vanhoucke et al. (2011), Hwang & Sung (2014) and Gong et al. (2014) compress networks by using 8-bit precision, 3-bit precision, and vector quantizations, respectively. Courbariaux et al. (2015), Hubara et al. (2017), Lin et al. (2016), Rastegari et al. (2016) extend quantization based methods further. Another strategy is to reduce the computational burden by using low-rank decompositions of tensors (Denton et al., 2014; Lebedev et al., 2014; Jaderberg et al., 2014). See Cheng et al. (2017) for an overview of different methods.

Although the approaches mentioned above can help reducing computational burden, it does not deal with overparameterization and overfitting directly. From statistical perspective, most natural approach would be *pruning*. i.e. Identifying redundant parameters and removing it. Figure 4.5 shows how the fully connected layer from Figure 4.3 would look like after such a pruning process. Note that removing neurons is a special case of parameter pruning (i.e. removing all the arrows to and from the concerning neuron).

In statistics, there are well developed methods for testing whether a specific parameter is redundant or not. Most obvious approach would be performing a hypothesis testing on each parameter with a proper null value as null hypothesis. After this, one can remove the param-

## 4.7. Pruning

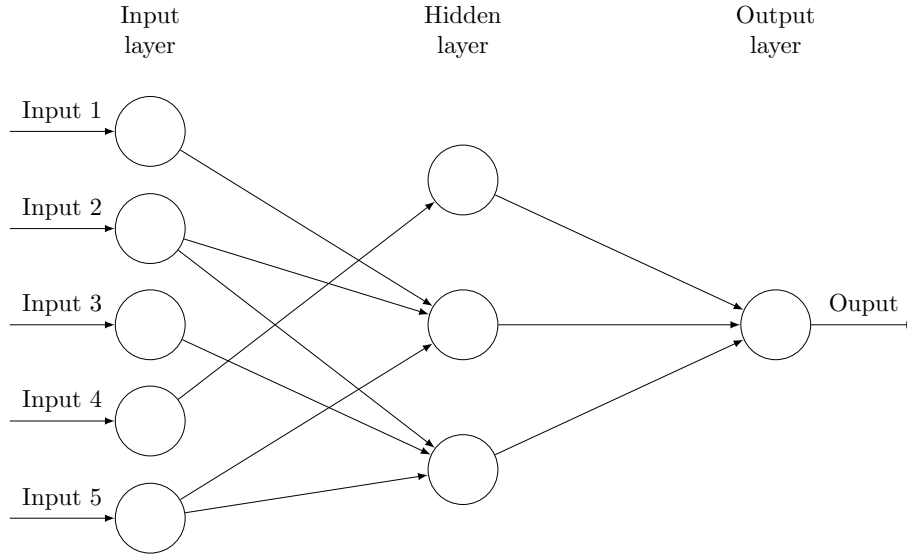


**Figure 4.5:** Pruned version of the fully connected layer from Figure 4.3. This figure is generated by adapting the code from Wibrow (2014).

ters above certain threshold. Another approach would be fitting nested models (e.g. the original model and one without the parameter of interest) and compute a model selection criterion such as AIC and BIC to compare them. Ideally, one would like to use similar approaches for evaluating the redundancy of parameters in a neural network. (Yet, if one is interested in evaluating each parameter individually, the second approach can be impractical since it will require fitting extremely large number of models.)

However, those approaches cannot be used directly for neural networks. The hypothesis testing approach (e.g. Wald test) is based on the asymptotic normality of the estimated parameter values. The problem is that, at the moment, there is almost no work done for establishing asymptotic properties of the modified versions of the MGD. Recently, Toulis et al. (2017) established the asymptotic normality of the vanilla SGD procedure, including an analytical expression for the covariance matrix. Although their work is a great step forward towards understanding the asymptotic properties of the SGD estimator, its result cannot be directly applied to the non-vanilla versions of SGD such as MGD, Adam and RMSprop. In addition, they had to make some strong assumptions like convexity, which is almost never the case for neural networks.

The approach based on the model selection between nested models is also not directly applicable for neural networks, since the theories from maximum likelihood estimation and from Toulis et al. (2017) do not hold for MGD estimator, especially when they are used with training strategies like Dropout (Srivastava et al., 2014) and Dropconnect (Wan et al., 2013). Even if we do have required analytical results, computing necessary quantities will be highly challenging. For example, for a model with 100 million parameters, inverting its Hessian matrix is infeasible for most of the hardwares available on the market at the moment. The approach from Hassibi et al. (1993) is thus also infeasible in practice.

Naturally, there have been several attempts on heuristically identifying redundant parameters

without using proper asymptotic properties. One of the most successful works is Han et al. (2015). Their idea is to prune the network parameters based on their magnitude. This is achieved by ranking the parameters based on their absolute value and removing the ones above certain threshold. Although this method might look, especially from statistical perspective, rather ungrounded and simple, it gives state-of-the-art results and used in real world applications. There have been several extensions of this approach. Han et al. (2016) combines the absolute value based pruning strategy from Han et al. (2015) with quantizing and Huffman encoding techniques. See et al. (2016) and Narang et al. (2017) extend Han et al. (2015) for neural machine translation and recurrent neural networks respectively. Yet, all three follow-up works keep the same absolute value based criterion as the one originally proposed by Han et al. (2015).

# 5 Summary of papers

## 5.1 Paper I

**Ko, V. & Hjort, N. L. (2019). Model robust inference with two-stage maximum likelihood estimation for copulas. *Journal of Multivariate Analysis*, 171: 362–381.**

Paper I addresses the impact of the ‘true model assumption’ (i.e. the candidate parametric model is correctly specified and captures the true model that generated data) on the two-stage maximum likelihood (ML) estimator for copula models (often referred to as inference function for margins (IFM)).

The impact of the true model assumption on the ML estimator is studied by White (1982) and Hardin (2003). However, the impact of the true model assumption on the two-stage ML estimator has not received much attention yet. We first identify where  $\hat{\theta}^{2\text{ML}}$ , the two-stage ML estimator of the copula parameter  $\theta$ , converges to. Then, without making the true model assumption, we establish the asymptotic normality of the whole parameter vector of the model  $\eta$ , including the expression for the asymptotic variance.  $\eta$  consists of  $\alpha$ , the parameters of marginal distributions estimated in the stage 1 and  $\theta$ , the copula parameter estimated in stage 2. We also mention that the asymptotic results can also be achieved by using the theories of generalized method of moments (GMM) in econometrics (Hall, 2005) with certain optimal weight matrix. Further, we show that assuming the true model yields some equality relationships of the quantities and that the model robust version of the asymptotic variance formula simplifies to the variance formula defined in Joe (2005). Based on the insight we obtained from this process, we propose some changes to the variance formula from Joe (2005) such that it is more consistent with the current convention of the ML estimator.

To study the numerical behavior of the true model assumption, we perform a set of simulation studies. We find that dropping the true model assumption can both increase and decrease the estimated asymptotic variance. One might find this unintuitive, because it is tempting to reason that dropping the true model assumption should lead to bigger variance. However, it is after all not that strange since dropping the true model assumption changes the interpretation of where  $\hat{\theta}^{2\text{ML}}$  is converging to (least false parameter vs. true parameter). We also test this phenomenon in the univariate case with a set of well-known univariate distributions by estimating their parameters with the ML estimator and looking at how the estimated asymptotic variance changes when the true model assumption is dropped. The simulation shows that dropping the true model assumption (i.e. using the model robust variance formula from White (1982) instead of the canonical one, which is the inverse of the negative Hessian matrix) can both increase and decrease the estimated asymptotic variance.

Further, we observe that the two-stage ML estimator is in most cases still highly efficient when the true model assumption is dropped. Another notable result regarding efficiency is that the

Hájek–Le Cam convolution theorem and related theorems on the optimality of the ML estimation, do not apply when the model is incorrectly specified. The Hájek–Le Cam convolution theorem (Hájek, 1970), combined with the Cramér–Rao lower bound (Cramér, 2016) theory, states that no asymptotically unbiased and normal competing estimator can have a smaller limiting variance matrix than that of the ML estimator. When the models are highly misspecified, it frequently happens that the asymptotic variance of the two-stage ML estimator is smaller than that of the full ML estimator. This occurs both when the model-robust and model non-robust variance formulas are used.

## 5.2 Paper II

**Ko, V. & Hjort, N. L. (2019). Copula information criterion for model selection with two-stage maximum likelihood estimation. *Econometrics and Statistics*, in press.**

Paper II develops CIC, a model robust model selection criterion for two-stage ML estimated copula models.

There has been a lack of proper model selection criterion for copula models that are estimated with the two-stage ML estimator. What is often done as an ad hoc solution in practice, is ‘piecewise model selection’. One first chooses distributions for each marginal distribution, by comparing candidate univariate distributions with a model selection criterion like AIC, separately for each marginal distribution. Once the models for the margins are chosen, one would compare different candidate copulas with AIC, sometimes computed only with the copula part of the log-likelihood. This approach cannot compare the candidate models as a whole and there is a need for a model selection criterion that allows direct, non-piecewise comparison between the copula models.

By using the asymptotic results from Paper I, we derive copula information criterion (CIC). In a nutshell, CIC aims for the model that minimizes the Kullback–Leibler divergence from the real data generating mechanism. CIC does not assume that the chosen parametric model captures this true model, unlike what is assumed for AIC. In this sense, CIC is analogous to the Takeuchi Information Criterion (TIC), which is defined for the maximum likelihood.

Further, if we do make such a true model assumption (i.e. the candidate parametric model is correctly specified and captures the true model that generated data), the CIC simplifies to the  $AIC_{2ML}$ . The  $AIC_{2ML}$  has the same bias correction term as AIC. The only difference between  $AIC_{2ML}$  and AIC is that AIC uses the log-likelihood maximized under the ML estimator, and that  $AIC_{2ML}$  uses the log-likelihood maximized under the two-stage ML estimator. To our knowledge, this is the first time that the validity of the  $AIC_{2ML}$  formula is proven for the two-stage ML estimator.

Table 5.1 shows an overview of the relationship between the model selection criteria discussed in Paper II. When the model (i.e. both copula and margins) is correctly specified, CIC and  $AIC_{2ML}$  become equal, and the same happens between TIC and AIC. Thus, one can compare CIC and  $AIC_{2ML}$  (or TIC and AIC in case of ML estimation) to check whether the model is correctly specified.

### 5.3. Paper III

Further, since both CIC and TIC are estimating the same part of the KL divergence under the same model robust environment, they are compatible. This implies that one can compare CIC to TIC to measure how much one loses in terms of KL divergence by using two-stage ML estimation instead of ML estimation. The same can be done by comparing  $AIC_{2ML}$  to AIC when one believes in the model. However, one should not compare CIC with AIC (or TIC with  $AIC_{2ML}$ ) since they are based on two different model beliefs (i.e. presence of the true model assumption). Figuratively speaking, one would in that situation be comparing comparing apples with pears.

|            |     | Model robust |             |
|------------|-----|--------------|-------------|
|            |     | Yes          | No          |
| Estimation | ML  | TIC          | AIC         |
|            | 2ML | CIC          | $AIC_{2ML}$ |

**Table 5.1:** An overview of the relationship between the model selection criteria discussed in Paper II. ‘Model robust = No’ means that the true model assumption is made.

As a numerical illustration, a set of simulation studies are performed. The first simulation study shows that the better model according to CIC also has in general better prediction performance measured in MSE. In the second simulation study, we show that CIC and  $AIC_{2ML}$  are indeed identical when the model is correctly specified. As the degree of misspecification grows, the bias correction term of CIC penalizes the misspecified model more heavily. Although this behavior of the bias correction part is an important conceptual advance of CIC, this is not sufficient to make CIC outperform  $AIC_{2ML}$  in practice. This is because each misspecified model has the bias correction term and they grow at different speeds, depending on the model. The difference between CIC and  $AIC_{2ML}$  becomes minimal as sample size grows because the log-likelihood part outgrows the bias correction part.

## 5.3 Paper III

**Ko, V., Hjort, N. L. & Hobæk Haff, I. (2019). Focused information criteria for copulas. *Scandinavian Journal of Statistics*, in press.**

Paper III extends the focused information criterion (FIC) from Jullum & Hjort (2017) to copula models.

Most model selection criteria like AIC, TIC, BIC and CIC measure how close a candidate model is to the true model in terms of a certain divergence measure. However, in many cases, the model itself is not the goal, but rather a stepping stone to estimate a specific quantity, such as the mean or the probability of certain event. The main idea behind the FIC, first introduced by Claeskens & Hjort (2003) and later modified by Jullum & Hjort (2017) is that one wants to select a model that is good for a specific task. This specific task is quantified as estimating a focus parameter, say  $T(G)$ , a functional of the data generating distribution. The functional needs to have a clear statistical interpretation across candidate models. Let  $F$  be the distribution function of the candidate model.  $T(\hat{F})$  is then the focus parameter computed with the estimated

candidate model.

When the parametric copula and marginal distributions are used and their parameters are estimated with ML estimator, the result from Jullum & Hjort (2017) holds directly. For the case when the parameters are estimated by two-stage ML estimator, we derive the FIC formula by using the result from Paper I. Further, by using the results from Chen & Fan (2005), we derive the FIC formula for the case when the pseudo-maximum-likelihood (PML) estimator is used to estimate the copula parameter and the marginal distributions are replaced with parametric margins afterwards. In addition, we confirm the validity of the AIC formula for this case.

One of the advantages of FIC is that it can be used to compare models that are estimated under different estimation schemes, including the non-parametric one. FIC can also easily be extended to other estimation schemes as long as the joint asymptotic distribution with the nonparametric focus parameters estimate can be derived. In addition, FIC is a model-robust model selection criterion since it does not assume that any of the candidate model captures the data generating model.

We perform a series of simulations to study the behavior of the FIC. The results show that models with lower FIC values give focus parameter estimates that are closer to the true parameter value, which is the aim of the method. Further, it turns out that FIC also fulfills its role as an estimator of the MSE. Moreover, when the focus parameter is heavily based on a specific part of the distribution, in particular the tails, the models favored by AIC do not necessarily give good estimates of the focus parameter. This seems natural since AIC aims for the model that minimizes overall closeness to the true model in terms of KL-divergence. This demonstrates that there is a clear advantage in using FIC instead of AIC in such situations.

## 5.4 Paper IV

**Ko, V., Oehmcke, S. & Gieseke, F. (2019). Magnitude and Uncertainty Pruning Criterion for Neural Networks. *Submitted for publication.***

Paper IV develops a model pruning criterion for neural networks. Note that since Paper IV is in the field of machine learning, it has somewhat different style of approach (i.e. heuristically driven model pruning criterion that focuses on real-world performance), compared to Paper I – III.

In recent years, deep neural networks have achieved state-of-the-art performance in a broad range of tasks and have gained big popularity in many fields like computer vision and natural language processing (Goodfellow et al., 2016). Typically, these deep neural networks have millions or even billions of parameters, which makes them both computationally and memory intensive. Denil et al. (2013) show that there is significant redundancy in the parameterization of many deep learning models. This overparameterization not only requires unnecessary computational resources, but can also lead to an overfitting of the models.

Ideally, one would like to identify redundant parameters of the model by performing null hypothesis testing on each parameter or by performing model selection with nested models. However, this is at the moment largely infeasible due to the lack of asymptotic theories on the mod-

#### 5.4. Paper IV

ified versions of stochastic gradient descent that are used in practice such as Adam (Kingma & Ba, 2015) and RMSprop (Tieleman & Hinton, 2012). Even in rare cases when the necessary theory is developed, it is often computationally infeasible to estimate the needed quantities due to the large number of parameters (e.g. computing and taking the inverse of Hessian matrix of size 100 million by 100 million).

Naturally, various alternative approaches have been suggested for decreasing the size and the computational footprint of neural networks. One of the canonical works in this field is Han et al. (2015), where they prune the network parameters based on their magnitude. In practice, this means that they simply look into the absolute value of each weight and remove the ones below certain threshold. Although this approach might seem rather brutal, especially from statistical perspective, it works fairly well in practice.

In this paper, we first briefly review how the Wald test statistic is derived and observe that the absolute value based pruning criterion from Han et al. (2015) is actually highly similar to the Wald test statistic. The only difference is that the Wald test statistic compensates for the uncertainty of the parameter by dividing with the standard error. Inspired by this observation, we propose magnitude and uncertainty (M&U) pruning criterion. The M&U pruning criterion can be seen as a generalization of both the absolute value based pruning criterion from Han et al. (2015) and the Wald test statistic. Further, we present a ‘pseudo bootstrap’ scheme, which can efficiently estimate the uncertainty of the weights by using their update information during training.

Our experiments, which are based on various neural network architectures and datasets, show that our M&U pruning criterion outperforms the criterion suggested in Han et al. (2015) most of the time, in terms of test accuracy of the pruned model. Thus, leading to models that require less memory and at smaller computational cost, while minimizing the loss of predictive power.

## 6 Discussion

This PhD thesis consists of four papers. There are possible future works, extensions and room for improvements for each paper. In this section, we focus on four of the relevant discussion topics.

The main contribution of Paper I is the development of model robust asymptotic results of the two-stage maximum likelihood estimator and the study on the impact of the true model assumption. The technical results and insights obtained in Paper I are crucial ground work for Paper II and III. In Paper II, the main contribution is that we derive the copula information criterion (CIC) for the copula models estimated with two-stage ML and that we show how CIC simplifies to  $AIC_{2ML}$  under the true model assumption. The main contribution of Paper III is that we extend the focused information criterion (FIC) framework to three different estimation schemes for copula. In Paper IV, we propose magnitude and uncertainty (M&U) pruning criterion, inspired by the Wald test statistic.

### 6.1 Analytical tool for analyzing the impact of true mode assumption

In Paper I, we derived asymptotic normality for copula models estimated with the two-stage ML estimator, without making the true model assumption. We also showed that making the true model assumption simplifies the expression for the asymptotic variance.

In the simulation part of Paper I, we used various copula models and observed that the presence of the true model assumption (i.e. using the simplified model non-robust variance formula instead of the model robust formula) can both decrease and increase the estimated variance value. At first glance, this might look counter-intuitive since one would expect that the use of the model non-robust variance formula would decrease the variance, since one is ignoring model uncertainty. However, realizing that making the true model assumption actually changes the interpretation of what the estimator is aiming for, this is not strange. In fact, one can observe the exact same phenomenon if one looks at the numerical impact of the true model assumption on the asymptotic variance of the ML estimated parameters for univariate distributions (i.e. using the inverse of observed information instead of so-called sandwich matrix).

In our simulation studies, we could not find any clear pattern on when the true model assumption leads to decrease or increase of the estimated variance. It would be very fruitful to develop an analytical method for analyzing what kind of effect the true model assumption has on the asymptotic variance, given a model. The insight obtained from such a method then can also be generalized into the univariate case for the ML estimator. This will bring a new insight to the classic literatures on the ML estimation of misspecified models including White (1982) and Hardin (2003).

## 6.2 CIC or AIC<sub>2ML</sub>?

In Paper II, we developed CIC, a model robust model selection criterion that is designed for copula models that are estimated with the two-stage ML estimator

$$\text{CIC} = 2\ell_n(\hat{\eta}^{2\text{ML}}) - 2\hat{p}_\eta^*,$$

where  $\ell_n(\hat{\eta}^{2\text{ML}})$  is the log-likelihood of the copula model, maximized with two-stage ML estimator and  $\hat{p}_\eta^*$  is a bias correction term, defined in Paper II.

In the same paper, we showed that making the true model assumption (i.e. the candidate parametric model is correctly specified and captures the true model that generated data) simplifies the CIC to

$$\text{AIC}_{2\text{ML}} = 2\ell_n(\hat{\eta}^{2\text{ML}}) - 2p_\eta,$$

where  $p_\eta$  indicate the length of the parameter vector  $\eta$ .

As we can see, the CIC and AIC<sub>2ML</sub> share the same log-likelihood part, but the difference is that AIC<sub>2ML</sub> has a simplified bias correction term. A sensible reader might ask: Does one should always use CIC instead of AIC<sub>2ML</sub> since it is model robust?

Although CIC has a conceptual advantage above AIC<sub>2ML</sub> that it is model robust, it does not often lead to better performance in practice. There are multiple reasons for this.

The main reason is that the log-likelihood part, the element that is shared by CIC and AIC<sub>2ML</sub> grows linearly with  $n$ , number of data points, while the bias correction terms do not. For reasonable sample sizes, the absolute value of the likelihood part is much bigger than the bias correction part. In practice, this means that if we compare model A and model B, the ranks between these two models are almost always determined by the difference between the log-likelihood terms.

For small sample sizes (depends on the number of parameters to be estimated, but generally  $n < 100$ ), the bias terms have more impact on which model is chosen as the ‘best model’. We can divide the case into two subcategories:

- Case 1. Model A is the true model and model B is misspecified.  
When we look at the bias correction terms, CIC punishes the misspecified model more since  $\hat{p}_\eta^*$  has higher value for the misspecified model. However, usually the log-likelihood of the correct model is already significantly higher than that of the misspecified model. This enables AIC<sub>2ML</sub> to pick the correct model as the winning model almost always. There is virtually no space for CIC to outperform AIC<sub>2ML</sub>.
- Case 2. Both model A and model B are misspecified.  
Since CIC has the  $\hat{p}_\eta^*$  term, which penalizes a misspecified model, one might expect that CIC would outperform AIC<sub>2ML</sub> in this case. However, from the Simulation 2 of Paper II, it turns out that the magnitude of the bias correction term of CIC is model-specific. In practice, this means that CIC tends to pick a specific model as a winning model, which is not necessarily the correct model (see Figure 4 from Paper II).

To sum up, there is in practice very little advantage to use CIC above  $AIC_{2ML}$ . Since the bias correction term of  $AIC_{2ML}$  is much simpler to compute, general advice would be to use  $AIC_{2ML}$ . Note that TIC, which is the ML estimator analogue of CIC, is seldom used in practice because of similar reasons (Burnham & Anderson, 2003).

Yet, CIC still plays an important role for derivation of  $AIC_{2ML}$  and  $\tilde{p}_\eta^*$  can be used as goodness-of-fit test.

### 6.3 Two-stage density estimators via a parametric start

Suppose that  $f_0(y, \theta)$  is a parametric model for a density function, perhaps rough and not a serious approximation to the true density, but a reasonable starting point. One approach would be to ‘expand’ or ‘correct’ such a sub-optimal model with another function  $f_{cr}(y, \gamma)$  via multiplication. The resulting density estimator after such an expansion can be written as

$$f(y, \theta, \gamma) = f_0(y, \theta) f_{cr}(y, \theta, \gamma).$$

A similar idea was explored in Hjort & Glad (1995) where the start model  $f_0(y, \theta)$  is parametric and the multiplicative correction term  $f_{cr}(y)$  is nonparametric. Here, we consider a situation where both the start and the correction parts are parametric. Such an approach would be useful for example in real-life situations where a density function does not fit well in certain regions of  $y$ . One possible candidate for  $f_{cr}(y, \theta, \gamma)$  is the log-linear model

$$f_{cr}(y, \theta, \gamma) = \exp \left[ \sum_{j=1}^m \gamma_j T_j(F_0(y, \theta)) - c_m(\gamma_1, \dots, \gamma_m) \right],$$

where  $F_0$  is the distribution function of  $f_0$  and

$$\begin{aligned} c_m(\gamma) &= c_m(\gamma_1, \dots, \gamma_m) \\ &= \log \left[ \int_{-\infty}^{\infty} f_0(y, \theta) \exp \left[ \sum_{j=1}^m \gamma_j T_j(F_0(y, \theta)) \right] dy \right] \\ &= \log \left[ \int_0^1 \exp \left[ \sum_{j=1}^m \gamma_j T_j(u) \right] du \right] \end{aligned}$$

is the required integration constant, which do not depend on  $f_0$ .  $T_1(u), \dots, T_m(u)$  are a series of basis functions, defined for  $u \in [0, 1]$ . One possible choice is  $T_j(u) = \sqrt{2} \cos(j\pi u)$ . This specific choice of  $T_j(u)$  has nice orthonormality property, that

$$\int_0^1 (T_j(u))^2 du = 1 \quad \text{and} \quad \int_0^1 T_j(u) T_k(u) du = 0 \text{ for } j \neq k.$$

One could estimate  $\theta$  and  $\gamma$  simultaneously by maximizing  $f(y, \theta, \gamma)$  and finding the ML estimator. However, it is in practice often simpler to estimate the start model  $f_0(y, \theta)$  first and then estimate the correction term well, rather than fitting the whole density at once. This leads to a

## 6.4. Utilizing uncertainty information from training

two-stage ML estimator.

*Stage 1:*

Obtain  $\hat{\theta}$  by maximizing  $f_0(y, \theta)$  with respect to  $\theta$ .

*Stage 2:*

Plugging  $\hat{\theta}$  from Stage 1 into  $f_{\text{cr}}(y, \theta, \gamma)$  gives

$$f_{\text{cr}}(y, \hat{\theta}, \gamma) = \exp \left[ \sum_{j=1}^m \gamma_j T_j(F_0(y, \hat{\theta})) - c_m(\gamma_1, \dots, \gamma_m) \right].$$

Obtain  $\hat{\gamma}$  by maximizing  $f_{\text{cr}}(y, \hat{\theta}, \gamma)$  with respect to  $\gamma$ .

By using the technical insight we obtained from Paper I and II, we can derive the large-sample result of such a two-stage estimator. Furthermore, we could also derive a TIC and AIC like model selection criterion for finding the best order  $m$ , for a given choice of  $f_0(y, \theta)$  and the  $T_j$  sequence. Obviously, such a model selection criterion can also be used for comparing different parametric candidates of  $f_0(y, \theta)$  and  $T_j$ .

## 6.4 Utilizing uncertainty information from training

In Paper IV, we proposed the magnitude and uncertainty (M&U) pruning criterion, which is a generalization of both the absolute value based pruning criterion from Han et al. (2015) and the Wald test statistic. To compute the M&U pruning criterion, the uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}$  needs to be estimated for each parameter in the model. Ideally, one would use the standard error as the uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}$ , like in the original Wald test statistic formula. However, this is at the moment infeasible due to lack of necessary asymptotic theories. Even in rare cases when the necessary theory is available, it is often computationally infeasible to estimate the needed quantities due to the large number of parameters (e.g. computing and taking the inverse of Hessian matrix of size above 100 million). As an alternative, we suggested estimating the standard error with the bootstrap procedure (Efron & Tibshirani, 1994). However, using bootstrap of size  $B$  increases the training time by a factor of  $B$ . For moderately sized networks, this can be doable. Yet, advanced and extensive deep network architectures can require multiple days, or even weeks of training. In this case, bootstrap is almost infeasible.

In the same paper, we proposed ‘pseudo bootstrap’ as a heuristic workaround. During the training, we keep track of the parameter values at each iteration. And after the training, we compute the standard deviation by using the saved parameter values from the last  $B$  iterations. This is extremely fast, since we only need to save the parameters and compute the standard deviation in the end. For the detailed motivation and specification of pseudo bootstrap, see Section 4.3 of Paper IV.

Although the experiments show that using the pseudo bootstrap leads to a pruning criterion that outperforms the canonical absolute value based criterion from Han et al. (2015), it is still a question whether the uncertainty observed during the (last iterations of) training is the same as

the type of uncertainty captured by standard error. The standard error of an estimated parameter can be interpreted as how uncertain the estimated parameter value is. Intuitively, the parameters that ‘fluctuated’ a lot during the last iterations of training, would also have higher standard error. However, the fluctuations during the training can be affected by many factors such as which data points were used during the observed iterations, behavior of the estimator, use of regulation tricks like batch normalization (Ioffe & Szegedy, 2015), Dropout (Srivastava et al., 2014) and Dropconnect (Wan et al., 2013).

There is at the moment no analytical or numerical study done on the relationship between the uncertainty during the training and the uncertainty of the final estimate. Identifying the relationship between them would open a possibility for a whole new framework that allows estimating parameter uncertainty (e.g. standard error) from the training information. This will lead to very efficient estimation of standard error for parameters in neural networks, which can be used for pruning, building confidence intervals and more.

# References

- AAS, K. & BERG, D. (2009). Models for construction of multivariate dependence – a comparison study. *The European Journal of Finance* **15**, 639–659.
- AAS, K., CZADO, C., FRIGESSI, A. & BAKKEN, H. (2009). Pair-copula constructions of multiple dependence. *Insurance: Mathematics and Economics* **44**, 182–195.
- ABADI, M., BARHAM, P., CHEN, J., CHEN, Z., DAVIS, A., DEAN, J., DEVIN, M., GHEMAWAT, S., IRVING, G., ISARD, M., KUDLUR, M., LEVENBERG, J., MONGA, R., MOORE, S., MURRAY, D. G., STEINER, B., TUCKER, P., VASUDEVAN, V., WARDEN, P., WICKE, M., YU, Y. & ZHENG, X. (2016). Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*.
- ACAR, E. F., CRAIU, R. V. & YAO, F. (2013). Statistical testing of covariate effects in conditional copula models. *Electronic Journal of Statistics* **7**, 2822–2850.
- AKAIKE, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control* **19**, 716–723.
- ANDERSEN, E. W. (2005). Two-stage estimation in copula models used in family studies. *Lifetime Data Analysis* **11**, 333–350.
- BEVACQUA, E., MARAUN, D., HOBÆK HAF, I., WIDMANN, M. & VRAC, M. (2017). Multivariate statistical modelling of compound events via pair-copula constructions: analysis of floods in Ravenna (Italy). *Hydrology and Earth System Sciences* **21**, 2701–2723.
- BURNHAM, K. P. & ANDERSON, D. R. (2003). *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer Science & Business Media.
- CASELLA, G. & BERGER, R. L. (2002). *Statistical Inference*. Duxbury Pacific Grove, CA.
- CHEN, X. & FAN, Y. (2005). Pseudo-likelihood ratio tests for semiparametric multivariate copula model selection. *Canadian Journal of Statistics* **33**, 389–414.
- CHENG, Y., WANG, D., ZHOU, P. & ZHANG, T. (2017). A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*.
- CLAESKENS, G. & HJORT, N. L. (2003). The focused information criterion [with discussion contributions and a rejoinder]. *Journal of the American Statistical Association* **98**, 900–916.

- CLAESKENS, G. & HJORT, N. L. (2008). *Model Selection and Model Averaging*. Cambridge University Press.
- COURBARIAUX, M., BENGIO, Y. & DAVID, J.-P. (2015). Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in neural information processing systems (NeurIPS)*.
- CRAMÉR, H. (2016). *Mathematical Methods of Statistics [re-issue of the 1946 classic]*. Princeton University Press.
- CYBENKO, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems* **2**, 303–314.
- DENIL, M., SHAKIBI, B., DINH, L., DE FREITAS, N. et al. (2013). Predicting parameters in deep learning. In *Advances in neural information processing systems (NeurIPS)*.
- DENTON, E. L., ZAREMBA, W., BRUNA, J., LECUN, Y. & FERGUS, R. (2014). Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in neural information processing systems (NeurIPS)*.
- DING, G. W. (2018). draw\_convnet. [https://github.com/gwding/draw\\_convnet](https://github.com/gwding/draw_convnet).
- DUMOULIN, V. & VISIN, F. (2016). A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*.
- EFRON, B. & TIBSHIRANI, R. J. (1994). *An Introduction to the Bootstrap*. CRC press.
- EMBRECHTS, P. (2009). Copulas: A personal view. *Journal of Risk and Insurance* **76**, 639–650.
- GENEST, C., GHOUDI, K. & RIVEST, L.-P. (1995). A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika* **82**, 543–552.
- GONG, Y., LIU, L., YANG, M. & BOURDEV, L. (2014). Compressing deep convolutional networks using vector quantization. *arXiv preprint arXiv:1412.6115*.
- GOODFELLOW, I., BENGIO, Y. & COURVILLE, A. (2016). *Deep Learning*. MIT press.
- HÁJEK, J. (1970). A characterization of limiting distributions of regular estimates. *Probability Theory and Related Fields* **14**, 323–330.
- HALL, A. R. (2005). *Generalized Method of Moments*. Oxford University Press.
- HAN, S., MAO, H. & DALLY, W. J. (2016). Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *International Conference on Learning Representations (ICLR)*.
- HAN, S., POOL, J., TRAN, J. & DALLY, W. (2015). Learning both weights and connections for efficient neural network. In *Advances in neural information processing systems (NeurIPS)*.

## REFERENCES

- HARDIN, J. W. (2003). The sandwich estimate of variance. In *Maximum likelihood estimation of misspecified models: Twenty years later*. Emerald Group Publishing Limited, pp. 45–73.
- HASSIBI, B., STORK, D. G. & WOLFF, G. J. (1993). Optimal brain surgeon and general network pruning. In *IEEE international conference on neural networks*.
- HJORT, N. L. & GLAD, I. K. (1995). Nonparametric density estimation with a parametric start. *The Annals of Statistics* , 882–904.
- HOBÆK HAFF, I. (2013). Parameter estimation for pair-copula constructions. *Bernoulli* **19**, 462–491.
- HOBÆK HAFF, I., AAS, K., FRIGESSI, A. & LACAL, V. (2016). Structure learning in Bayesian networks using regular vines. *Computational Statistics & Data Analysis* **101**, 186–208.
- HOBÆK HAFF, I., FRIGESSI, A. & MARAUN, D. (2015). How well do regional climate models simulate the spatial dependence of precipitation? an application of pair-copula constructions. *Journal of Geophysical Research: Atmospheres* **120**, 2624–2646.
- HORNIK, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural networks* **4**, 251–257.
- HUBARA, I., COURBARIAUX, M., SOUDRY, D., EL-YANIV, R. & BENGIO, Y. (2017). Quantized neural networks: Training neural networks with low precision weights and activations. *The Journal of Machine Learning Research (JMLR)* **18**, 6869–6898.
- HWANG, K. & SUNG, W. (2014). Fixed-point feedforward deep neural network design using weights+ 1, 0, and- 1. In *2014 IEEE Workshop on Signal Processing Systems (SiPS)*. IEEE.
- IOFFE, S. & SZEGEDY, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (PMLR)*.
- JADERBERG, M., VEDALDI, A. & ZISSERMAN, A. (2014). Speeding up convolutional neural networks with low rank expansions. *arXiv preprint arXiv:1405.3866* .
- JOE, H. (1997). *Multivariate Models and Multivariate Dependence Concepts*. CRC Press.
- JOE, H. (2005). Asymptotic efficiency of the two-stage estimation method for copula-based models. *Journal of Multivariate Analysis* **94**, 401–419.
- JOE, H. (2014). *Dependence Modeling with Copulas*. CRC Press.
- JULLUM, M. & HJORT, N. L. (2017). Parametric or nonparametric: the FIC approach. *Statistica Sinica* **27**, 951–981.
- JULLUM, M. & HJORT, N. L. (2018). What price semiparametric Cox regression? *Lifetime data analysis* , 1–33.
- KIM, G., SILVAPULLE, M. J. & SILVAPULLE, P. (2007). Comparison of semiparametric and parametric methods for estimating copulas. *Computational Statistics & Data Analysis* **51**, 2836–2850.

- KINGMA, D. P. & BA, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.
- KRIZHEVSKY, A., SUTSKEVER, I. & HINTON, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*.
- KULLBACK, S. & LEIBLER, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics* **22**, 79–86.
- LE CAM, L. (1990). Maximum likelihood: an introduction. *International Statistical Review* **58**, 153–171.
- LEBEDEV, V., GANIN, Y., RAKHUBA, M., OSELEDITS, I. & LEMPITSKY, V. (2014). Speeding-up convolutional neural networks using fine-tuned cp-decomposition. *arXiv preprint arXiv:1412.6553*.
- LECUN, Y., BOTTOU, L., BENGIO, Y., HAFFNER, P. et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86**, 2278–2324.
- LIN, Z., COURBARIAUX, M., MEMISEVIC, R. & BENGIO, Y. (2016). Neural networks with few multiplications. In *International Conference on Learning Representations (ICLR)*.
- NARANG, S., ELSER, E., DIAMOS, G. & SENGUPTA, S. (2017). Exploring sparsity in recurrent neural networks. In *International Conference on Learning Representations (ICLR)*.
- NELSEN, R. B. (2006). *An Introduction to Copulas*. Springer Science & Business Media.
- PALARO, H. & HOTTA, L. (2006). Using conditional copula to estimate value at risk. *Journal of Data Science* **4**, 93–115.
- PASZKE, A., GROSS, S., CHINTALA, S., CHANAN, G., YANG, E., DEVITO, Z., LIN, Z., DESMAISON, A., ANTIGA, L. & LERER, A. (2017). Automatic differentiation in pytorch.
- PATTON, A. J. (2002). *Applications of Copula Theory in Financial Econometrics*. Ph.D. thesis, University of California, San Diego.
- PATTON, A. J. (2006). Modelling asymmetric exchange rate dependence. *International Economic Review* **47**, 527–556.
- RASTEGARI, M., ORDONEZ, V., REDMON, J. & FARHADI, A. (2016). Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision (ECCV)*. Springer.
- SCHWARZ, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics* **6**, 461–464.
- SEE, A., LUONG, M.-T. & MANNING, C. D. (2016). Compression of neural machine translation models via pruning. In *The SIGNLL Conference on Computational Natural Language Learning (CoNLL)*.

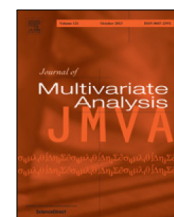
## REFERENCES

- SHAO, J. (2003). *Mathematical Statistics*. Springer Texts in Statistics. Springer.
- SHIH, J. H. & LOUIS, T. A. (1995). Inferences on the association parameter in copula models for bivariate survival data. *Biometrics* **51**, 1384–1399.
- SIMONYAN, K. & ZISSERMAN, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* .
- SINGHAL, H. (2017). Convolutional neural network with tensorflow implementation. <https://medium.com/data-science-group-iitr/building-a-convolutional-neural-network-in-python-with-tensorflow-d251c>  
Accessed: 2019-04-22.
- SKLAR, A. (1959). Fonctions de répartition à n dimensions et leurs marges. *Publ. Inst. Statist. Univ. Paris* **8**, 229–231.
- SPIEGELHALTER, D. J., BEST, N. G., CARLIN, B. P. & VAN DER LINDE, A. (2002). Bayesian measures of model complexity and fit. *Journal of the royal statistical society: Series b (statistical methodology)* **64**, 583–639.
- SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I. & SALAKHUTDINOV, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research (JMLR)* **15**, 1929–1958.
- TAKEUCHI, K. (1976). The distribution of information statistics and the criterion of goodness of fit of models. *Mathematical Science* **153**, 12–18.
- TIELEMAN, T. & HINTON, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning* **4**, 26–31.
- TOULIS, P., AIROLDI, E. M. et al. (2017). Asymptotic and finite-sample properties of estimators based on stochastic gradients. *The Annals of Statistics* **45**, 1694–1727.
- TSUKAHARA, H. (2005). Semiparametric estimation in copula models. *Canadian Journal of Statistics* **33**, 357–375.
- VANHOUCKE, V., SENIOR, A. & MAO, M. Z. (2011). Improving the speed of neural networks on cpus .
- VELIČKOVIĆ, P. (2016). Tikz. <https://github.com/PetarV-/TikZ>.
- WAN, L., ZEILER, M., ZHANG, S., LE CUN, Y. & FERGUS, R. (2013). Regularization of neural networks using dropconnect. In *International Conference on Machine Learning (ICML)*.
- WHITE, H. (1982). Maximum likelihood estimation of misspecified models. *Econometrica: Journal of the Econometric Society* , 1–25.

- WIBROW, M. (2014). Drawing neural network with tikz. <https://tex.stackexchange.com/questions/153957/drawing-neural-network-with-tikz>. Accessed: 2019-04-22.
- XU, J. J. (1996). *Statistical modelling and inference for multivariate and longitudinal discrete response data*. Ph.D. thesis, University of British Columbia.







# Model robust inference with two-stage maximum likelihood estimation for copulas

Vinnie Ko<sup>\*</sup>, Nils Lid Hjort

Department of Mathematics, University of Oslo, PB 1053, Blindern, NO-0316, Oslo, Norway



## ARTICLE INFO

### Article history:

Received 9 May 2018

Available online 21 January 2019

### AMS 2010 subject classifications:

primary 62H12

secondary 62F12

### Keywords:

Copula

Inference functions for margins

Large-sample inference

Model misspecification

Model robustness

Two-stage maximum likelihood

## ABSTRACT

This article is concerned with inference in parametric copula setups, where both the marginals and the copula have parametric forms. For such models, two-stage maximum likelihood estimation, often referred to as inference function for margins, is used as an attractive alternative to the full maximum likelihood estimation strategy. Previous studies of the two-stage maximum likelihood estimator have largely been based on the assumption that the chosen parametric model captures the true model that generated data. We study the impact of dropping this true model assumption, both theoretically and numerically. We first show that the two-stage maximum likelihood estimator is consistent for a well-defined least false parameter value, different from the analogous least false parameter associated with the full maximum likelihood procedure. Then we demonstrate limiting normality of the full vector of estimators, with concise matrix notation for the variance matrices involved. Along with consistent estimators for these, we have built a model-robust machinery for inference in parametric copula models. The special case where the parametric model is assumed to hold corresponds to situations studied earlier in the literature, with simpler formulas for variance matrices. As a numerical illustration, we perform a set of simulations. We also analyze five-dimensional Norwegian precipitation data. We find that the variance of the copula parameter estimate can both increase and decrease, by dropping the true model assumption. In addition, we observe that the two-stage maximum likelihood estimator is still highly efficient when the true model assumption is dropped and thus the model robust asymptotic variance formulas are used. Additionally, we discover that using highly misspecified models can lead to situations where the asymptotic variance of the two-stage maximum likelihood estimator is lower than that of full maximum likelihood estimator. Our results are also used to analyze the mean squared error properties for both the full and the two-stage maximum likelihood estimators of any focus parameter.

© 2019 Elsevier Inc. All rights reserved.

## 1. Introduction and copula models

The popularity of copula modeling and methods has increased rapidly in the last decade and now they are regularly used in fields like biostatistics, hydrology, finance and actuarial science; see Embrechts [10]. One of the more popular estimation techniques for parametric copulas with parametric margins is the two-stage maximum likelihood estimation (two-stage ML estimation), first introduced by Shih and Louis [35]. This two-stage method is also often referred to as inference function

<sup>\*</sup> Corresponding author.

E-mail address: [vinniebk@math.uio.no](mailto:vinniebk@math.uio.no) (V. Ko).

for margins (IFM), a term coined by Xu [39]. The asymptotic behavior and efficiency of this method have been studied both theoretically and numerically in [5,19,20,24,35,39].

Although the insights and results from these studies are fruitful, they are generally based on the assumption that the chosen copula and marginal distributions are the true model that generated data. We refer to this as the ‘true model assumption’, and the limitations and impact of this assumption for two-stage ML estimation in copula contexts have not been studied earlier, to our knowledge. Andersen [4] has worked with model robust versions of asymptotic variances. This was in the context of composite likelihoods, however, and her work does not give attention to the inference and impact of model robustness. In this paper we develop model robust inference methods based on two-stage ML estimators, and study their theoretical and numerical properties under possible model misspecification.

Our technical setting is as follows. Let  $(Y_1, \dots, Y_d)^\top$  be a  $d$ -variate continuous random vector originating from a joint density  $g(y_1, \dots, y_d)$ , and further let  $y_1 = (y_{1,1}, \dots, y_{1,d})^\top, \dots, y_n = (y_{n,1}, \dots, y_{n,d})^\top$  be mutually independent observations of this variable. Typically, this true joint distribution  $g$  is unknown. Let  $f(y_1, \dots, y_d, \eta)$  be our choice of parametric approximation of  $g$ , with  $\eta$  the parameter vector, belonging to some connected subset of the appropriate Euclidean space. Further,  $G$  and  $F(\cdot, \eta)$  indicate cumulative distribution functions corresponding to  $g$  and  $f(\cdot, \eta)$ , respectively. In addition,  $G_j(y_j)$  and  $F_j(y_j, \alpha_j)$  indicate the  $j$ th marginal distribution functions corresponding to  $G$  and  $F(\cdot, \eta)$ , respectively, with  $\alpha_j$  the parameter vector pertaining to modeling margin component  $j \in \{1, \dots, d\}$ .

Starting from the joint distribution  $F(\cdot, \eta)$  under certain regularity conditions, Sklar's theorem [36] implies that there exists a copula  $C(u_1, \dots, u_d, \theta)$  that satisfies

$$F(y_1, \dots, y_d, \eta) = C\{F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta\}, \quad (1)$$

with  $\theta$  the vector of parameters for the copula. For the full parameter vector, now conveniently blocked into parameters for margins and the part for the copula, we use

$$\eta = (\alpha^\top, \theta^\top)^\top = (\alpha_1^\top, \dots, \alpha_d^\top, \theta^\top)^\top.$$

When  $F(y_1, \dots, y_d, \eta)$  is continuous,  $C(\cdot, \theta)$  is unique. If we assume that  $F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d)$  are absolutely continuous and strictly increasing,  $C(\cdot, \theta)$  can be differentiated and (1) becomes

$$f(y_1, \dots, y_d, \eta) = c\{F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta\} \prod_{j=1}^d f_j(y_j, \alpha_j), \quad (2)$$

where  $c(u_1, \dots, u_d) = \partial^d C(u_1, \dots, u_d, \theta) / \partial u_1 \cdots \partial u_d$  and  $f_j(y_j, \alpha_j) = \partial F_j(y_j, \alpha_j) / \partial y_j$ ; see Nelsen [32]. Analogously, we can decompose the true density  $g$  into marginal densities and copula density and obtain

$$g(y_1, \dots, y_d) = c_0\{G_1(y_1), \dots, G_d(y_d)\} \prod_{j=1}^d g_j(y_j),$$

with  $c_0$  the true copula.

One of the most important families of copulas is the Archimedean class. An Archimedean copula has the form

$$C(u_1, \dots, u_d) = \phi^{-1}\{\phi(u_1) + \dots + \phi(u_d)\},$$

where  $\phi$  is its generator; see, e.g., [31]. One of the most common Archimedean copulas is the Gumbel copula with parameter  $\theta \in [1, \infty)$ , which is defined for all  $u_1, \dots, u_d \in (0, 1)$ , as

$$C(u_1, \dots, u_d) = \exp[-(|\ln u_1|^\theta + \dots + |\ln u_d|^\theta)^{1/\theta}].$$

Another member of the Archimedean family is the Frank copula with parameter  $\theta \in (0, \infty)$ , defined for all  $u_1, \dots, u_d \in (0, 1)$ , defined as

$$C(u_1, \dots, u_d) = -\frac{1}{\theta} \ln \left[ 1 + \frac{\prod_{i=1}^d \{\exp(-\theta u_i) - 1\}}{\{\exp(-\theta) - 1\}^{d-1}} \right].$$

For further parametric copula constructions, see, e.g., [11–13,19,21,32], and with several multiparameter copula models exhibited and discussed in [9,33].

The further structure of this paper is as follows. In Section 2, we briefly describe how the two-stage ML estimation method works. In Section 3, we first derive the limit distribution for two-stage ML estimators outside model conditions, including proof of consistency towards the relevant least false parameter. After this, we examine consequences of the true model assumption for the asymptotic distribution. The resulting model non-robust asymptotic variance formula is essentially similar to that given in Joe [20], but one difference is that we choose to use quantities that are more in line with the classical ML estimation theory. Our results give rise to clear recipes for confidence intervals, confidence curves, and hypothesis tests, based on two-stage ML estimators, discussed in Section 4.

In Sections 5 and 6, we study the numerical behavior of our model robust inference methods against that of the model non-robust version by using a set of simulations and a dataset of Norwegian precipitation. Generally speaking, the two-stage

ML estimator  $\tilde{\eta}$  has a variance matrix, say  $\Sigma$ , for which our apparatus provides two estimators:  $\tilde{\Sigma}_A$ , computed under model conditions, and  $\tilde{\Sigma}_B$  using the model-robust machinery. When the true model really is captured by the candidate model, then both variance matrix estimators tend to be in agreement. Next, when the data generating mechanism lies outside the model, the second method aims at the correct matrix, whereas the first is not consistent. The disagreement between these two variance matrix estimators gets larger as the degree of misspecification increases. Furthermore, we observe that a high degree of model misspecification may lead to situations where the limiting variance matrix of the two-stage ML estimator is smaller than the corresponding variance matrix of the full ML estimator. This cannot happen when the data generating mechanism is inside the parametric model. By classical theorems on the optimality of ML estimators, such as the Hájek–Le Cam convolution theorem [15], each competing sequence of estimators will have a limiting variance matrix at least as large as the one for ML estimation (the matrix difference is nonnegative definite), under model conditions and a few further regularity assumptions on competitors. But these optimality theorems for ML estimation do not apply when the data generating mechanism is outside the parametric model.

In Section 7, our methods and results are used to examine the mean squared error properties of both full and two-stage maximum likelihood estimators for any given focus parameter. This also leads to a new focused information criterion for copulas, when examining different candidate models. In Section 8, we offer a list of concluding remarks, some pointing to further research work. In particular, we mention model selection criteria, and explain briefly that our two-stage ML machinery may be extended to classes of conditional copula regression models.

## 2. Two-stage maximum likelihood

### 2.1. Maximum likelihood and Kullback–Leibler divergence

With observations from a model parametrized via a parameter vector  $\eta$ , the ML estimator  $\hat{\eta}$  is the maximizer of  $\ell(\eta)$ , the log-likelihood function of the model for the given observations. Properties of ML estimators are extensively covered by the classic literature in statistics, including [6,29,38]. In this paper, we use  $\hat{\cdot}$  to indicate that a quantity is estimated by full or joint ML, and  $\tilde{\cdot}$  to indicate that the quantity in question is estimated by two-stage ML, covered in Section 2.2.

The Kullback–Leibler (KL) divergence [28] measures the extent to which one probability distribution diverges from another. The KL divergence from  $g$  to  $f$  is defined as

$$\text{KL}\{g, f(\cdot, \eta)\} = \int g(y) \ln\{g(y)/f(y, \eta)\} dy.$$

In our technical setting, the true density  $g$  is the same for all models. Thus, minimizing the KL divergence is equivalent to maximizing  $\int g(y) \ln f(y, \eta) dy$ .

Under the usual regularity assumptions and assuming that the integral is finite, the Law of Large Numbers gives

$$\frac{1}{n} \ell(\eta) = \frac{1}{n} \sum_{i=1}^n \ln f(y_i, \eta) \xrightarrow{p} \int g(y) \ln f(y, \eta) dy = E_G\{\ln f(Y, \eta)\}.$$

Here and later it is understood that the convergence relates to the sample size  $n$  growing beyond bounds. The above result implies under mild and standard regularity conditions that  $\hat{\eta}$ , the ML estimator of  $\eta$ , will tend asymptotically to  $\eta_{0,\text{ML}}$ , the least false parameter value and the maximizer of  $\int g(y) \ln f(y, \eta) dy$ . Thus,

$$\hat{\eta} \xrightarrow{p} \eta_{0,\text{ML}} = \arg \min_{\eta} \text{KL}\{g, f(\cdot, \eta)\} = \arg \max_{\eta} \int g(y) \ln f(y, \eta) dy.$$

In other words, the  $\eta_{0,\text{ML}}$  is the parameter vector making the candidate model closest to the true model, in the sense of KL. When the parametric model is correctly specified, i.e.,  $g(y) = f(y, \eta_{0,\text{ML}})$ , for a suitable  $\eta_{0,\text{ML}}$ , the minimum of the KL divergence is zero and  $\eta_{0,\text{ML}}$  is called true parameter value. By applying the decomposition (2) on our setting and  $f(y, \eta)$ , we have that  $\hat{\eta}$  is a consistent estimator of the least false parameter  $\eta_{0,\text{ML}}$ , say  $\eta_{0,\text{ML}} = (\alpha_{0,\text{ML}}^\top, \theta_{0,\text{ML}}^\top)^\top$ .

### 2.2. Two-stage maximum likelihood estimator

When the dimension  $d$  of the copula gets higher, the ML estimator becomes computationally more demanding and not even feasible when the number of parameters is high [19]. To avoid this problem, the two-stage ML is a natural alternative estimation strategy, first proposed, for the two-dimensional case, by Shih and Louis [35].

When parametric families for the copula and the margins are chosen, the two-stage ML estimator works as follows.

*Stage 1:* For each  $j \in \{1, \dots, d\}$ , obtain  $\tilde{\alpha}_j$ , the marginal ML estimate of  $\alpha_j$ , by maximizing, with respect to  $\alpha_j$ ,

$$\ell_{f_j} = \sum_{i=1}^n \ln f_j(y_{i,j}, \alpha_j).$$

Stage 2: Plug in  $\tilde{\alpha}_1, \dots, \tilde{\alpha}_d$  from Stage 1, to get

$$\ell(\tilde{\alpha}, \theta) = \sum_{i=1}^n [\ln f_1(y_{i,1}, \tilde{\alpha}_1) + \dots + \ln f_d(y_{i,d}, \tilde{\alpha}_d) + \ln c\{F_1(y_{i,1}, \tilde{\alpha}_1), \dots, F_d(y_{i,d}, \tilde{\alpha}_d), \theta\}].$$

Then,  $\theta$  is estimated by maximizing  $\ell(\tilde{\alpha}, \theta)$  with respect to  $\theta$ , yielding

$$\tilde{\theta} = \arg \max_{\theta} \ell(\tilde{\alpha}, \theta) = \arg \max_{\theta} \ell_c(\tilde{\alpha}, \theta),$$

where

$$\ell_c(\alpha, \theta) = \sum_{i=1}^n \ln c\{F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta\}.$$

The two-stage ML estimate  $\tilde{\eta} = (\tilde{\alpha}^\top, \tilde{\theta}^\top)^\top = (\tilde{\alpha}_1^\top, \dots, \tilde{\alpha}_d^\top, \tilde{\theta}^\top)^\top$  obtained in this way satisfies

$$\left( \frac{\partial}{\partial \tilde{\alpha}_d} \ell_{f_1}, \dots, \frac{\partial}{\partial \tilde{\alpha}_d} \ell_{f_d}, \frac{\partial}{\partial \tilde{\theta}} \ell_c \right) = (0, \dots, 0).$$

There are  $p_1 + \dots + p_d + q$  parameters and equations here, where  $p_j$  is the dimension of  $\alpha_j$  and  $q$  the dimension of  $\theta$ . In the next section we demonstrate limiting normality for the full  $(p_1 + \dots + p_d + q)$ -dimensional  $\tilde{\eta}$ . Note that since the main focus of this paper is two-stage ML estimation, we will simplify the notation for the least false parameter of two-stage ML by using  $\eta_0 = \eta_{0,2ML}$  throughout this paper.

### 3. Large-sample behavior of two-stage maximum likelihood estimators

To work out clear limit theorems for the two-stage ML estimators we are helped by the following regularity assumptions.

A1:  $\eta \in \Theta$ , where  $\Theta$  is compact.

A2:  $\ln f_j(y, \alpha_j)$  is twice differentiable with respect to  $\alpha_j$ .

A3:  $\ln c\{F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta\}$  is thrice differentiable with respect to  $\alpha$  and  $\theta$ .

A4: There exists a function  $R(y_j)$  such that  $E_G\{R(Y_j)\} < \infty$  and  $|\ln f_j(y_j, \alpha_j)| \leq R(y_j)$  for all  $y_j, \alpha_j$ .

A5: There exists a function  $R(y, \alpha)$  such that  $E_G\{R(Y, \alpha)\} < \infty$  and  $|\ln c(y, \alpha, \theta)| \leq R(y, \alpha)$  for all  $y, \alpha, \theta$ .

In Stage 1 of two-stage ML estimation, the parameters of each margin are estimated by using separate ML estimation. We may consequently use large-sample results from ML estimation theory directly, regarding the behavior of  $\tilde{\alpha}_j$ . In particular, the estimator  $\tilde{\alpha}_j$  is now aiming for the least false value instead of the usual true parameter value under the true model assumption.

#### 3.1. Large-sample results for Stage 1 of two-stage ML estimation

The model-robust asymptotic proprieties of the ML estimator are already covered by White [38]. Here, we briefly recap the result.

**Lemma 1.** Let  $\tilde{\alpha}$  be the ML estimator of the margin parameter vector  $\alpha = (\alpha_1, \dots, \alpha_d)^\top$  from Stage 1 and let  $\alpha_0 = (\alpha_{0,1}, \dots, \alpha_{0,d})^\top$  be the least false value. Then we have

$$\sqrt{n}(\tilde{\alpha} - \alpha_0) = \sqrt{n} \mathcal{I}_\alpha^{-1} U_{n,\alpha}(\alpha_0) + o_p(1) \rightsquigarrow \mathcal{I}_\alpha^{-1} A_\alpha \sim \mathcal{N}(0, \mathcal{I}_\alpha^{-1} K_\alpha \mathcal{I}_\alpha^{-1}),$$

where

$$U_{n,\alpha}(\alpha) = \begin{pmatrix} U_{n,\alpha_1}(\alpha_1) \\ \vdots \\ U_{n,\alpha_d}(\alpha_d) \end{pmatrix} = \begin{pmatrix} \frac{1}{n} \sum_{i=1}^n U_{\alpha_1}(y_{i,1}, \alpha_1) \\ \vdots \\ \frac{1}{n} \sum_{i=1}^n U_{\alpha_d}(y_{i,d}, \alpha_d) \end{pmatrix} = \begin{pmatrix} \frac{1}{n} \sum_{i=1}^n \frac{\partial}{\partial \alpha_1} \ln f_1(y_{i,1}, \alpha_1) \\ \vdots \\ \frac{1}{n} \sum_{i=1}^n \frac{\partial}{\partial \alpha_d} \ln f_d(y_{i,d}, \alpha_d) \end{pmatrix},$$

$$K_\alpha = \begin{pmatrix} K_{\alpha_1} & K_{\alpha_1, \alpha_2} & \dots & K_{\alpha_1, \alpha_d} \\ K_{\alpha_2, \alpha_1} & K_{\alpha_2} & \dots & K_{\alpha_2, \alpha_d} \\ \vdots & \vdots & \ddots & \vdots \\ K_{\alpha_d, \alpha_1} & K_{\alpha_d, \alpha_2} & \dots & K_{\alpha_d} \end{pmatrix}, \quad K_{\alpha_j} = K_{\alpha_j, \alpha_j},$$

$$K_{\alpha_j, \alpha_k} = \text{cov}_G\{U_{\alpha_j}(y_j, \alpha_{0,j}), U_{\alpha_k}(y_k, \alpha_{0,k})\} = E_G\{U_{\alpha_j}(y_j, \alpha_{0,j}) U_{\alpha_k}(y_k, \alpha_{0,k})^\top\},$$

$$\mathcal{I}_\alpha = \begin{pmatrix} \mathcal{I}_{\alpha_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathcal{I}_{\alpha_d} \end{pmatrix}, \quad \mathcal{I}_{\alpha_j} = - \int g_j \frac{\partial^2}{\partial \alpha_{0,j} \partial \alpha_{0,j}^\top} \ln f_j(y_j, \alpha_{0,j}) dy_j = -E_{G_j}\{H_{\alpha_j}(y_j, \alpha_{0,j})\}.$$

We have used  $H_{\alpha_j}(y_j, \alpha_j)$  for the Hessian matrix operation on  $\ln f_j(y_j, \alpha_j)$ .

For the proof, see the proof of Theorem 3.2 in White [38].

Under the assumption that the margins are correctly specified, one has  $K_{\alpha_j} = \mathcal{I}_{\alpha_j}$ . When  $j \neq k$ , however, it holds in general that  $K_{\alpha_j, \alpha_k} \neq 0$ . The asymptotic variance matrix will hence keep the sandwich form even under the assumption that margins are correctly specified.

### 3.2. Large-sample results for Stage 2 of two-stage ML estimation

Under the assumption that all margins and the copula are correctly specified, it is well known that  $\tilde{\theta}$  is a consistent estimator of the true parameter value  $\theta_0$ , i.e.,  $\text{KL}\{c_0, c(\cdot, \theta_0)\} = 0$ . However, without this assumption, the consistency of  $\tilde{\theta}$  needs more care, also since it needs to be clarified precisely what it is aiming for.

**Lemma 2.** Consider the function

$$M(\alpha_0, \theta) = \int g \ln c\{F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d}), \theta\} dy$$

and assume that there exists  $\theta_0$ , a unique and well-separated point of maximum of  $M(\alpha_0, \theta)$ , which satisfies

$$\sup\{M(\alpha_0, \theta): d(\theta, \theta_0) < \varepsilon\} < M(\alpha_0, \theta_0)$$

for every  $\varepsilon > 0$ ; here  $d(\theta, \theta_0)$  refers to Euclidean distance. Let  $M_n(\tilde{\alpha}, \theta)$  be the Stage 2 sample version of  $M(\alpha_0, \theta)$ , viz.

$$M_n(\tilde{\alpha}, \theta) = \frac{1}{n} \sum_{i=1}^n \ln c\{F_1(y_{i,1}, \tilde{\alpha}_1), \dots, F_d(y_{i,d}, \tilde{\alpha}_d), \theta\},$$

where the  $\tilde{\alpha}_j$ s are the estimates from the Stage 1. Then  $\tilde{\theta}$ , the maximizer of  $M_n(\tilde{\alpha}, \theta)$ , is a consistent estimator of  $\theta_0$ , the least false parameter value.

The proof is given in the Online Supplement [26].

So, under margin and copula misspecification,  $\tilde{\theta}$  is still consistent, but for the appropriate least false parameter value, rather than for any ‘true’ parameter value. The divergence in question is of the KL type form

$$\int g \ln \frac{c_0\{F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d})\}}{c\{F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d}), \theta\}} dy.$$

This leads to a precise notion of the least false parameter vector  $\eta_0 = (\alpha_0^\top, \theta_0^\top)^\top$  associated with two-stage ML estimation.

Based on the consistency lemma above, we now derive the model robust asymptotic distribution of the two-stage ML estimator.

**Proposition 1.** With  $\tilde{\eta}$  the two-stage ML estimator of  $\eta$ , we have

$$\sqrt{n}(\tilde{\eta} - \eta_0) = \sqrt{n} \begin{pmatrix} \mathcal{I}_\alpha & 0 \\ \mathcal{I}_{\alpha,\theta}^\top & \mathcal{I}_\theta \end{pmatrix}^{-1} \begin{pmatrix} U_{n,\alpha}(\alpha_0) \\ U_{n,\theta}(\alpha_0, \theta_0) \end{pmatrix} + \begin{pmatrix} o_p(1) \\ o_p(1) \end{pmatrix} \rightsquigarrow \mathcal{I}_\eta^{-1} \Lambda_\eta \sim \mathcal{N}(0, V_\eta),$$

where

$$V_\eta = \mathcal{I}_\eta^{-1} K_\eta (\mathcal{I}_\eta^{-1})^\top,$$

$$U_{n,\theta}(\alpha, \theta) = \frac{1}{n} \sum_{i=1}^n U_\theta(y_i, \alpha, \theta) = \frac{1}{n} \sum_{i=1}^n \frac{\partial}{\partial \theta} \ln c\{F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta\},$$

$$K_\eta = \begin{pmatrix} K_\alpha & K_{\alpha,\theta} \\ K_{\alpha,\theta}^\top & K_\theta \end{pmatrix},$$

$$K_\theta = \text{var}_G U_\theta(y, \alpha_0, \theta_0) = E\{U_\theta(y, \alpha_0, \theta_0) U_\theta(y, \alpha_0, \theta_0)^\top\},$$

$$K_{\alpha,\theta} = \text{cov}_G\{U_\alpha(y, \alpha_0), U_\theta(y, \alpha_0, \theta_0)\} = E\{U_\alpha(y, \alpha_0) U_\theta(y, \alpha_0, \theta_0)^\top\},$$

$$\mathcal{I}_\eta = \begin{pmatrix} \mathcal{I}_\alpha & 0 \\ \mathcal{I}_{\alpha,\theta}^\top & \mathcal{I}_\theta \end{pmatrix},$$

$$\mathcal{I}_\theta = - \int g \frac{\partial^2}{\partial \theta_0 \partial \theta_0^\top} \ln c\{F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d}), \theta\} dy = -E_G\{H_\theta(y, \alpha_0, \theta_0)\},$$

$$\mathcal{I}_{\alpha,\theta} = - \int g \frac{\partial^2}{\partial \alpha_0 \partial \theta_0^\top} \ln c\{F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d}), \theta\} dy = -E_G\{H_{\alpha,\theta}(y, \alpha_0, \theta_0)\}.$$

The proof is given in the Online Supplement [26].

Compared to the results from [20,39] where margins and copula are assumed to be correctly specified, the asymptotic variance expression in Proposition 1 has a more general form. Writing it in block matrix form gives

$$V_\eta = \begin{pmatrix} V_\alpha & V_{\alpha,\theta} \\ V_{\alpha,\theta}^\top & V_\theta \end{pmatrix} = \begin{pmatrix} \mathcal{I}_\alpha^{-1} & 0 \\ -\mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} & \mathcal{I}_\theta^{-1} \end{pmatrix} \begin{pmatrix} K_\alpha & K_{\alpha,\theta} \\ K_\alpha^\top & K_\theta \end{pmatrix} \begin{pmatrix} \mathcal{I}_\alpha^{-1} & -\mathcal{I}_\alpha^{-1} \mathcal{I}_{\alpha,\theta} \mathcal{I}_\theta^{-1} \\ 0 & \mathcal{I}_\theta^{-1} \end{pmatrix}, \quad (3)$$

where

$$\begin{aligned} V_\alpha &= \mathcal{I}_\alpha^{-1} K_\alpha \mathcal{I}_\alpha^{-1}, \\ V_\theta &= \mathcal{I}_\theta^{-1} K_\theta \mathcal{I}_\theta^{-1} + \mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} K_\alpha \mathcal{I}_\alpha^{-1} \mathcal{I}_{\alpha,\theta} \mathcal{I}_\theta^{-1} - \mathcal{I}_\theta^{-1} K_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} \mathcal{I}_{\alpha,\theta} \mathcal{I}_\theta^{-1} - \mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} K_{\alpha,\theta} \mathcal{I}_\theta^{-1}, \\ V_{\alpha,\theta} &= \mathcal{I}_\alpha^{-1} K_{\alpha,\theta} \mathcal{I}_\theta^{-1} - \mathcal{I}_\alpha^{-1} K_\alpha \mathcal{I}_\alpha^{-1} \mathcal{I}_{\alpha,\theta} \mathcal{I}_\theta^{-1}. \end{aligned}$$

Andersen [4] has a similar result, but in the context of a composite likelihood construction.

Further, since the two-stage ML estimator can be seen as a special case of the generalized method of moments (GMM) in econometrics, Proposition 1 and Lemma 2 can also be seen as the direct result of consistency and asymptotic normality of the GMM estimator with the optimal weight matrix  $W = K_\eta^{-1}$ . See Hall [16] for details of the GMM estimator. Similarly, Proposition 1 can also be derived by using the theories of estimating equations, like in Section 5.5 of Joe [21].

Since the main focus of our paper is finding the exact effect of the true model assumption, i.e.,  $f = g$ , we now look at how the true model assumption simplifies the result from Proposition 1. As already mentioned in Section 3.1, the true model assumption in Stage 1 gives equality  $K_{\alpha_j} = \mathcal{I}_{\alpha_j}$  for all  $j \in \{1, \dots, d\}$ , where  $d$  indicates the dimension of the data.

Joe [20] implements this result by replacing each  $\mathcal{I}_{\alpha_j}$  with  $K_{\alpha_j}$  in his asymptotic variance formula. Although this method is theoretically correct, it is not the most economical way of stating results, since  $V_\eta$  contains more  $\mathcal{I}_{\alpha_j}$ s than  $K_{\alpha_j}$ s; note that every  $K_{\alpha_j}$  is ‘sandwiched’ by  $\mathcal{I}_{\alpha_j}$  in (3). In addition, this is in opposition to the usual practice of maximum likelihood theory, where the asymptotic variance is defined and used as the inverse of Fisher information, as opposed to the variance of the score function vector.

Thus, under the true model assumption, we choose to simplify  $V_\eta$  by replacing  $K_\alpha$  with

$$K_\alpha^{\text{TMA}} = \begin{pmatrix} \mathcal{I}_{\alpha_1} & K_{\alpha_1,\alpha_2} & \dots & K_{\alpha_1,\alpha_d} \\ K_{\alpha_2,\alpha_1} & \mathcal{I}_{\alpha_2} & \dots & K_{\alpha_2,\alpha_d} \\ \vdots & \vdots & \ddots & \vdots \\ K_{\alpha_d,\alpha_1} & K_{\alpha_d,\alpha_2} & \dots & \mathcal{I}_{\alpha_d} \end{pmatrix}.$$

Although our method is theoretically the same as that of Joe [20], it will make a difference in practice since it is almost never the case that the two matrices  $K_{\alpha_j}$  and  $\mathcal{I}_{\alpha_j}$ , though identical as population quantities, have empirical estimates that are very close.

**Lemma 3.** Under the assumption that the margins and copula are correctly specified, one has  $K_{\alpha,\theta} = 0$ .

The proof is given in the Online Supplement [26].

In two-stage ML estimation, the log-likelihood functions are different in Stages 1 and 2. This implies that the true model assumption does not give  $\mathcal{I}_{\alpha,\theta} = K_{\alpha,\theta}$ . Instead, the true model assumption for two-stage ML estimation yields a different relationship, as follows.

**Lemma 4.** Let

$$U_\alpha^*(y, \alpha) = \begin{pmatrix} U_{\alpha_1}^*(y, \alpha_1) \\ \vdots \\ U_{\alpha_d}^*(y, \alpha_d) \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial \alpha_1} \ln c\{F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta\} \\ \vdots \\ \frac{\partial}{\partial \alpha_d} \ln c\{F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta\} \end{pmatrix}$$

and

$$K_{\alpha,\theta}^* = \text{cov}_G\{U_\alpha^*(y, \alpha_0), U_\theta(y, \alpha_0, \theta_0)\} = E_G\{U_\alpha^*(y, \alpha_0) U_\theta(y, \alpha_0, \theta_0)^\top\}.$$

Under the assumption that the margins and copula are correctly specified, one has  $\mathcal{I}_{\alpha,\theta} = K_{\alpha,\theta}^*$ .

The proof is given in the Online Supplement [26].

**Lemma 5.** Under the assumption that the margins and copula are correctly specified, one has  $\mathcal{I}_\theta = K_\theta$ .

The proof is given in the Online Supplement [26].

### 3.3. Impact of model misspecification on the copula parameter

The most apparent consequence of margin misspecification is that the two-stage ML copula parameter estimate  $\tilde{\theta}$  is no longer consistent for the true parameter value. Another consequence is the change in asymptotic variance  $V_\eta$ . Applying the lemmas resulting from true model assumption to (3) gives

$$V_\eta^{\text{TMA}} = \begin{pmatrix} V_\alpha^{\text{TMA}} & V_{\alpha,\theta}^{\text{TMA}} \\ (V_{\alpha,\theta}^{\text{TMA}})^\top & V_\theta^{\text{TMA}} \end{pmatrix} = \begin{pmatrix} \mathcal{I}_\alpha^{-1} & 0 \\ -\mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} & \mathcal{I}_\theta^{-1} \end{pmatrix} \begin{pmatrix} K_\alpha^{\text{TMA}} & 0 \\ 0 & \mathcal{I}_\theta \end{pmatrix} \begin{pmatrix} \mathcal{I}_\alpha^{-1} & -\mathcal{I}_\alpha^{-1} \mathcal{I}_{\alpha,\theta} \mathcal{I}_\theta^{-1} \\ 0 & \mathcal{I}_\theta^{-1} \end{pmatrix}, \quad (4)$$

where

$$V_\alpha^{\text{TMA}} = \mathcal{I}_\alpha^{-1} K_\alpha^{\text{TMA}} \mathcal{I}_\alpha^{-1}, \quad V_\theta^{\text{TMA}} = \mathcal{I}_\theta^{-1} + \mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} K_\alpha^{\text{TMA}} \mathcal{I}_\alpha^{-1} \mathcal{I}_{\alpha,\theta} \mathcal{I}_\theta^{-1}, \quad V_{\alpha,\theta}^{\text{TMA}} = -\mathcal{I}_\alpha^{-1} K_\alpha^{\text{TMA}} \mathcal{I}_\alpha^{-1} \mathcal{I}_{\alpha,\theta} \mathcal{I}_\theta^{-1}.$$

In Section 5 we illustrate the difference between (3) and (4) in practice by using simulated data. In Section 6, we illustrate the same difference by using real-life data.

## 4. Inference

Having established limiting normality of the two-stage ML estimator  $\tilde{\eta}$  in Proposition 1, we can derive limiting normality also for smooth parameter functions  $\tilde{\mu} = \mu(\tilde{\eta})$ , via the delta method. To use such results in practice we need consistent estimators of all variances.

The general formula for the variance matrix of the limiting distribution of  $\tilde{\eta}$  has the form  $V_\eta = \mathcal{I}_\eta^{-1} K_\eta (\mathcal{I}_\eta^{-1})^\top$ . These components are population quantities, defined as means and variance matrices of random variables. Consistent estimators for these components emerge generally speaking by using plug-in sample averages  $\tilde{h}_n = \sum_{i=1}^n h(y_i, \tilde{\eta})/n$  for estimating a required  $h_0 = E_G[h(y, \eta_0)]$ . For regularity conditions that secure convergence in probability of such  $\tilde{h}_n$  to  $h_0$ , see [22]. This general recipe leads to consistent estimators of  $\mathcal{I}_\eta^{-1}$ ,  $K_\eta$  above, and hence of  $\tilde{V}_\eta = \tilde{\mathcal{I}}_\eta^{-1} \tilde{K}_\eta (\tilde{\mathcal{I}}_\eta^{-1})^\top$ .

Consider now such a focus parameter  $\mu = \mu(\eta) = \mu(\alpha, \theta)$ , a parameter of primary interest, a smooth function of the model parameters. In addition to the ML estimator  $\tilde{\mu} = \mu(\tilde{\alpha}, \tilde{\theta})$ , we may define the associated two-stage ML estimator  $\tilde{\mu} = \mu(\tilde{\alpha}, \tilde{\theta})$ , for which the delta method yields

$$\sqrt{n}(\tilde{\mu} - \mu_0) \rightsquigarrow \mathcal{N}(0, \tau^2), \quad (5)$$

with  $\tau^2 = c^\top V_\eta c$ , and  $c = \partial \mu(\eta_0)/\partial \eta_0$ . Then  $\tilde{\tau}^2 = \tilde{c}^\top \tilde{V}_\eta \tilde{c}$  is consistent for  $\tau^2$ , with  $\tilde{c} = \partial \mu(\tilde{\eta})/\partial \eta$ . This leads to confidence intervals, for any parameter of interest, as with

$$\tilde{\mu} \pm 1.96 \tilde{\tau}/\sqrt{n} \quad (6)$$

for the approximate 95% interval. There are different versions of (6), corresponding to different ways of applying the variance matrix formula  $V_\eta$  above. One may use components for  $\tilde{V}_\eta$  in a model-robust fashion, or the version where the parametric model assumption is trusted, leading to say  $\tilde{V}_\eta^{\text{TMA}}$  and to a consequent  $\tilde{\tau}^{\text{TMA}}$ . Also, results similar to and in fact more familiar than those of (5) and (6) are easy to write down and use for the full ML estimation method, leading to  $\hat{\mu} \pm 1.96 \hat{\tau}/\sqrt{n}$  etc.

We note that full confidence distributions can be computed and displayed, to supplement the two-stage ML based point estimate  $\tilde{\mu}$  and estimated standard deviation  $\tilde{\tau}/\sqrt{n}$ . These are random curves  $cc(\mu)$ , one such for each focus parameter, constructed post data, with the property that  $\Pr\{cc(\mu) \leq \alpha\}$  is equal to or approximately equal to  $\alpha$ , for all confidence levels  $\alpha$ . In addition to the easy to use first-order large-sample confidence curve

$$cc(\mu) = \Phi\{\sqrt{n}(\mu - \tilde{\mu})/\tilde{\tau}\},$$

somewhat more elaborate and better approximations may be constructed via the methods in Chapters 3–4 of Schweder and Hjort [34], involving generalizations of the Wilks type theorems. In Section 6 such confidence curves are computed and displayed for relevant parameters pertaining to Norwegian precipitation data.

Our methodology lends itself nicely also to hypothesis testing. If  $\mu = \mu(\alpha, \theta)$  is a parameter where a certain null value  $\mu_{\text{null}}$  is of interest, then we may test  $\mathcal{H}_{\text{null}}: \mu = \mu_{\text{null}}$  via inspection of the associated confidence interval, or via some fully or nearly equivalent route. This in particular applies for testing independence in the model structure, for the full  $d$ -dimensional vector or for a subset, if this corresponds to a null value for the  $\theta$  parameter.

We may also use the developed machinery to test whether aspects of two or more sets of data are identical or different. If one has data from two groups thought to be not very different, say  $A$  and  $B$ , one may fit the same copula model to both, yielding parameter estimates for  $\eta_A = (\alpha_A, \theta_A)$  and  $\eta_B = (\alpha_B, \theta_B)$ , along with variance estimators. With this taken care as explained above, one may then test the hypothesis  $\mathcal{H}_0: \theta_A = \theta_B$ . Specifically, with sample sizes  $n_A$  and  $n_B$ , we would have

$$\tilde{\theta}_A - \tilde{\theta}_B \approx_d \mathcal{N}(\theta_A - \theta_B, W),$$

say, with  $W = \Sigma_A/n_A + \Sigma_B/n_B$ . Here  $\Sigma_A$  and  $\Sigma_B$  are the variance matrices appearing in the limit distributions for  $\sqrt{n_A}(\tilde{\theta}_A - \theta_A)$  and  $\sqrt{n_B}(\tilde{\theta}_B - \theta_B)$ . The test statistic

$$Z = (\tilde{\theta}_A - \tilde{\theta}_B)^\top \tilde{W}^{-1}(\tilde{\theta}_A - \tilde{\theta}_B),$$

with  $\tilde{W}$  involving consistent estimators for  $\Sigma_A$  and  $\Sigma_B$ , would then follow an approximate  $\chi_q^2$  null distribution, with  $q$  the dimension of  $\theta$ .

**Table 1**  
Description of the models used in simulation 1.

|                       | Copula                     | Margin 1                                                | Margin 2                                           | $\widehat{\text{KL}}(g, f)$ |
|-----------------------|----------------------------|---------------------------------------------------------|----------------------------------------------------|-----------------------------|
| Data generating model | Gaussian<br>$\theta = 0.3$ | Weibull<br>$\alpha_1 = (1.5, 4)^\top$<br>(shape, scale) | Gamma<br>$\alpha_2 = (2, 1)^\top$<br>(shape, rate) |                             |
| Model 1               | Gaussian                   | Weibull                                                 | Gamma                                              | 0                           |
| Model 2               | Gaussian                   | Weibull                                                 | Generalized gamma                                  | 0                           |
| Model 3               | Gaussian                   | Log-normal                                              | Log-normal                                         | 0.140                       |
| Model 4               | Student $t$                | Weibull                                                 | Gamma                                              | 0.020                       |
| Model 5               | Student $t$                | Log-normal                                              | Log-normal                                         | 0.166                       |
| Model 6               | Frank                      | Weibull                                                 | Gamma                                              | 0.004                       |
| Model 7               | Frank                      | Log-normal                                              | Log-normal                                         | 0.139                       |

**Table 2**  
Description of the models used in simulation 2.

|                       | Copula                 | Margin 1                                                | Margin 2                                              | Margin 3                                           | Margin 4                                           | $\widehat{\text{KL}}(g, f)$ |
|-----------------------|------------------------|---------------------------------------------------------|-------------------------------------------------------|----------------------------------------------------|----------------------------------------------------|-----------------------------|
| Data generating model | Gumbel<br>$\theta = 3$ | Weibull<br>$\alpha_1 = (1.5, 4)^\top$<br>(shape, scale) | Weibull<br>$\alpha_2 = (2, 3)^\top$<br>(shape, scale) | Gamma<br>$\alpha_3 = (2, 1)^\top$<br>(shape, rate) | Gamma<br>$\alpha_4 = (3, 1)^\top$<br>(shape, rate) |                             |
| Model 1               | Gumbel                 | Weibull                                                 | Weibull                                               | Gamma                                              | Gamma                                              | 0                           |
| Model 2               | Gumbel                 | Weibull                                                 | Weibull                                               | Generalized Gamma                                  | Generalized Gamma                                  | 0                           |
| Model 3               | Gumbel                 | Log-normal                                              | Log-normal                                            | Log-normal                                         | Log-normal                                         | 0.298                       |
| Model 4               | Survival Clayton       | Weibull                                                 | Weibull                                               | Gamma                                              | Gamma                                              | 0.433                       |
| Model 5               | Survival Clayton       | Log-normal                                              | Log-normal                                            | Log-normal                                         | Log-normal                                         | 0.508                       |
| Model 6               | Frank                  | Weibull                                                 | Weibull                                               | Gamma                                              | Gamma                                              | 0.263                       |
| Model 7               | Frank                  | Log-normal                                              | Log-normal                                            | Log-normal                                         | Log-normal                                         | 0.558                       |

## 5. Simulation study

To study the impact of the true model assumption on two-stage ML estimation, we have performed a set of simulations. In simulation 1, we study the impact of the true model assumption on a number of 2-dimensional copula models. In simulation 2, we do the same with 4-dimensional copula models. In simulation 3, we look at the case where the degree of misspecification increases or decreases gradually.

### 5.1. Simulation 1 (2-dimensional case) and simulation 2 (4-dimensional case)

Tables 1 and 2 contain descriptions of the models that were used to generate the data and the models that were used to fit the data in simulation 1 and simulation 2, respectively. The candidate models are chosen in such a way that they illustrate different types and degrees of misspecification. The column  $\widehat{\text{KL}}(g, f)$  indicates estimated Kullback–Leibler divergence with ML estimated least false parameter values. We used two sample sizes  $n = 100$  and  $n = 1000$ . For each sample size, we drew 1000 samples and the results were averaged. Some of the candidate models are easily distinguishable from the data generating model for a sample size 1000, based on likelihood-based model selection criteria such as AIC. However, since our goal is to investigate different degrees of model misspecification, we included them in our simulation.

For each model, ML and two-stage ML estimations were performed both assuming and not assuming that the model is the true model that generated data. The column ‘True model assumption’ in Tables 3–6 indicates the presence of this assumption. Dropping the true model assumption leads to the model robust asymptotic variance formulas. For the ML estimator, this is the so-called ‘sandwich estimator’. For a copula model, this estimator can be obtained straightforwardly by applying classical theory covered in [8,17,38]. When we make the true model assumption, the sandwich estimator simplifies to the inverse of the Fisher information. For two-stage ML estimators, dropping the true model assumption yields (3) as asymptotic variance, and assuming the true model, this simplifies into (4).

Below,  $\hat{\theta}$  and  $\tilde{\theta}$  indicate the ML and the two-stage ML estimate of the copula parameter, respectively. Also,  $\sqrt{n} \text{SE}(\hat{\theta})$  is the square root of the estimated asymptotic variance of  $\hat{\theta}$ . It is estimated by choosing the relevant formula, depending on the presence of true model assumption, and, in general terms, replacing  $E_G\{h(y, \eta_0)\}$  by  $\sum_{i=1}^n h(y_i, \hat{\eta})/n$  or  $\sum_{i=1}^n h(y_i, \tilde{\eta})/n$ ; see the discussion in Section 4.

Next, let  $q_{0.8}$  indicate the vector that contains the 0.8-quantile value of each margin according to the data generating model, i.e.,  $q_{0.8} = (G_1^{-1}(0.8), \dots, G_d^{-1}(0.8))$ . With  $\Pr(q_{0.8} < Y)$  we mean the joint probability that each marginal variable has larger value than the corresponding 0.8-quantile value. This joint upper probability is sensitive to the tail behavior of both margins and copula. The notations  $\widehat{\Pr}(q_{0.8} < Y)$  and  $\tilde{\Pr}(q_{0.8} < Y)$  refer to ML and two-stage ML estimates of this joint probability, respectively. Also,  $\sqrt{n} \text{SE}(\widehat{\Pr}(q_{0.8} < Y))$  is the square root of the estimated asymptotic variance of  $\widehat{\Pr}(q_{0.8} < Y)$ , and  $\sqrt{n} \text{SE}(\tilde{\Pr}(q_{0.8} < Y))$  is the two-stage ML estimator analog. These asymptotic variances are obtained by

**Table 3**  
Result from simulation 1 with sample size 100. For clarification of the column labels, see Section 5.1.

| Simulation 1 ( $n = 100$ ) |           |                       |                                  |                                  |                   |                                          |                                  |                |                                  |
|----------------------------|-----------|-----------------------|----------------------------------|----------------------------------|-------------------|------------------------------------------|----------------------------------|----------------|----------------------------------|
|                            |           | True model assumption | MLE                              |                                  |                   |                                          |                                  |                |                                  |
|                            |           |                       | $\widehat{\theta}$               | (95% CI)                         | Coverage $\theta$ | $\sqrt{n} \text{SE}(\widehat{\theta})$   | $\widehat{\Pr}(q_{0.8} < Y)$     | (95% CI)       | Coverage $\Pr(q_{0.8} < Y)$      |
| Model 1                    | No<br>Yes | 0.298                 | (0.107, 0.489)<br>(0.121, 0.475) | 0.951<br>0.939                   | 0.974<br>0.902    | 0.066                                    | (0.048, 0.085)<br>(0.049, 0.083) | 0.959<br>0.939 | 0.094<br>0.087                   |
| Model 2                    | No<br>Yes |                       | 0.299                            | (0.109, 0.489)<br>(0.122, 0.476) | 0.952<br>0.939    |                                          | 0.970<br>0.904                   | 0.066          | (0.048, 0.085)<br>(0.049, 0.083) |
| Model 3                    | No<br>Yes | 0.298                 |                                  | (0.095, 0.500)<br>(0.121, 0.474) | 0.961<br>0.929    | 1.032<br>0.902                           | 0.066                            |                | (0.047, 0.086)<br>(0.049, 0.083) |
| Model 4                    | No<br>Yes |                       | 0.297                            | (0.098, 0.496)<br>(0.102, 0.491) | n/a<br>n/a        | 1.016<br>0.992                           |                                  | 0.069          | (0.050, 0.088)<br>(0.050, 0.087) |
| Model 5                    | No<br>Yes | 0.303                 |                                  | (0.091, 0.514)<br>(0.102, 0.503) | n/a<br>n/a        | 1.079<br>1.022                           | 0.069                            |                | (0.049, 0.089)<br>(0.050, 0.088) |
| Model 6                    | No<br>Yes |                       | 1.844                            | (0.504, 3.183)<br>(0.597, 3.090) | n/a<br>n/a        | 6.835<br>6.358                           |                                  | 0.065          | (0.047, 0.083)<br>(0.048, 0.082) |
| Model 7                    | No<br>Yes | 2.003                 |                                  | (0.527, 3.480)<br>(0.682, 3.325) | n/a<br>n/a        | 7.535<br>6.742                           | 0.067                            |                | (0.047, 0.087)<br>(0.049, 0.085) |
|                            |           |                       | True model assumption            | Two-stage MLE                    |                   |                                          |                                  |                |                                  |
|                            |           | $\widetilde{\theta}$  |                                  | (95% CI)                         | Coverage $\theta$ | $\sqrt{n} \text{SE}(\widetilde{\theta})$ | $\widetilde{\Pr}(q_{0.8} < Y)$   | (95% CI)       | Coverage $\Pr(q_{0.8} < Y)$      |
| Model 1                    | No<br>Yes | 0.298                 | (0.107, 0.488)<br>(0.121, 0.475) | 0.950<br>0.939                   | 0.973<br>0.903    | 0.066                                    | (0.048, 0.085)<br>(0.049, 0.083) | 0.957<br>0.938 | 0.094<br>0.087                   |
| Model 2                    | No<br>Yes |                       | 0.297                            | (0.108, 0.487)<br>(0.120, 0.475) | 0.951<br>0.941    |                                          | 0.967<br>0.905                   | 0.066          | (0.048, 0.085)<br>(0.049, 0.083) |
| Model 3                    | No<br>Yes | 0.298                 |                                  | (0.095, 0.500)<br>(0.117, 0.478) | 0.961<br>0.936    | 1.032<br>0.923                           | 0.066                            |                | (0.047, 0.086)<br>(0.049, 0.084) |
| Model 4                    | No<br>Yes |                       | 0.296                            | (0.101, 0.491)<br>(0.103, 0.490) | n/a<br>n/a        | 0.995<br>0.986                           |                                  | 0.069          | (0.050, 0.087)<br>(0.050, 0.087) |
| Model 5                    | No<br>Yes | 0.302                 |                                  | (0.094, 0.510)<br>(0.098, 0.505) | n/a<br>n/a        | 1.062<br>1.038                           | 0.069                            |                | (0.049, 0.089)<br>(0.049, 0.088) |
| Model 6                    | No<br>Yes |                       | 1.828                            | (0.506, 3.150)<br>(0.583, 3.072) | n/a<br>n/a        | 6.746<br>6.350                           |                                  | 0.065          | (0.047, 0.083)<br>(0.048, 0.082) |
| Model 7                    | No<br>Yes | 1.974                 |                                  | (0.534, 3.415)<br>(0.638, 3.311) | n/a<br>n/a        | 7.351<br>6.819                           | 0.067                            |                | (0.047, 0.086)<br>(0.049, 0.085) |

writing  $\Pr(q_{0.8} < Y)$  as a function of  $\eta$  and applying the delta method. With those standard errors, we also constructed confidence intervals. The reported confidence intervals in Tables 3–6 are obtained by averaging the confidence intervals from 1000 samples.

In the cases where the true value of  $\theta$  or  $\Pr(q_{0.8} < Y)$  was known, we also computed coverage to supplement the confidence interval. The coverage is computed as the number of times that the confidence interval captured the true value divided by the total number of simulations.

When estimating the above mentioned quantities, the main computational bottleneck is estimating the  $K$ -matrices. For instance, the matrix

$$\mathcal{I}_{\theta} = - \int g \frac{\partial^2}{\partial \theta_0 \partial \theta_0^{\top}} \ln c\{F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d}), \theta\} \, \mathrm{d}y$$

is estimated by

$$\tilde{\mathcal{I}}_{\theta} = - \frac{1}{n} \sum_{i=1}^n \frac{\partial^2}{\partial \theta \partial \theta^{\top}} \ln c\{F_1(y_{i,1}, \tilde{\alpha}_1), \dots, F_d(y_{i,d}, \tilde{\alpha}_d), \tilde{\theta}\}.$$

**Table 4**

Result from simulation 1 with sample size 1000. For clarification of the column labels, see Section 5.1.

| Simulation 1 ( $n = 1000$ ) |     |                       |                      |                |                   |                                          |                                |                |                             |                                                    |
|-----------------------------|-----|-----------------------|----------------------|----------------|-------------------|------------------------------------------|--------------------------------|----------------|-----------------------------|----------------------------------------------------|
|                             |     | True model assumption | MLE                  |                |                   |                                          |                                |                |                             |                                                    |
|                             |     |                       | $\widehat{\theta}$   | (95% CI)       | Coverage $\theta$ | $\sqrt{n} \text{SE}(\widehat{\theta})$   | $\widehat{\Pr}(q_{0.8} < Y)$   | (95% CI)       | Coverage $\Pr(q_{0.8} < Y)$ | $\sqrt{n} \text{SE}(\widehat{\Pr}(q_{0.8} < Y))$   |
| Model 1                     | No  | Yes                   | 0.298                | (0.241, 0.356) | 0.950             | 0.930                                    | 0.066                          | (0.060, 0.072) | 0.954                       | 0.090                                              |
|                             | Yes |                       |                      | (0.242, 0.355) | 0.947             | 0.908                                    |                                | (0.061, 0.071) | 0.948                       | 0.088                                              |
| Model 2                     | No  | Yes                   | 0.299                | (0.241, 0.356) | 0.953             | 0.929                                    | 0.066                          | (0.060, 0.072) | 0.954                       | 0.090                                              |
|                             | Yes |                       |                      | (0.242, 0.355) | 0.947             | 0.909                                    |                                | (0.061, 0.071) | 0.949                       | 0.088                                              |
| Model 3                     | No  | Yes                   | 0.291                | (0.230, 0.352) | 0.942             | 0.987                                    | 0.065                          | (0.059, 0.071) | 0.938                       | 0.095                                              |
|                             | Yes |                       |                      | (0.234, 0.347) | 0.927             | 0.914                                    |                                | (0.060, 0.071) | 0.919                       | 0.088                                              |
| Model 4                     | No  | Yes                   | 0.299                | (0.241, 0.356) | n/a               | 0.930                                    | 0.067                          | (0.061, 0.072) | 0.950                       | 0.089                                              |
|                             | Yes |                       |                      | (0.241, 0.356) | n/a               | 0.930                                    |                                | (0.061, 0.072) | 0.951                       | 0.090                                              |
| Model 5                     | No  | Yes                   | 0.294                | (0.233, 0.355) | n/a               | 0.985                                    | 0.066                          | (0.060, 0.072) | 0.939                       | 0.094                                              |
|                             | Yes |                       |                      | (0.234, 0.353) | n/a               | 0.964                                    |                                | (0.061, 0.072) | 0.936                       | 0.092                                              |
| Model 6                     | No  | Yes                   | 1.806                | (1.407, 2.204) | n/a               | 6.422                                    | 0.064                          | (0.059, 0.070) | 0.914                       | 0.088                                              |
|                             | Yes |                       |                      | (1.417, 2.194) | n/a               | 6.269                                    |                                | (0.059, 0.070) | 0.905                       | 0.086                                              |
| Model 7                     | No  | Yes                   | 1.958                | (1.520, 2.397) | n/a               | 7.076                                    | 0.067                          | (0.061, 0.073) | 0.944                       | 0.096                                              |
|                             | Yes |                       |                      | (1.544, 2.372) | n/a               | 6.674                                    |                                | (0.061, 0.072) | 0.932                       | 0.091                                              |
|                             |     | True model assumption | Two-stage MLE        |                |                   |                                          |                                |                |                             |                                                    |
|                             |     |                       | $\widetilde{\theta}$ | (95% CI)       | Coverage $\theta$ | $\sqrt{n} \text{SE}(\widetilde{\theta})$ | $\widetilde{\Pr}(q_{0.8} < Y)$ | (95% CI)       | Coverage $\Pr(q_{0.8} < Y)$ | $\sqrt{n} \text{SE}(\widetilde{\Pr}(q_{0.8} < Y))$ |
| Model 1                     | No  | Yes                   | 0.298                | (0.241, 0.356) | 0.950             | 0.929                                    | 0.066                          | (0.060, 0.072) | 0.954                       | 0.090                                              |
|                             | Yes |                       |                      | (0.242, 0.355) | 0.947             | 0.909                                    |                                | (0.061, 0.071) | 0.948                       | 0.088                                              |
| Model 2                     | No  | Yes                   | 0.298                | (0.241, 0.356) | 0.952             | 0.929                                    | 0.066                          | (0.060, 0.072) | 0.954                       | 0.090                                              |
|                             | Yes |                       |                      | (0.242, 0.355) | 0.948             | 0.909                                    |                                | (0.061, 0.071) | 0.949                       | 0.088                                              |
| Model 3                     | No  | Yes                   | 0.291                | (0.230, 0.352) | 0.942             | 0.987                                    | 0.065                          | (0.059, 0.071) | 0.938                       | 0.095                                              |
|                             | Yes |                       |                      | (0.233, 0.349) | 0.931             | 0.938                                    |                                | (0.060, 0.071) | 0.926                       | 0.090                                              |
| Model 4                     | No  | Yes                   | 0.298                | (0.241, 0.356) | n/a               | 0.929                                    | 0.067                          | (0.061, 0.072) | 0.949                       | 0.089                                              |
|                             | Yes |                       |                      | (0.241, 0.356) | n/a               | 0.930                                    |                                | (0.061, 0.072) | 0.950                       | 0.090                                              |
| Model 5                     | No  | Yes                   | 0.294                | (0.233, 0.354) | n/a               | 0.983                                    | 0.066                          | (0.060, 0.072) | 0.940                       | 0.094                                              |
|                             | Yes |                       |                      | (0.232, 0.355) | n/a               | 0.988                                    |                                | (0.060, 0.072) | 0.942                       | 0.095                                              |
| Model 6                     | No  | Yes                   | 1.801                | (1.405, 2.197) | n/a               | 6.385                                    | 0.064                          | (0.059, 0.070) | 0.913                       | 0.088                                              |
|                             | Yes |                       |                      | (1.413, 2.189) | n/a               | 6.260                                    |                                | (0.059, 0.070) | 0.905                       | 0.086                                              |
| Model 7                     | No  | Yes                   | 1.944                | (1.511, 2.376) | n/a               | 6.982                                    | 0.066                          | (0.060, 0.072) | 0.946                       | 0.095                                              |
|                             | Yes |                       |                      | (1.525, 2.362) | n/a               | 6.748                                    |                                | (0.061, 0.072) | 0.940                       | 0.092                                              |

When calculating this matrix, a for-loop can be avoided by swapping the order of summation and differentiation. When computing

$$\tilde{K}_{\theta} = \frac{1}{n} \sum_{i=1}^n U_{\theta}(y_i, \tilde{\alpha}, \tilde{\theta}) U_{\theta}(y_i, \tilde{\alpha}, \tilde{\theta})^{\top},$$

however, the same trick cannot be used and we are forced to use a time-consuming for-loop.

Now we look at the results from the simulations. From the result of simulations 1 and 2, we can observe that models that are correctly specified or have a small degree of misspecification (e.g., Student  $t$ -copula instead of Gaussian copula and Generalized Gamma distribution instead of Gamma distribution), the presence of the true model assumption makes virtually no difference. This confirms the finding from Section 3 that when the model is correctly specified, the model robust variance formula (3) and the non-robust version (4) coincide. Further, for the ML estimation part, it echoes the classical finding that the ‘sandwich estimator’ becomes the inverse of the Fisher information when the model is correctly specified.

When the degree of misspecification increases (e.g., models 5 and 7 in simulation 2 where the log-normal density has a heavier tail than the data generating margins and the Frank copula has lighter tails than the Gumbel copula), we see that assuming the true model (i.e., using (4) instead of (3)) results in a larger difference in estimated the asymptotic variance. Often, assuming the true model results in a smaller asymptotic variance. This sounds intuitive because assuming the true model entails ignoring the model uncertainty and thus reduces the variance. In other words, ignoring the model uncertainty often leads to over-confident confidence intervals. The degree of over-confidence will be larger when the sample size is small, because the asymptotic variance is divided by the sample size when a confidence interval is computed. This also implies that

**Table 5**

Result from simulation 2 with sample size 100. For clarification of the column labels, see Section 5.1.

| Simulation 2 ( $n = 100$ ) |     |                       |                  |          |                   |                                      |                            |          |                             |
|----------------------------|-----|-----------------------|------------------|----------|-------------------|--------------------------------------|----------------------------|----------|-----------------------------|
|                            |     | True model assumption | MLE              |          |                   |                                      |                            |          |                             |
|                            |     |                       | $\hat{\theta}$   | (95% CI) | Coverage $\theta$ | $\sqrt{n} \text{SE}(\hat{\theta})$   | $\hat{\Pr}(q_{0.8} < Y)$   | (95% CI) | Coverage $\Pr(q_{0.8} < Y)$ |
| Model 1                    | No  | 3.022                 | (2.582, 3.462)   | 0.95     | 2.246             | 0.132                                | (0.122, 0.142)             | 0.943    | 0.050                       |
|                            | Yes |                       | (2.571, 3.473)   | 0.954    | 2.302             |                                      | (0.122, 0.142)             | 0.953    | 0.051                       |
| Model 2                    | No  | 3.024                 | (2.552, 3.496)   | 0.95     | 2.407             | 0.132                                | (0.122, 0.143)             | 0.944    | 0.054                       |
|                            | Yes |                       | (2.570, 3.478)   | 0.953    | 2.315             |                                      | (0.122, 0.142)             | 0.944    | 0.051                       |
| Model 3                    | No  | 2.869                 | (2.447, 3.291)   | 0.863    | 2.153             | 0.128                                | (0.118, 0.139)             | 0.905    | 0.053                       |
|                            | Yes |                       | (2.469, 3.268)   | 0.858    | 2.039             |                                      | (0.119, 0.138)             | 0.905    | 0.050                       |
| Model 4                    | No  | 2.867                 | (2.010, 3.724)   | n/a      | 4.374             | 0.123                                | (0.105, 0.140)             | 0.855    | 0.089                       |
|                            | Yes |                       | (2.108, 3.626)   | n/a      | 3.874             |                                      | (0.107, 0.138)             | 0.797    | 0.078                       |
| Model 5                    | No  | 3.907                 | (2.793, 5.021)   | n/a      | 5.683             | 0.139                                | (0.125, 0.154)             | 0.755    | 0.072                       |
|                            | Yes |                       | (2.970, 4.843)   | n/a      | 4.779             |                                      | (0.127, 0.151)             | 0.674    | 0.061                       |
| Model 6                    | No  | 11.160                | (8.667, 13.652)  | n/a      | 12.715            | 0.107                                | (0.092, 0.122)             | 0.136    | 0.077                       |
|                            | Yes |                       | (9.375, 12.944)  | n/a      | 9.106             |                                      | (0.096, 0.118)             | 0.040    | 0.056                       |
| Model 7                    | No  | 13.839                | (10.242, 17.436) | n/a      | 18.350            | 0.121                                | (0.104, 0.138)             | 0.815    | 0.086                       |
|                            | Yes |                       | (11.421, 16.258) | n/a      | 12.340            |                                      | (0.110, 0.132)             | 0.545    | 0.058                       |
|                            |     | True model assumption | Two-stage MLE    |          |                   |                                      |                            |          |                             |
|                            |     |                       | $\tilde{\theta}$ | (95% CI) | Coverage $\theta$ | $\sqrt{n} \text{SE}(\tilde{\theta})$ | $\tilde{\Pr}(q_{0.8} < Y)$ | (95% CI) | Coverage $\Pr(q_{0.8} < Y)$ |
| Model 1                    | No  | 3.001                 | (2.514, 3.487)   | 0.931    | 2.484             | 0.131                                | (0.121, 0.142)             | 0.941    | 0.056                       |
|                            | Yes |                       | (2.508, 3.493)   | 0.936    | 2.512             |                                      | (0.120, 0.143)             | 0.941    | 0.057                       |
| Model 2                    | No  | 2.989                 | (2.502, 3.475)   | 0.935    | 2.482             | 0.131                                | (0.120, 0.142)             | 0.939    | 0.056                       |
|                            | Yes |                       | (2.493, 3.484)   | 0.944    | 2.527             |                                      | (0.120, 0.142)             | 0.942    | 0.058                       |
| Model 3                    | No  | 2.980                 | (2.524, 3.435)   | 0.947    | 2.323             | 0.131                                | (0.121, 0.141)             | 0.957    | 0.053                       |
|                            | Yes |                       | (2.549, 3.411)   | 0.938    | 2.199             |                                      | (0.121, 0.141)             | 0.941    | 0.050                       |
| Model 4                    | No  | 2.619                 | (1.852, 3.386)   | n/a      | 3.912             | 0.117                                | (0.100, 0.135)             | 0.671    | 0.090                       |
|                            | Yes |                       | (1.893, 3.346)   | n/a      | 3.706             |                                      | (0.101, 0.134)             | 0.621    | 0.085                       |
| Model 5                    | No  | 3.180                 | (2.378, 3.983)   | n/a      | 4.094             | 0.129                                | (0.115, 0.143)             | 0.940    | 0.072                       |
|                            | Yes |                       | (2.426, 3.934)   | n/a      | 3.846             |                                      | (0.116, 0.142)             | 0.913    | 0.067                       |
| Model 6                    | No  | 9.884                 | (7.994, 11.775)  | n/a      | 9.646             | 0.099                                | (0.085, 0.112)             | 0        | 0.068                       |
|                            | Yes |                       | (8.322, 11.447)  | n/a      | 7.974             |                                      | (0.088, 0.110)             | 0        | 0.056                       |
| Model 7                    | No  | 9.637                 | (7.652, 11.623)  | n/a      | 10.130            | 0.097                                | (0.083, 0.111)             | 0.001    | 0.073                       |
|                            | Yes |                       | (8.109, 11.165)  | n/a      | 7.797             |                                      | (0.086, 0.108)             | 0        | 0.056                       |

when hypothesis testing is carried out on the copula parameter or on a function that takes the copula parameter as an input, the choice between the model-robust and model non-robust variance formula can lead to different outcomes of the test.

Although the true model assumption often results in a smaller variance of the copula parameter, there are cases when this does not hold, e.g., model 1 from simulation 3 in Section 5.2 and the model for precipitation data in Section 6. This is in line with the fact that we could not find any analytical evidence for the inequality  $V_{\theta}^{\text{TMA}} < V_{\theta}$ . In addition, the true model assumption changes the interpretation of  $\tilde{\theta}$  from ‘estimate of the least false parameter value’ to ‘estimate of the true value that generated the data’.

Joe [20] carried out an extensive study on the asymptotic efficiency for two-stage ML estimators and concluded that two-stage estimation is highly efficient in most cases. By comparing the results from ML and two-stage ML estimations, we can largely confirm that this is also the case when the model robust variance formula is used, particularly when the sample size  $n$  is large, as the two variance matrix estimators involved are estimating the same quantity.

Another notable result regarding efficiency is that the Hájek–Le Cam convolution theorem, and related theorems on the optimality of ML estimation, do not apply when the model is incorrectly specified. The Hájek–Le Cam convolution theorem [15], combined with the Cramér–Rao lower bound [30] theory, states that no asymptotically unbiased and normal competing estimator can have a smaller limiting variance matrix than that of the full ML estimator. Models 5 and 7 in both the two- and four-dimensional cases have a relatively large degree of model misspecification. For these models, it frequently happens that the asymptotic variance of the two-stage ML estimator is smaller than that of the full ML estimator. This occurs both when the model-robust and model non-robust variance formulas are used. The largest difference occurs for model 7 from simulation 2, where the degree of model misspecification is high (the copula and all four margins are misspecified).

**Table 6**

Result from simulation 2 with sample size 1000. For clarification of the column labels, see Section 5.1.

| Simulation 2 ( $n = 1000$ ) |     |                  |                  |                   |                                      |                            |                |                             |                                                |
|-----------------------------|-----|------------------|------------------|-------------------|--------------------------------------|----------------------------|----------------|-----------------------------|------------------------------------------------|
| True model assumption       |     | MLE              |                  |                   |                                      |                            |                |                             |                                                |
|                             |     | $\hat{\theta}$   | (95% CI)         | Coverage $\theta$ | $\sqrt{n} \text{SE}(\hat{\theta})$   | $\hat{\Pr}(q_{0.8} < Y)$   | (95% CI)       | Coverage $\Pr(q_{0.8} < Y)$ | $\sqrt{n} \text{SE}(\hat{\Pr}(q_{0.8} < Y))$   |
| Model 1                     | No  | 3.002            | (2.861, 3.143)   | 0.947             | 2.271                                | 0.132                      | (0.129, 0.135) | 0.946                       | 0.051                                          |
|                             | Yes |                  | (2.861, 3.143)   | 0.949             | 2.278                                |                            | (0.129, 0.135) | 0.949                       | 0.051                                          |
| Model 2                     | No  | 3.002            | (2.848, 3.156)   | 0.950             | 2.482                                | 0.132                      | (0.129, 0.135) | 0.945                       | 0.056                                          |
|                             | Yes |                  | (2.860, 3.144)   | 0.951             | 2.287                                |                            | (0.129, 0.135) | 0.948                       | 0.051                                          |
| Model 3                     | No  | 2.853            | (2.713, 2.993)   | 0.460             | 2.255                                | 0.128                      | (0.125, 0.132) | 0.500                       | 0.056                                          |
|                             | Yes |                  | (2.728, 2.978)   | 0.382             | 2.019                                |                            | (0.125, 0.132) | 0.409                       | 0.050                                          |
| Model 4                     | No  | 2.785            | (2.516, 3.055)   | n/a               | 4.346                                | 0.122                      | (0.116, 0.128) | 0.050                       | 0.092                                          |
|                             | Yes |                  | (2.551, 3.019)   | n/a               | 3.780                                |                            | (0.117, 0.127) | 0.024                       | 0.080                                          |
| Model 5                     | No  | 3.862            | (3.487, 4.237)   | n/a               | 6.049                                | 0.140                      | (0.135, 0.144) | 0.133                       | 0.078                                          |
|                             | Yes |                  | (3.569, 4.155)   | n/a               | 4.728                                |                            | (0.136, 0.143) | 0.064                       | 0.061                                          |
| Model 6                     | No  | 11.067           | (10.234, 11.900) | n/a               | 13.436                               | 0.107                      | (0.102, 0.112) | 0                           | 0.083                                          |
|                             | Yes |                  | (10.509, 11.624) | n/a               | 8.997                                |                            | (0.103, 0.110) | 0                           | 0.056                                          |
| Model 7                     | No  | 13.777           | (12.619, 14.934) | n/a               | 18.677                               | 0.121                      | (0.116, 0.127) | 0.019                       | 0.088                                          |
|                             | Yes |                  | (13.019, 14.535) | n/a               | 12.227                               |                            | (0.118, 0.125) | 0.002                       | 0.057                                          |
| True model assumption       |     | Two-stage MLE    |                  |                   |                                      |                            |                |                             |                                                |
|                             |     | $\tilde{\theta}$ | (95% CI)         | Coverage $\theta$ | $\sqrt{n} \text{SE}(\tilde{\theta})$ | $\tilde{\Pr}(q_{0.8} < Y)$ | (95% CI)       | Coverage $\Pr(q_{0.8} < Y)$ | $\sqrt{n} \text{SE}(\tilde{\Pr}(q_{0.8} < Y))$ |
| Model 1                     | No  | 2.999            | (2.842, 3.156)   | 0.960             | 2.532                                | 0.132                      | (0.128, 0.135) | 0.962                       | 0.057                                          |
|                             | Yes |                  | (2.842, 3.156)   | 0.964             | 2.537                                |                            | (0.128, 0.135) | 0.960                       | 0.057                                          |
| Model 2                     | No  | 2.997            | (2.840, 3.153)   | 0.961             | 2.527                                | 0.132                      | (0.128, 0.135) | 0.961                       | 0.057                                          |
|                             | Yes |                  | (2.839, 3.154)   | 0.961             | 2.539                                |                            | (0.128, 0.135) | 0.960                       | 0.057                                          |
| Model 3                     | No  | 2.987            | (2.838, 3.136)   | 0.943             | 2.399                                | 0.132                      | (0.128, 0.135) | 0.945                       | 0.054                                          |
|                             | Yes |                  | (2.846, 3.128)   | 0.932             | 2.277                                |                            | (0.128, 0.135) | 0.938                       | 0.052                                          |
| Model 4                     | No  | 2.586            | (2.339, 2.833)   | n/a               | 3.985                                | 0.117                      | (0.112, 0.123) | 0                           | 0.093                                          |
|                             | Yes |                  | (2.357, 2.816)   | n/a               | 3.703                                |                            | (0.112, 0.123) | 0                           | 0.086                                          |
| Model 5                     | No  | 3.178            | (2.915, 3.441)   | n/a               | 4.248                                | 0.129                      | (0.125, 0.134) | 0.839                       | 0.074                                          |
|                             | Yes |                  | (2.937, 3.419)   | n/a               | 3.887                                |                            | (0.125, 0.134) | 0.791                       | 0.068                                          |
| Model 6                     | No  | 9.856            | (9.254, 10.459)  | n/a               | 9.723                                | 0.099                      | (0.095, 0.103) | 0                           | 0.069                                          |
|                             | Yes |                  | (9.358, 10.355)  | n/a               | 8.040                                |                            | (0.095, 0.102) | 0                           | 0.057                                          |
| Model 7                     | No  | 9.647            | (9.007, 10.288)  | n/a               | 10.337                               | 0.097                      | (0.093, 0.102) | 0                           | 0.075                                          |
|                             | Yes |                  | (9.153, 10.142)  | n/a               | 7.980                                |                            | (0.094, 0.101) | 0                           | 0.058                                          |

When it comes to the asymptotic variance of the upper tail probability  $\Pr(q_{0.8} < Y)$ , we often observe that the true model assumption decreases the asymptotic variance. Yet, in some cases, we observe that the true model assumption increases the asymptotic variance. This happens mostly when the true model assumption increases the asymptotic variance of  $\theta$ , but there are also cases where the true model assumption increases the asymptotic  $\Pr(q_{0.8} < Y)$  while it decreases the asymptotic variance of  $\theta$ . We suspect that this is due to the fact that the whole  $V_{\eta}$ , including its non-diagonal elements, is used through the delta method formula to compute  $V_{\Pr(q_{0.8} < Y)}$ . To check whether this also happens in non-copula models, we simulated a large number of univariate data and fitted well-known distributions and computed upper tail probability from them. We could observe that the true model assumption can increase the limiting variance of the upper tail probability while the limiting variance of parameters decreases. Further, from the same univariate simulation, we also observed that using the true model assumption sometimes increases the limiting variance of parameters.

## 5.2. Simulation 3 (gradual change in misspecification)

To study the difference in the numerical behavior between (4) and (3) in more detail, including the cases where the degree of misspecification is very small, we ran a simulation with gradual change in misspecification. There are three scenarios and for each scenario, we generated a dataset of size 100. The data generating algorithm generated  $(1 - \delta) \times 100\%$  of data points from model 1 and  $\delta \times 100\%$  data points from model 2. So,  $\delta$  can be seen as the degree of ‘contamination’ to model 1 by model 2. Then, we fitted both model 1 and model 2 to these data by using two-stage ML. This process was repeated 100,000 times and the results were averaged. We used 100 different values of  $\delta$ , which were spread evenly between 0 and 1. Table 7 describes the model 1 and model 2 that were used in each scenario. In scenario 1, model 1 gets contaminated only in the

**Table 7**  
Description of the scenarios used in simulation 3.

|            |         | Copula                     | Margin 1                                                 | Margin 2                                                 |
|------------|---------|----------------------------|----------------------------------------------------------|----------------------------------------------------------|
| Scenario 1 | Model 1 | Gaussian<br>$\theta = 0.3$ | Weibull<br>$\alpha_1 = (1, 3)^\top$<br>(shape, scale)    | Gamma<br>$\alpha_2 = (2, 1)^\top$<br>(shape, rate)       |
|            | Model 2 | Gaussian<br>$\theta = 0.3$ | Log-normal<br>$\alpha_1 = (0.5, 1.3)^\top$<br>(mean, SD) | Log-normal<br>$\alpha_2 = (0.4, 0.8)^\top$<br>(mean, SD) |
| Scenario 2 | Model 1 | Gaussian<br>$\theta = 0.3$ | Weibull<br>$\alpha_1 = (1, 3)^\top$<br>(shape, scale)    | Gamma<br>$\alpha_2 = (2, 1)^\top$<br>(shape, rate)       |
|            | Model 2 | Frank<br>$\theta = 3$      | Weibull<br>$\alpha_1 = (1, 3)^\top$<br>(shape, scale)    | Gamma<br>$\alpha_2 = (2, 1)^\top$<br>(shape, rate)       |
| Scenario 3 | Model 1 | Gaussian<br>$\theta = 0.3$ | Weibull<br>$\alpha_1 = (1, 3)^\top$<br>(shape, scale)    | Gamma<br>$\alpha_2 = (2, 1)^\top$<br>(shape, rate)       |
|            | Model 2 | Frank<br>$\theta = 3$      | Log-normal<br>$\alpha_1 = (0.5, 1.3)^\top$<br>(mean, SD) | Log-normal<br>$\alpha_2 = (0.4, 0.8)^\top$<br>(mean, SD) |

**Table 8**  
Description of the model for precipitation data.

|       | Copula | Margin 1<br>(Vestby) | Margin 2<br>(Ski) | Margin 3<br>(Lørenskog) | Margin 4<br>(Nannestad) | Margin 5<br>(Hurdal) |
|-------|--------|----------------------|-------------------|-------------------------|-------------------------|----------------------|
| Model | Gumbel | Gamma                | Gamma             | Log-normal              | Gamma                   | Log-normal           |

margins. In scenario 2, model 1 gets contaminated only in the copula. In scenario 3, model 1 gets contaminated both in copula and margins. For each model and scenario, we estimated the asymptotic variance of  $\theta$  and  $\Pr(q_{0.8} < Y)$  by assuming the true model and using Eq. (4) and by not assuming the true model and using Eq. (3).

Figs. 1, 2 and 3 show the results of scenarios 1, 2 and 3, respectively. We can observe that the true model assumption can both decrease and increase the estimated asymptotic variance of  $\theta$  and  $\Pr(q_{0.8} < Y)$ . And it seems like it is model-specific. For example, model 1, which is same across all three scenarios, always shows increased variance of copula parameter  $\theta$  under the true model assumption, regardless of what kind of contamination it gets.

A general pattern that we can clearly observe in all scenarios is that the difference between the two asymptotic variance formulas (4) and (3), regardless of whether it is positive or negative, gets smaller as the degree of misspecification decreases. As expected, this pattern is most clearly visible in scenario 3 where both copula and margins are misspecified.

## 6. Precipitation data

The precipitation data consist of daily measurements of precipitation in millimeters at five different meteorological stations in Norway (Vestby, Ski, Lørenskog, Nannestad and Hurdal) from January 1, 1990 to December 31, 2006. These data were provided by the Norwegian Meteorological Institute and used previously in Aas and Berg [1] (with one station less) and Hobæk Haff [18]. Following the example of two previous papers that used these data, we modeled only positive precipitation by removing all observations for which at least one of the stations has recorded zero precipitation, resulting in 5536 observations. The main advantage of this is that we remove time dependence.

To choose an adequate model, we first fitted a set of well-known distributions for each margin by using ML estimation and evaluated them by AIC. After the marginal distributions were chosen, we performed a probability integral transformation for each margin and fitted a set of well-known copulas by using ML estimation and evaluated these by AIC. The best model obtained in this fashion is described in Table 8. The choice of the Gumbel copula corresponds well with the fact that there are indications of strong upper, but not lower, tail dependence. This is visible in Figure 2 from the Online Supplement [26], which contains simulated scatter plots between the first two variables of the 5-dimensional Gumbel copula. Since the Gumbel copula is a member of the Archimedean copula family, the scatter plots between other possible combinations among the five variables will be virtually the same as the one between variable 1 and variable 2. They are therefore not displayed.

Figure 1 from the Online Supplement [26] shows that the marginal distributions of the fitted model are highly non-normal and can differ between meteorological stations.

Table 9 shows the result from the model described in Table 8. With ML estimation, ignoring model uncertainty gives smaller values of the asymptotic variances. However, with two-stage ML estimators, this is not the case. This phenomenon was discussed earlier in Section 5.

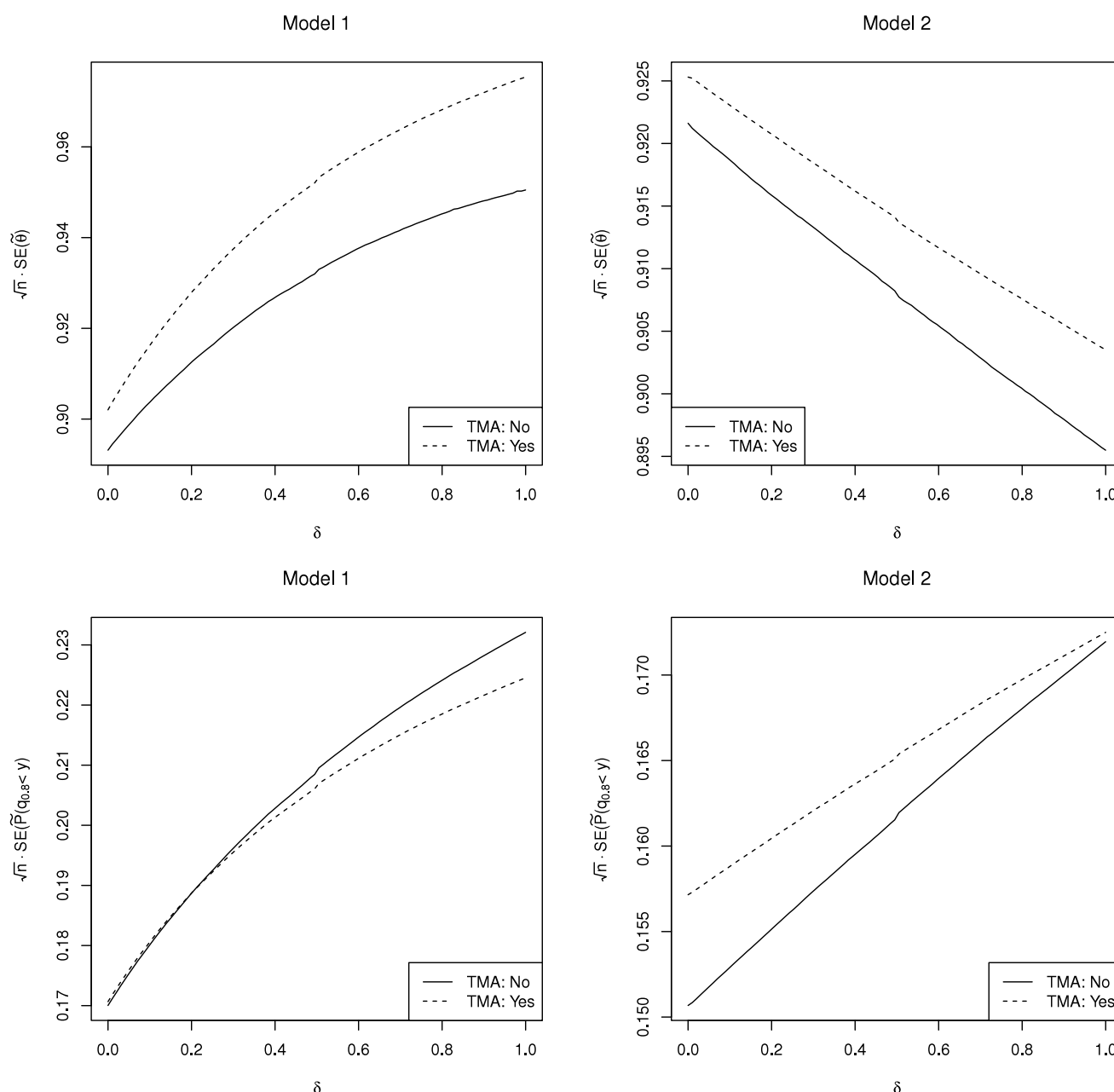


**Fig. 1.** Result of simulation 3, scenario 1. (Models 1 and 2 have the same copula, but different margins.) TMA stands for ‘true model assumption’. The dashed lines are computed with the model non-robust asymptotic variance formula (4) and the solid lines are computed with the model robust asymptotic variance formula (3).

Furthermore, we observe that the asymptotic variance of the two-stage ML estimator is smaller than that of the ML estimator. As already discussed in Section 5, this happens when the degree of misspecification is high and consequently the Hájek–Le Cam convolution theorem is not applicable. In our case, we used the five-dimensional Gumbel copula, which is a member of the Archimedean copula family, to model the precipitation in five locations. The advantage of the Archimedean family copulas is that we can model high-dimensional dependency by using only one parameter. The disadvantage, especially when dimension is high, is that the dependency structure and strength is the same between all different pairs of variables. The dependency relationship between  $u_1$  and  $u_2$ , for example, is the same as the dependency relationship between  $u_3$  and  $u_5$ , in such models. Here  $u_j = F_j(y_j)$ , for  $j \in \{1, \dots, 5\}$ , and these are uniform. The pairwise pseudo-observations plot (Figure 3 in Supplement B of Hobæk Haff [18]), however, shows that the pairwise dependencies among the stations are not the same.

This rigidity of Archimedean copulas is causing a misspecification, and therefore we see that the asymptotic variance of two-stage ML estimators is smaller than that of the full ML. Aas and Berg [1] suggest pair-copula constructions as a method to overcome this limitation.

Fig. 4 visualizes Table 9 by using confidence curves, see Section 4. For details about confidence curves, see Schweder and Hjort [34]. In Fig. 4, the confidence intervals from the model non-robust formula (dashed curves) and the confidence

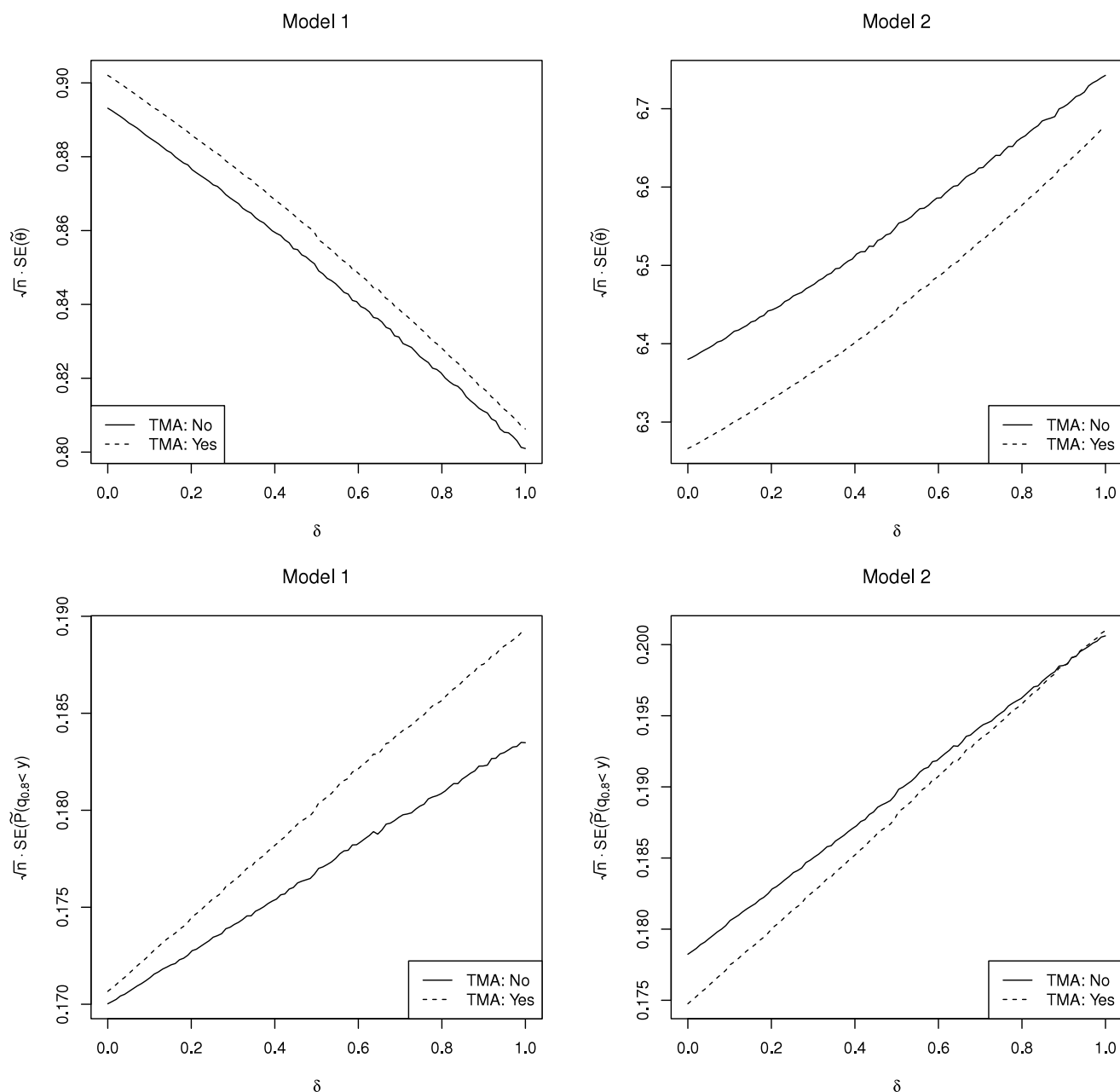


**Fig. 2.** Result of simulation 3, scenario 2. (Models 1 and 2 have the same margins, but different copulas.) TMA stands for ‘true model assumption’. The dashed lines are computed with the model non-robust asymptotic variance formula (4) and the solid lines are computed with the model robust asymptotic variance formula (3).

intervals from the model robust formula (non-dashed lines) are quite close to each other. This is partly due to the fact that the sample size ( $n = 5536$ ) is relatively large and thus scales down the asymptotic variance when a confidence interval is computed. For datasets with smaller number of observations, the choice of model (non-)robust formula will make a bigger difference.

The joint upper tail probability  $\Pr(q_{0.8} < Y)$  can be interpreted as the probability that, given that there is precipitation in all five locations, there is precipitation higher than the 0.8-quantile values in all five locations within the same day.

## 7. Comparing estimation schemes

Methods and results reached in the present paper may be exploited for examining and assessing the impact of model misspecification for different types of outcomes. Some quantities will be more affected by misspecification than others, depending on the particularities of the models entertained, the underlying data generating mechanism, and also the sample size. It is also a consequence of our results that the two-stage based estimator may have lower variance than the full ML

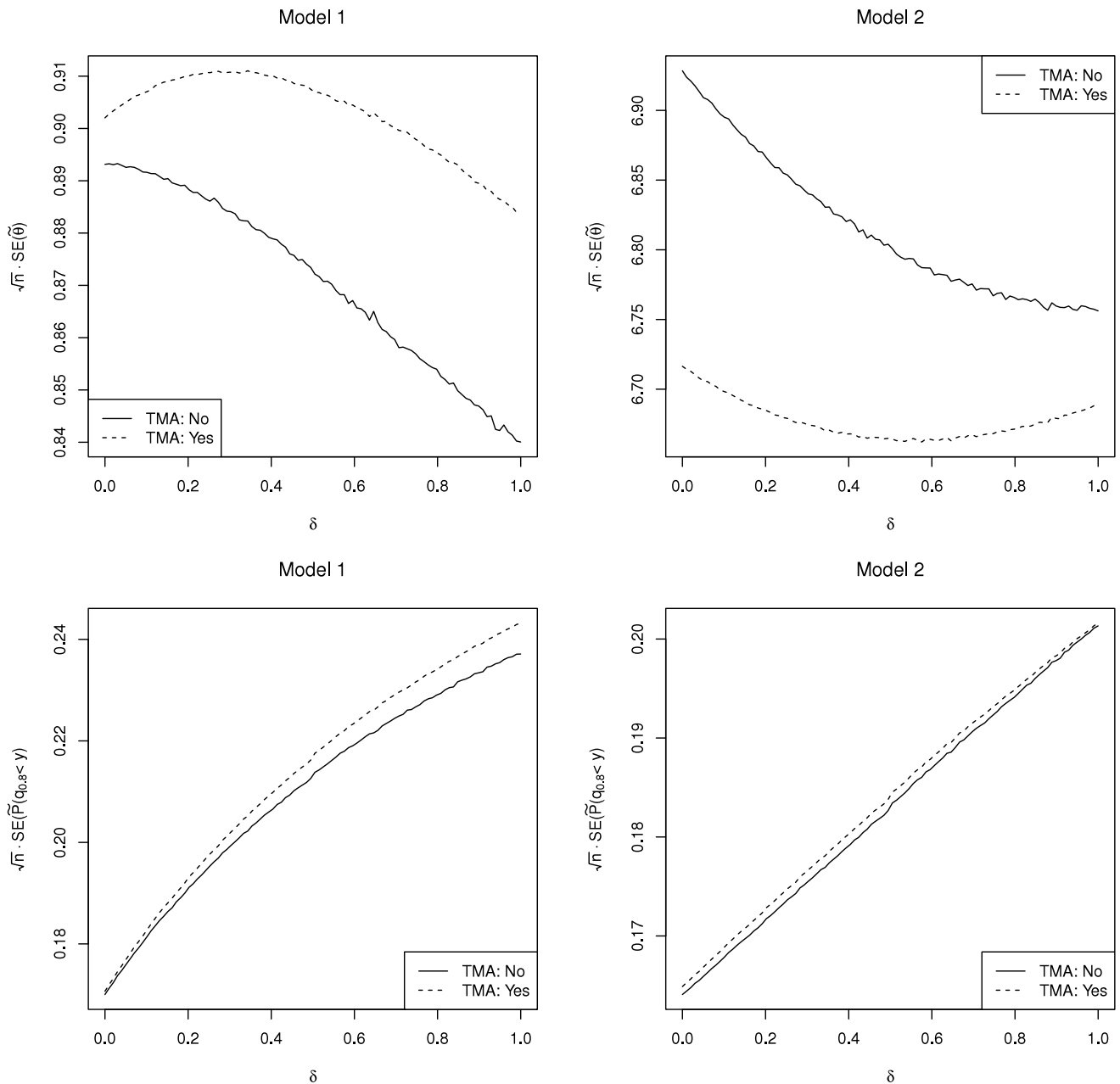


**Fig. 3.** Result of simulation 3, scenario 3. (Models 1 and 2 have different margins and different copulas.) TMA stands for ‘true model assumption’. The dashed lines are computed with the model non-robust asymptotic variance formula (4) and the solid lines are computed with the model robust asymptotic variance formula (3).

**Table 9**

Result from the model described in Table 8 fitted with the precipitation data. For the description of column labels, see Section 5.1.

| True model assumption | MLE              |                |                                      |                            |                |                                                |
|-----------------------|------------------|----------------|--------------------------------------|----------------------------|----------------|------------------------------------------------|
|                       | $\hat{\theta}$   | (95% CI)       | $\sqrt{n} \text{SE}(\hat{\theta})$   | $\hat{\Pr}(q_{0.8} < Y)$   | (95% CI)       | $\sqrt{n} \text{SE}(\hat{\Pr}(q_{0.8} < Y))$   |
| No                    | 2.351            | (2.299, 2.404) | 1.996                                | 0.108                      | (0.101, 0.114) | 0.246                                          |
| Yes                   |                  | (2.304, 2.398) | 1.796                                |                            | (0.102, 0.114) | 0.226                                          |
| True model assumption | Two-stage MLE    |                |                                      |                            |                |                                                |
|                       | $\tilde{\theta}$ | (95% CI)       | $\sqrt{n} \text{SE}(\tilde{\theta})$ | $\tilde{\Pr}(q_{0.8} < Y)$ | (95% CI)       | $\sqrt{n} \text{SE}(\tilde{\Pr}(q_{0.8} < Y))$ |
| No                    | 2.196            | (2.154, 2.238) | 1.578                                | 0.101                      | (0.096, 0.107) | 0.207                                          |
| Yes                   |                  | (2.151, 2.241) | 1.696                                |                            | (0.095, 0.107) | 0.225                                          |

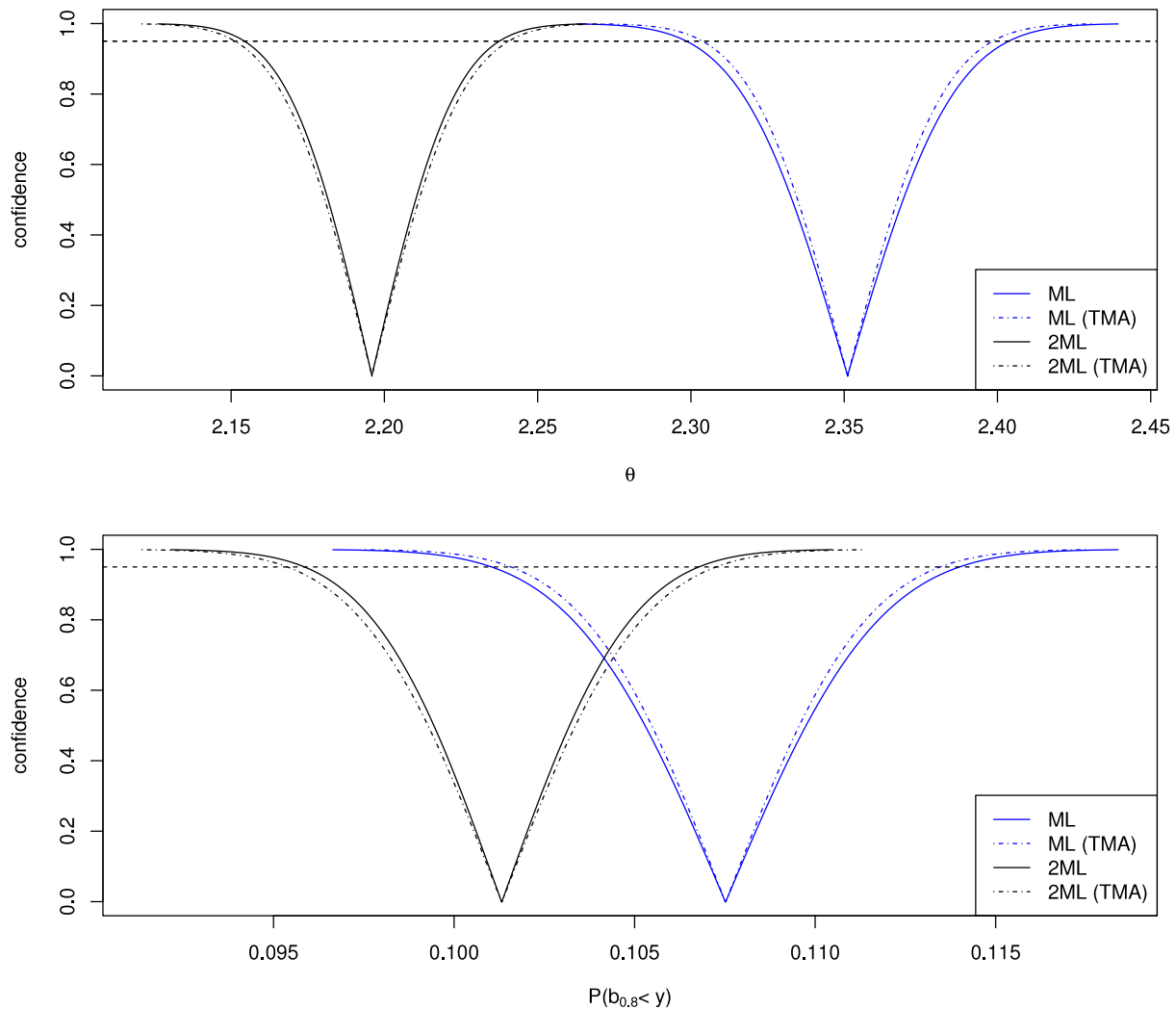


**Fig. 4.** Confidence curves for  $\theta$  and  $\Pr(q_{0.8} < Y)$ . Blue curves are from the model fitted with ML and black curves are from the model fitted with two-stage ML. Dashed curves correspond to the true model assumption. The dashed horizontal lines indicate point-wise 95% confidence. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

estimator, outside model conditions. The extent to which this is an advantage or not then depends on the implicit biases, when used to estimate well-defined population quantities.

Consider any parameter of interest, say  $\mu = \mu(G)$ , expressed in terms of the underlying data generating mechanism  $G$ , the joint cumulative distribution function for  $y = (y_1, \dots, y_d)^\top$ . When  $G$  is modeled as in (1)–(2), the parameter of interest may be expressed in terms of the parameters of that model, i.e.,  $\mu = \mu(\eta) = \mu(\alpha, \theta)$ . Under mild regularity conditions, also regarding the smoothness of  $\mu$ , methods and results obtained in Section 4 lead to

$$\sqrt{n}(\hat{\mu}_{\text{ML}} - \mu_{0,\text{ML}}) \rightsquigarrow \mathcal{N}(0, \tau_{\text{ML}}), \quad \sqrt{n}(\tilde{\mu}_{2\text{ML}} - \mu_{0,2\text{ML}}) \rightsquigarrow \mathcal{N}(0, \tau_{2\text{ML}}),$$

for the ML and the two-stage ML method, respectively. Here  $\mu_{0,\text{ML}} = \mu(\eta_{0,\text{ML}})$  and  $\mu_{0,2\text{ML}} = \mu(\eta_{0,2\text{ML}})$  are the appropriate least false parameter values for  $\mu = \mu(\eta)$ , involving the least false parameter values that the ML and the two-stage ML method are aiming for, as in Section 4. Also,  $\tau_{\text{ML}} = c_{\text{ML}}^\top V_{\eta,\text{ML}} c_{\text{ML}}$  and  $\tau_{2\text{ML}} = c_{2\text{ML}}^\top V_{\eta,2\text{ML}} c_{2\text{ML}}$ , with gradient vectors  $c = \partial \mu(\eta) / \partial \eta$  evaluated at respectively the  $\eta_{0,\text{ML}}$  and  $\eta_{0,2\text{ML}}$  positions in the  $\eta = (\alpha^\top, \theta^\top)^\top$  parameter space. There is also a third natural estimator to be reckoned with here, namely the nonparametric  $\hat{\mu}_{\text{np}} = \mu(G_n)$ , exploiting the direct empirical distribution of the observations, and for which a similar result  $\sqrt{n}(\hat{\mu}_{\text{np}} - \mu) \rightsquigarrow \mathcal{N}(0, \tau_{\text{np}})$  holds. This essentially requires the existence of an influence function for  $\mu(G)$  with a finite variance.

Importantly, these results give rise to mean squared error (MSE) approximations, for the three natural estimators of the focus parameter. We may write

$$\hat{\mu}_{\text{np}} = \mu + n^{-1/2} \tau_{\text{np}}^{1/2} Z_{n,\text{np}}, \quad \hat{\mu}_{\text{ML}} = \mu_{0,\text{ML}} + n^{-1/2} \tau_{\text{ML}}^{1/2} Z_{n,\text{ML}}, \quad \tilde{\mu}_{2\text{ML}} = \mu_{0,2\text{ML}} + n^{-1/2} \tau_{2\text{ML}}^{1/2} Z_{n,2\text{ML}},$$

with the three  $Z_n$  variables tending in distribution to the standard normal  $\mathcal{N}(0, 1)$ . These results invite the first-order large-sample approximations

$$\text{MSE}_{\text{np}} = 0^2 + \tau_{\text{np}}/n, \quad \text{MSE}_{\text{ML}} = b_{\text{ML}}^2 + \tau_{\text{ML}}/n, \quad \text{MSE}_{2\text{ML}} = b_{2\text{ML}}^2 + \tau_{2\text{ML}}/n,$$

involving in addition to the approximate variances also the biases  $b_{\text{ML}} = \mu_{0,\text{ML}} - \mu$  and  $b_{2\text{ML}} = \mu_{0,2\text{ML}} - \mu$ . The nonparametric estimator has zero asymptotic bias.

Depending on the specifics of the models studied, and the focus parameter examined, each of the three estimators might win, in the sense of having the smallest MSE. Particularly with a complicated  $\mu$ , the parametric variances might be significantly smaller than the nonparametric ones, so either the ML or the two-stage method might win, for small or moderate sample sizes, if the biases are not large. It is also clear that if the parametric model is not accurate, leading to non-zero biases  $b_{\text{ML}}$  and  $b_{2\text{ML}}$ , the nonparametric estimation method will win for sufficiently large  $n$ . What is best in practice, for different focus parameters, when working with a given dataset, becomes a matter of balancing bias with variance.

In work flowing from the present paper, the authors have pursued these ideas, by constructing asymptotically unbiased estimators of the quantities involved in the MSE expressions. Incidentally, this requires the full simultaneous limit distribution of the  $Z_n$  variables above, in that limit distributions of  $\hat{b}_{\text{ML}} = \hat{\mu}_{\text{ML}} - \hat{\mu}_{\text{np}}$  and  $\hat{b}_{2\text{ML}} = \hat{\mu}_{2\text{ML}} - \hat{\mu}_{\text{np}}$  also play a role in the construction of estimators of the squared biases involved. The end result is a focused information criterion (FIC) for copula, in the tradition of the earlier developed focused information criterion (see Claeskens and Hjort [7,8] and Jullum and Hjort [22,23]), but now involving at least three estimators for each focus parameter. If the statistician examines say  $k$  different parametric copula models, each of type (1)–(2), then  $2k + 1$  different estimators are evaluated, assessed, and ranked, via the FIC for copula. See Ko et al. [27] for details, discussion, and applications.

## 8. Conclusions and further research

In our paper we have reached precise results for the behavior of two-stage ML estimators for parametric copula models, both under and outside model conditions. This has then led to a full and flexible machinery for inference, including confidence intervals and curves for focus parameters, testing hypotheses, etc. Here we offer some concluding remarks, some pointing to further research.

The assumption that the chosen parametric model captures the true model that generated data leads to a set of simplifications in our apparatus. For instance, the model robust asymptotic variance (3) simplifies to the model non-robust asymptotic variance formula (4), which is essentially the same as the asymptotic variance formula from Joe [20]. The difference is that we choose to use the inverse of the Fisher information matrix instead of the covariance of score vectors. We believe that this is more concordant with the current practice of ML theory. Although these two definitions of asymptotic variances are theoretically the same under the true model assumption, the practical difference between them can be non-negligible when a real-life dataset is used. A further study on the impact of making one vs. the other choice for estimating the population Fisher information matrix, either in the usual fashion based on the Hessian matrix at the ML position, or using the variance matrix for the score vectors, would be useful not only for the copula community, but also more generally for statisticians applying ML theory in other contexts.

The methods in this paper require that the first and second derivatives of the density function with respect to the model parameters can be obtained. When it is not feasible to obtain the derivatives, one can consider to use the resampling methods mentioned in Joe [20,21] as alternatives.

Our simulation study shows that assuming that the model is true (i.e., using the model non-robust variance formula) can both decrease and increase the asymptotic variance of the copula parameter estimate, depending on the parametric form of the model. The degree of decrease or increase gets larger as the degree of misspecification increases.

Since the asymptotic standard deviation is divided by  $\sqrt{n}$  when computing a confidence interval, the overconfidence or underconfidence coming from the true model assumption gets larger as sample size decreases. When carrying out hypothesis testing for the copula parameter, or on a function that takes the copula parameter as an input, the choice of model-robust variance formula or not can therefore lead to different decisions.

Joe [20] compares asymptotic relative efficiency (ARE) of the two-stage ML estimator with that of the full ML and concludes that two-stage ML method typically has good ARE. We could observe that the two-stage ML estimator is still highly efficient when the true model assumption is dropped and the model robust asymptotic variance formulas are used.

When models are highly misspecified, however, we see that the asymptotic variance of two-stage ML estimators can be smaller than that of the ML estimator. This happens both when the model-robust and model non-robust variance formulas are used. This effect increases as the degree of model misspecification and dimension increases. This relates to the fact that the Hájek–Le Cam convolution theorem, along with theorems of a similar nature for the optimality of ML estimation for parametric models, does not apply when the models are misspecified.

When the true model assumption decreases the variance of the estimator of the copula parameter, we sometimes observe that the asymptotic variance of the joint upper tail probability increases. A possible reason is that this asymptotic variance is computed with the delta method by using the whole variance matrix of  $\tilde{\eta}$  (or  $\hat{\eta}$ ), including the covariances between copula and margin parameters. A further theoretical and numerical study on the property of variance transformation through the delta method would be fruitful, especially considering the fact that this also happens in various other types of models.

Our general results concerning misspecified models involve both the notion of least false parameters (those aimed at by respectively the ML and the two-stage ML methods) and ensuing limit variance matrices. It appears possible to study how much these vary from their true counterparts, under model conditions, in terms of functional derivatives, in certain types of model neighborhoods. The  $O(1/\sqrt{n})$  model misspecification from Chapters 6 and 7 of Claeskens and Hjort [8] can be used for such purposes. Such considerations could then be used to pinpoint which parameter estimands  $\mu = \mu(\eta)$  are more robust

than others, with respect to moderate misspecification. It is outside the scope of the present paper to pursue this issue, however. We also consider the development of Section 7, with concrete analysis for each focus parameter  $\mu = \mu(\eta)$ , to be more directly informative in this regard.

The authors of this study are currently developing model selection criteria for two-stage ML estimators that utilize the model robust large-sample distribution from Section 3 of this article [25]. These model selection criteria will complement earlier model selection methodology efforts, such as the Copula Information Criterion (CIC) developed by Grønneberg and Hjort [14]. One particular goal of these extended efforts will be to aid model building and selection for pair-copula constructions [1,2].

The methodology for two-stage ML estimation for copula models, inside and outside model conditions, has been developed in this paper, primarily aiming for the case of iid sequences of vector observations. Importantly, in the presence of relevant covariate information, the large-sample results and ensuing inference methods can be extended to various classes of conditional copula regression models, with the required extra efforts. This will then lead to further estimation and inference tools for models worked in, e.g., [3,37]. The covariates may influence the margins, the copula mechanism, or both. As one such example, we might fit the Norwegian precipitation data of Section 6 using the Gumbel copula, but now with

$$\theta_i = \theta \exp\{\gamma_1(x_i - 1990) + \gamma_2(x_i - 1990)^2\},$$

where  $x_i$  is the calendar year for observation  $i \in \{1, \dots, n\}$ . This leads to estimates and confidence curves for  $\gamma_1$ ,  $\gamma_2$  and related parameters, via the two-stage construction

$$\ell_c(\tilde{\alpha}, \theta, \gamma) = \sum_{i=1}^n \ln c\{F_1(y_{i,1}, \tilde{\alpha}_1), \dots, F_d(y_{i,d}, \tilde{\alpha}_d), \theta_i\},$$

and allows one the opportunity to assess any changes of the copula mechanism over time.

## Acknowledgments

The authors would like to thank Ingrid Hobæk Haff for her valuable comments and fruitful discussions. They are also grateful to two anonymous referees and the Editor-in-Chief, Christian Genest, for their careful reading of and constructive comments on the first version of our article. In addition, the authors acknowledge partial funding from the Norwegian Research Council supported research group FocuStat: Focus Driven Statistical Inference With Complex Data, and from the Department of Mathematics at the University of Oslo, Norway.

## Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.jmva.2019.01.004>.

## References

- [1] K. Aas, D. Berg, Models for construction of multivariate dependence: A comparison study, *Eur. J. Finance* 15 (2009) 639–659.
- [2] K. Aas, C. Czado, A. Frigessi, H. Bakken, Pair-copula constructions of multiple dependence, *Insurance Math. Econom.* 44 (2009) 182–195.
- [3] E.F. Acar, R.V. Craiu, F. Yao, Statistical testing of covariate effects in conditional copula models, *Electron. J. Statist.* 7 (2013) 2822–2850.
- [4] E.W. Andersen, Composite likelihood and two-stage estimation in family studies, *Biostatistics* 5 (2004) 15–30.
- [5] E.W. Andersen, Two-stage estimation in copula models used in family studies, *Lifetime Data Anal.* 11 (2005) 333–350.
- [6] G. Casella, R.L. Berger, *Statistical Inference*, Duxbury Pacific Grove, CA, 2002.
- [7] G. Claeskens, N.L. Hjort, The focused information criterion [with discussion contributions and a rejoinder], *J. Amer. Statist. Assoc.* 98 (2003) 900–916.
- [8] G. Claeskens, N.L. Hjort, *Model Selection and Model Averaging*, Cambridge University Press, Cambridge, 2008.
- [9] S.G. Coles, J. Heffernan, J.A. Tawn, Dependence measures for extreme value analyses, *Extremes* 2 (1999) 339–365.
- [10] P. Embrechts, Copulas: A personal view, *J. Risk Insur.* 76 (2009) 639–650.
- [11] C. Genest, A.-C. Favre, Everything you always wanted to know about copula modeling but were afraid to ask, *J. Hydrol. Eng.* 12 (2007) 347–368.
- [12] C. Genest, J.G. Nešlehová, L.-P. Rivest, The class of multivariate max-id copulas with  $L_1$ -norm symmetric exponent measure, *Bernoulli J.* 24 (2018) 3751–3790.
- [13] C. Genest, L.-P. Rivest, Statistical inference procedures for bivariate Archimedean copulas, *J. Amer. Statist. Assoc.* 88 (1993) 1034–1043.
- [14] S. Grønneberg, N.L. Hjort, The copula information criteria, *Scand. J. Stat.* 41 (2014) 436–459.
- [15] J. Hájek, A characterization of limiting distributions of regular estimates, *Probab. Theory Related Fields* 14 (1970) 323–330.
- [16] A.R. Hall, *Generalized Method of Moments*, Oxford University Press, Oxford, 2005.
- [17] J.W. Hardin, The sandwich estimate of variance, in: T.B. Fomby, R.C. Hill (Eds.), *Maximum Likelihood Estimation of Misspecified Models: Twenty Years Later*, Emerald Group Publishing Limited, 2003, pp. 45–73.
- [18] I. Hobæk Haff, Parameter estimation for pair-copula constructions, *Bernoulli* 19 (2013) 462–491.
- [19] H. Joe, *Multivariate Models and Multivariate Dependence Concepts*, Chapman & Hall, London, 1997.
- [20] H. Joe, Asymptotic efficiency of the two-stage estimation method for copula-based models, *J. Multivariate Anal.* 94 (2005) 401–419.
- [21] H. Joe, *Dependence Modeling with Copulas*, CRC Press, Boca Raton, FL, 2014.
- [22] M. Jullum, N.L. Hjort, Parametric or nonparametric: The FIC approach, *Statist. Sinica* 27 (2017) 951–981.
- [23] M. Jullum, N.L. Hjort, What price semiparametric Cox regression? *Lifetime Data Anal.* (2018) <http://dx.doi.org/10.1007/s10985-018-9450-7>, (in press).
- [24] G. Kim, M.J. Silvapulle, P. Silvapulle, Comparison of semiparametric and parametric methods for estimating copulas, *Comput. Statist. Data Anal.* 51 (2007) 2836–2850.
- [25] V. Ko, N.L. Hjort, Copula information criterion for model selection with two-stage maximum likelihood estimation, *Econometr. Statist.* (2019) (in press).

- [26] V. Ko, N.L. Hjort, Supplement to “Model robust inference with two-stage maximum likelihood estimation for copulas”, *J. Multivariate Anal.* (2019).
- [27] V. Ko, N.L. Hjort, I. Hobæk Haff, Focused information criterion for copulas, *Scand. J. Stat.* (2019) in press.
- [28] S. Kullback, R.A. Leibler, On information and sufficiency, *Ann. Math. Stat.* 22 (1951) 79–86.
- [29] L. Le Cam, Maximum likelihood: An introduction, *Internat. Statist. Rev.* 58 (1990) 153–171.
- [30] E.L. Lehmann, *Elements of Large-Sample Theory*, Springer Science & Business Media, 2004.
- [31] A.J. McNeil, J. Nešlehová, Multivariate Archimedean copulas,  $d$ -monotone functions and  $\ell_1$ -norm symmetric distributions, *Ann. Statist.* 37 (2009) 3059–3097.
- [32] R.B. Nelsen, *An Introduction to Copulas*, second ed., Springer, New York, 2006.
- [33] A.K. Nikoloulopoulos, H. Joe, H. Li, Vine copulas with asymmetric tail dependence and applications to financial return data, *Comput. Statist. Data Anal.* 56 (2012) 3659–3673.
- [34] T. Schweder, N.L. Hjort, *Confidence, Likelihood, Probability: Inference With Confidence Distributions*, Cambridge University Press, Cambridge, 2016.
- [35] J.H. Shih, T.A. Louis, Inferences on the association parameter in copula models for bivariate survival data, *Biometrics* 51 (1995) 1384–1399.
- [36] A. Sklar, Fonctions de répartition à  $n$  dimensions et leurs marges, *Publ. Inst. Statist. Univ. Paris* 8 (1959) 229–231.
- [37] N. Veraverbeke, M. Omelka, I. Gijbels, Estimation of a conditional copula and association measures, *Scand. J. Stat.* 38 (2011) 766–780.
- [38] H. White, Maximum likelihood estimation of misspecified models, *Econometrica* 50 (1982) 1–25.
- [39] J.J. Xu, *Statistical Modelling and Inference for Multivariate and Longitudinal Discrete Response Data* (Ph.D. thesis), University of British Columbia, Vancouver, BC, Canada, 1996.

# Supplementary Material: Model robust inference with two-stage maximum likelihood estimation for copulas

Vinnie Ko<sup>a,\*</sup>, Nils Lid Hjort<sup>a</sup>

<sup>a</sup>*Department of Mathematics, University of Oslo  
PB 1053, Blindern, NO-0316 Oslo, Norway*

---

## Abstract

This is a supplement to the article Ko and Hjort [2]. Section 1 provides proofs of the key results from that paper, and Section 2 provides figures referred to in its Section 6.

---

## 1. Proofs of main paper results

### 1.1. Proof of Lemma 2

*Proof.* For the estimator  $\tilde{\alpha}$  from stage 1, we have  $\tilde{\alpha} \xrightarrow{p} \alpha_0$ . The Taylor series expansion of  $M_n(\tilde{\alpha}, \theta)$  around  $\alpha_0$ , with  $\theta$  fixed as an arbitrary constant such that  $\theta \in \Theta$ , gives

$$M_n(\tilde{\alpha}, \theta) = M_n(\alpha_0, \theta) + \frac{\partial M_n(\alpha_0, \theta)}{\partial \alpha} (\tilde{\alpha} - \alpha_0) + o_p(n^{-1/2}).$$

Since  $\tilde{\alpha} \xrightarrow{p} \alpha_0$ , we have, for every  $\varepsilon > 0$  and all  $\theta \in \Theta$ , that

$$|M_n(\tilde{\alpha}, \theta) - M_n(\alpha_0, \theta)| < \varepsilon. \quad (1)$$

Le Cam's uniform convergence theorem (Ferguson [1, Theorem 16(a)]) gives directly:

$$\Pr \left\{ \limsup_{n \rightarrow \infty} \sup_{\theta \in \Theta} |M_n(\alpha_0, \theta) - M(\alpha_0, \theta)| = 0 \right\} = 1 \quad (2)$$

By combining (1) and (2), we have

$$\sup_{\theta \in \Theta} |M_n(\tilde{\alpha}, \theta) - M(\alpha_0, \theta)| \xrightarrow{p} 0. \quad (3)$$

In other words,  $M_n(\tilde{\alpha}, \theta)$  converges uniformly in probability to  $M(\alpha_0, \theta)$ .

Now, we mimic the proof of Theorem 5.7 in van der Vaart [4]. Consider the estimator  $\tilde{\theta}_n$  that nearly maximizes  $M_n(\tilde{\alpha}, \theta)$  and satisfies

$$M_n(\tilde{\alpha}, \tilde{\theta}_n) \geq \sup_{\theta} M_n(\tilde{\alpha}, \theta) - o_p(1).$$

---

\*Corresponding author. Email address: vinniebk@math.uio.no

Then we have certainly  $M_n(\tilde{\alpha}, \tilde{\theta}_n) \geq M_n(\tilde{\alpha}, \theta_0) - o_p(1)$ . By adding  $-M(\alpha_0, \tilde{\theta}_n) + o_p(1)$  on both sides, we get

$$\begin{aligned} M_n(\tilde{\alpha}, \theta_0) - M(\alpha_0, \tilde{\theta}_n) &\leq M_n(\tilde{\alpha}, \tilde{\theta}_n) - M(\alpha_0, \tilde{\theta}_n) + o_p(1) \\ &\leq \sup_{\theta} |M_n(\tilde{\alpha}, \theta) - M(\alpha_0, \theta)| + o_p(1) \xrightarrow{P} 0 \end{aligned}$$

by (3).

Because of the assumption that is made in the beginning of the lemma, for every  $\varepsilon > 0$ , there exists  $\eta > 0$  such that  $M(\alpha_0, \theta) < M(\alpha_0, \theta_0) - \eta$  for every  $\theta$  satisfying  $d(\theta, \theta_0) \geq \varepsilon$ . This implies that the event  $\{d(\theta, \theta_0) \geq \varepsilon\}$  is contained in the event  $\{M(\alpha_0, \theta) < M(\alpha_0, \theta_0) - \eta\}$ . The probability of the latter event converges to 0. Thus,  $d(\tilde{\theta}_n, \theta_0)$  tends to zero in probability.  $\square$

*Proof of Proposition 1*

*Proof.*

**Lemma 1.** *Let  $\tilde{\theta}$  be the two-stage ML estimator of  $\theta$  and  $\theta_0$  be the least false value of this parameter. Then we have*

$$\tilde{\theta} - \theta_0 = I_{\theta}^{-1} \frac{1}{n} \sum_{i=1}^n U_{\theta}(y_i, \tilde{\alpha}, \theta_0) + o_p(n^{-1/2}).$$

*Proof.* The Taylor series expansion of  $U_{n,\theta}(\tilde{\alpha}, \tilde{\theta})$  around  $\theta_0$  yields

$$0 = U_{n,\theta}(\tilde{\alpha}, \tilde{\theta}) = U_{n,\theta}(\tilde{\alpha}, \theta_0) + H_{n,\theta}(\tilde{\alpha}, \theta_0)(\tilde{\theta} - \theta_0) + o_p(n^{-1/2}),$$

which gives

$$\tilde{\theta} - \theta_0 = -H_{n,\theta}(\tilde{\alpha}, \theta_0)^{-1} U_{n,\theta}(\tilde{\alpha}, \theta_0) + o_p(n^{-1/2}). \quad (4)$$

From stage 1,  $\tilde{\alpha}$  is a consistent estimator of  $\alpha_0$ , i.e.,  $\tilde{\alpha} \xrightarrow{P} \alpha_0$ . Since  $H_{\theta}(y, \alpha, \theta_0)$  is differentiable at  $\alpha_0$  and there exists a Jacobian  $\dot{H}_{\theta}(y, \alpha_0, \theta_0)$ , we can apply the multivariate delta method (Lehmann [3, Theorem 3.7]) and obtain  $H_{n,\theta}(\tilde{\alpha}, \theta_0) \xrightarrow{P} H_{n,\theta}(\alpha_0, \theta_0)$ . Further, by the law of large numbers, we have

$$-H_{n,\theta}(\alpha_0, \theta_0) \xrightarrow{P} -E_G [H_{\theta}(y, \alpha_0, \theta_0)] = I_{\theta}.$$

Thus, we have  $-H_{n,\theta}(\tilde{\alpha}, \theta_0) \xrightarrow{P} I_{\theta}$  and (4) becomes

$$\tilde{\theta} - \theta_0 = I_{\theta}^{-1} U_{n,\theta}(\tilde{\alpha}, \theta_0) + o_p(n^{-1/2}).$$

$\square$

**Lemma 2.** *With notation as in the previous lemma, we have*

$$\begin{aligned} \tilde{\theta} - \theta_0 &= I_{\theta}^{-1} \left\{ U_{n,\theta}(\alpha_0, \theta_0) - I_{\alpha,\theta}^{\top} (\tilde{\alpha} - \alpha_0) \right\} + o_p(n^{-1/2}) \\ &= I_{\theta}^{-1} \left\{ U_{n,\theta}(\alpha_0, \theta_0) - I_{\alpha,\theta}^{\top} I_{\alpha}^{-1} U_{n,\alpha}(\alpha_0) \right\} + o_p(n^{-1/2}). \end{aligned}$$

*Proof.* The Taylor series expansion of  $U_{n,\theta}(\tilde{\alpha}, \theta_0)$  around  $\alpha_0$  yields

$$\begin{aligned}
U_{n,\theta}(\tilde{\alpha}, \theta_0) &= U_{n,\theta}(\alpha_0, \theta_0) + \left( \frac{\partial U_{n,\theta}(\alpha_0, \theta_0)}{\partial \alpha_0^\top} \right)^\top (\tilde{\alpha} - \alpha_0) + o_p(n^{-1/2}) \\
&= U_{n,\theta}(\alpha_0, \theta_0) + H_{n,\alpha,\theta}^\top(\alpha_0, \theta_0) (\tilde{\alpha} - \alpha_0) + o_p(n^{-1/2}) \\
&= U_{n,\theta}(\alpha_0, \theta_0) - \mathcal{I}_{\alpha,\theta}^\top (\tilde{\alpha} - \alpha_0) + o_p(n^{-1/2}) \\
&= U_{n,\theta}(\alpha_0, \theta_0) - \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} U_{n,\alpha}(\alpha_0) + o_p(n^{-1/2}).
\end{aligned}$$

Plugging this result into Lemma 1 gives

$$\begin{aligned}
\tilde{\theta} - \theta_0 &= \mathcal{I}_\theta^{-1} \left\{ U_{n,\theta}(\alpha_0, \theta_0) - \mathcal{I}_{\alpha,\theta}^\top (\tilde{\alpha} - \alpha_0) \right\} + o_p(n^{-1/2}) \\
&= \mathcal{I}_\theta^{-1} \left\{ U_{n,\theta}(\alpha_0, \theta_0) - \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} U_{n,\alpha}(\alpha_0) \right\} + o_p(n^{-1/2}).
\end{aligned}$$

□

Finally, we obtain Proposition 1 by combining the result from Lemma 2 from this supplementary material with the result from Lemma 1 from [2] by using Slutsky's theorem:

$$\begin{aligned}
\sqrt{n}(\tilde{\eta} - \eta_0) &= \sqrt{n} \left( \frac{\tilde{\alpha} - \alpha_0}{\tilde{\theta} - \theta_0} \right) \\
&= \sqrt{n} \left( \frac{\mathcal{I}_\alpha^{-1} U_{n,\alpha}(\alpha_0) + o_p(n^{-1/2})}{\mathcal{I}_\theta^{-1} (U_{n,\theta}(\alpha_0, \theta_0) - \mathcal{I}_{\alpha,\theta}^\top \mathcal{I}_\alpha^{-1} U_{n,\alpha}(\alpha_0)) + o_p(n^{-1/2})} \right) \\
&= \sqrt{n} \left( \frac{\mathcal{I}_\alpha}{\mathcal{I}_{\alpha,\theta}^\top} \quad 0 \right)^{-1} \begin{pmatrix} U_{n,\alpha}(\alpha_0) \\ U_{n,\theta}(\alpha_0, \theta_0) \end{pmatrix} + \begin{pmatrix} o_p(1) \\ o_p(1) \end{pmatrix} \\
&\xrightarrow{d} \begin{pmatrix} \mathcal{I}_\alpha & 0 \\ \mathcal{I}_{\alpha,\theta}^\top & \mathcal{I}_\theta \end{pmatrix}^{-1} \begin{pmatrix} \Lambda_\alpha \\ \Lambda_\theta \end{pmatrix} \\
&= \mathcal{I}_\eta^{-1} \Lambda_\eta \sim \mathcal{N} \left( 0, \mathcal{I}_\eta^{-1} K_\eta (\mathcal{I}_\eta^{-1})^\top \right).
\end{aligned}$$

□

### *Proof of Lemma 3*

*Proof.* We assume that margins and copula are correctly specified, i.e.,  $f(\cdot, \eta_0) = g$ , or simply  $g = f$  in concise notation. Then

$$\begin{aligned}
K_{\alpha_1,\theta} &= \text{Cov}_G(U_{\alpha_1}(y_1, \alpha_{0,1}), U_\theta(y, \alpha_0, \theta_0)) \\
&= \mathbb{E}_G \left[ U_{\alpha_1}(y_1, \alpha_{0,1}) U_\theta(y, \alpha_0, \theta_0)^\top \right] \\
&= \int_{y_1} \cdots \int_{y_d} g U_{\alpha_1}(y_1, \alpha_{0,1}) U_\theta(y, \alpha_0, \theta_0)^\top dy_1 \cdots dy_d \\
&= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \left( \int_{y_2} \cdots \int_{y_d} g U_\theta(y, \alpha_0, \theta_0)^\top dy_2 \cdots dy_d \right) dy_1 \\
&= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \left( \int_{y_2} \cdots \int_{y_d} g \left( \frac{\partial \ln c(F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d}), \theta_0)}{\partial \theta_0} \right)^\top dy_2 \cdots dy_d \right) dy_1 \\
&= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \left( \int_{y_2} \cdots \int_{y_d} g \left( \frac{\partial \ln f(y, \alpha_{0,1}, \theta_0)}{\partial \theta_0} \right)^\top dy_2 \cdots dy_d \right) dy_1 \\
&= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \left( \int_{y_2} \cdots \int_{y_d} f \left( \frac{\partial \ln f(y, \alpha_{0,1}, \theta_0)}{\partial \theta_0} \right)^\top dy_2 \cdots dy_d \right) dy_1.
\end{aligned}$$

By using that  $\partial \ln f / \partial \theta_0 = (1/f) \partial f / \partial \theta_0$ , this becomes

$$\begin{aligned}
K_{\alpha_1, \theta} &= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \left( \int_{y_2} \cdots \int_{y_d} \left( \frac{\partial f(y, \alpha_{0,1}, \theta_0)}{\partial \theta_0} \right)^\top dy_2 \cdots dy_d \right) dy_1 \\
&= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \left( \frac{\partial}{\partial \theta_0} \int_{y_2} \cdots \int_{y_d} f dy_2 \cdots dy_d \right)^\top dy_1 \\
&= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \left( \frac{\partial}{\partial \theta_0} f_1 \right)^\top dy_1 \\
&= \int_{y_1} U_{\alpha_1}(y_1, \alpha_{0,1}) \cdot 0 dy_1 \\
&= 0.
\end{aligned}$$

Similarly,  $K_{\alpha_j, \theta} = 0$  for all  $j = 1, \dots, d$ . Thus,  $K_{\alpha, \theta} = 0$ . □

#### *Proof of Lemma 4*

*Proof.* We assume again that margins and copula are correctly specified, i.e.,  $g = f(\cdot, \eta_0)$ , or  $g = g$ , and have

$$\begin{aligned}
E_G [U_\theta(y, \alpha_0, \theta_0)] &= \int g \frac{\partial \ln c}{\partial \theta_0} dy \\
&= \int f \frac{\partial \ln c}{\partial \theta_0} dy \\
&= \int f \frac{1}{c} \frac{\partial c}{\partial \theta_0} dy \\
&= \int \left( \prod_{j=1}^d f_j \right) \frac{\partial c}{\partial \theta_0} dy \\
&= \int \frac{\partial c \prod_{j=1}^d f_j}{\partial \theta_0} dy \\
&= \int \frac{\partial f}{\partial \theta_0} dy \\
&= \frac{\partial}{\partial \theta_0} \int f dy,
\end{aligned}$$

which is zero. Also,

$$\begin{aligned}
0 &= \frac{\partial \mathbb{E}_G [U_\theta(y, \alpha_0, \theta_0)]^\top}{\partial \alpha_0} \\
&= \frac{\partial}{\partial \alpha_0} \int f \frac{\partial \ln c}{\partial \theta_0^\top} dy \\
&= \int \left( f \frac{\partial^2 \ln c}{\partial \alpha_0 \partial \theta_0^\top} + \frac{\partial f}{\partial \alpha_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right) dy \\
&= \int \left( f \frac{\partial^2 \ln c}{\partial \alpha_0 \partial \theta_0^\top} + f \frac{\partial \ln f}{\partial \alpha_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right) dy \\
&= \int \left\{ f \frac{\partial^2 \ln c}{\partial \alpha_0 \partial \theta_0^\top} + f \left( \frac{\partial \sum_{j=1}^d \ln f_j}{\partial \alpha_0} + \frac{\partial \ln c}{\partial \alpha_0} \right) \frac{\partial \ln c}{\partial \theta_0^\top} \right\} dy \\
&= -\mathbb{E}_G \left[ -\frac{\partial^2 \ln c}{\partial \alpha_0 \partial \theta_0^\top} \right] + \mathbb{E}_G \left[ \frac{\partial \sum_{j=1}^d \ln f_j}{\partial \alpha_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right] + \mathbb{E}_G \left[ \frac{\partial \ln c}{\partial \alpha_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right] \\
&= -\mathbb{E}_G [-H_{\alpha, \theta}(y, \alpha_0, \theta_0)] + \mathbb{E}_G [U_\alpha(y, \alpha_0) U_\theta(y, \alpha_0, \theta_0)^\top] + \mathbb{E}_G [U_\alpha^*(y, \alpha_0) U_\theta(y, \alpha_0, \theta_0)^\top] \\
&= -\mathcal{I}_{\alpha, \theta} + K_{\alpha, \theta} + K_{\alpha, \theta}^* \\
&= -\mathcal{I}_{\alpha, \theta} + K_{\alpha, \theta}^*.
\end{aligned}$$

So,  $\mathcal{I}_{\alpha, \theta} = K_{\alpha, \theta}^*$ . □

*Proof of Lemma 5*

*Proof.* We assume that margins and copula are correctly specified, i.e.,  $g = f(\cdot, \eta_0)$ . Then

$$\begin{aligned}
0 &= \frac{\partial \mathbb{E}_G [U_\theta(y, \alpha_0, \theta_0)]^\top}{\partial \theta_0} \\
&= \frac{\partial}{\partial \theta_0} \int f \frac{\partial \ln c}{\partial \theta_0^\top} dy \\
&= \int f \left( \frac{\partial^2 \ln c}{\partial \theta_0 \partial \theta_0^\top} + \frac{\partial f}{\partial \theta_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right) dy \\
&= \int \left( f \frac{\partial^2 \ln c}{\partial \theta_0 \partial \theta_0^\top} + f \frac{\partial \ln f}{\partial \theta_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right) dy \\
&= \int \left( f \frac{\partial^2 \ln c}{\partial \theta_0 \partial \theta_0^\top} + f \frac{\partial \ln c}{\partial \theta_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right) dy \\
&= -\mathbb{E}_G \left[ -\frac{\partial^2 \ln c}{\partial \theta_0 \partial \theta_0^\top} \right] + \mathbb{E}_G \left[ \frac{\partial \ln c}{\partial \theta_0} \frac{\partial \ln c}{\partial \theta_0^\top} \right] \\
&= -\mathbb{E}_G [-H_\theta(y, \alpha_0, \theta_0)] + \mathbb{E}_G [U_\theta(y, \alpha_0, \theta_0) U_\theta(y, \alpha_0, \theta_0)^\top] \\
&= -\mathcal{I}_\theta + K_\theta.
\end{aligned}$$

So,  $\mathcal{I}_\theta = K_\theta$ . □

## 2. Figures from Section 6

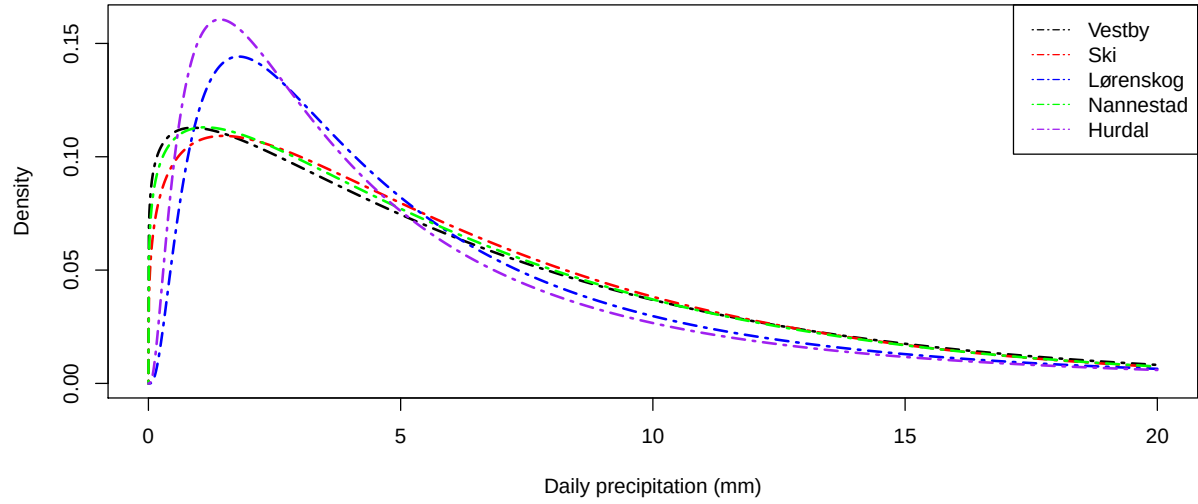


Figure 1: Density plot of fitted marginal distributions of precipitation data.

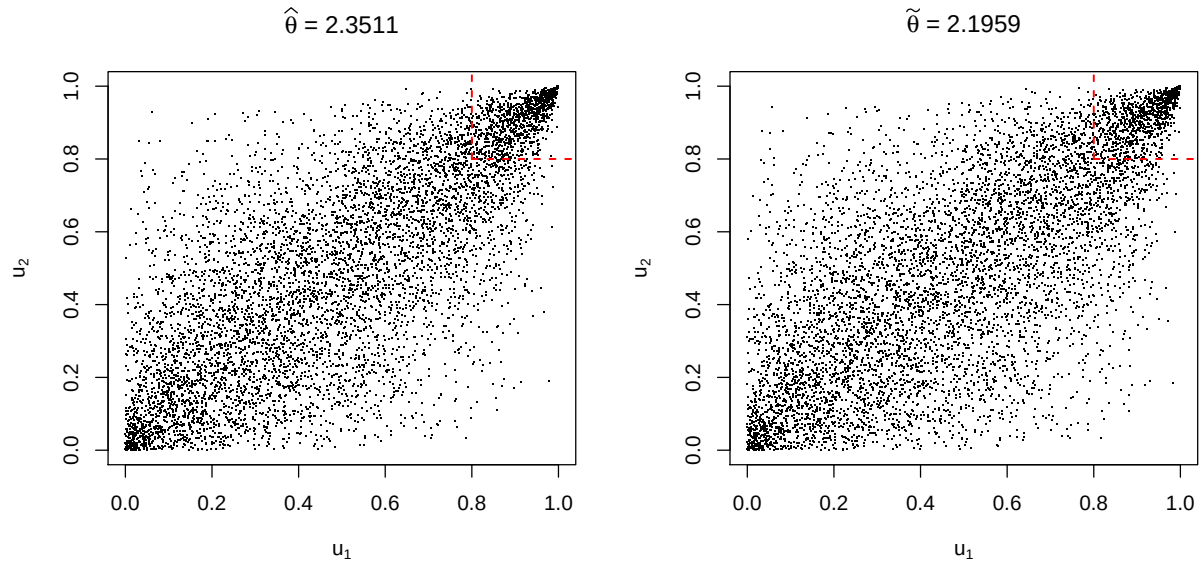


Figure 2: Scatter plots between the first 2 variables of 4-dimensional Gumbel copula. The plots are generated by simulating from the fitted models with precipitation data. The left side is fitted with ML and the right side fitted with two-stage ML. The areas in the upper corners indicated by the red dashed lines are the regions where both variables have higher values than their 0.8-quantiles.

## References

- [1] T. S. Ferguson, A Course in Large Sample Theory, Chapman & Hall London, 1996.
- [2] V. Ko, N. L. Hjort, Model robust inference with two-stage maximum likelihood estimation for copulas, Journal of Multivariate Analysis (2019). (in press).
- [3] E. L. Lehmann, Elements of Large-Sample Theory, Springer Science & Business Media, 2004.
- [4] A. W. van der Vaart, Asymptotic Statistics, Cambridge University Press, 2000.









Contents lists available at ScienceDirect

## Econometrics and Statistics

journal homepage: [www.elsevier.com/locate/ecosta](http://www.elsevier.com/locate/ecosta)

# Copula information criterion for model selection with two-stage maximum likelihood estimation

Vinnie Ko\*, Nils Lid Hjort

Department of Mathematics, University of Oslo, Blindern, NO-0316, Oslo PB 1053, Norway

## ARTICLE INFO

### Article history:

Received 26 February 2018

Revised 22 January 2019

Accepted 22 January 2019

Available online xxx

### Keywords:

Akaike information criterion

Copula

Copula information criterion

Inference functions for margins

Model robust

Two-stage maximum likelihood

## ABSTRACT

In parametric copula setups, where both the margins and copula have parametric forms, two-stage maximum likelihood estimation, often referred to as inference functions for margins, is used as an attractive alternative to the full maximum likelihood estimation strategy. Exploiting the existing model robust inference of two-stage maximum likelihood estimation, a copula information criterion (CIC) for model selection is developed. In a nutshell, CIC aims for the model that minimizes the Kullback–Leibler divergence from the real data generating mechanism. CIC does not assume that the chosen parametric model captures this true model, unlike what is assumed for AIC. In this sense, CIC is analogous to the Takeuchi Information Criterion (TIC), which is defined for the full maximum likelihood. If the additional assumption that a candidate model is correctly specified is made, then CIC for that model simplifies to AIC. Additionally, CIC can easily be extended to the conditional copula setup where covariates are parametrically linked to the copula model. As a numerical illustration, simulation studies were performed to show that the better model according to CIC also has better prediction performance in general. The result also shows that the bias correction term of CIC penalizes the misspecified model more heavily. This bias correction term has a strong positive relationship with the prediction performance of the model. So, a model with bad prediction performance is being penalized more by CIC. Although this behavior of the bias correction part is an important conceptual advance of CIC, this is not sufficient to make CIC outperform AIC in practice. This is because each misspecified model has the bias correction term and they grow at different speeds, depending on the model. The difference between CIC and AIC becomes minimal as sample size grows because the log-likelihood part outgrows the bias correction part.

© 2019 EcoSta Econometrics and Statistics. Published by Elsevier B.V. All rights reserved.

## 1. Introduction and copula models

One of the main practical issues in copula modeling is model selection. In the full parametric setup, where both the copula and margins are assumed to have a parametric form, one often has multiple candidates for both the copula and margins. As the dimension of the model increases, a list of possible combinations of margins and copula grows rapidly. Hence, there is a need for a model selection criterion that can evaluate each model systematically according to a certain philosophy or criteria and assign a score to each model. In the end, one would choose the model with the best score.

\* Corresponding author.

E-mail address: [vinniebk@math.uio.no](mailto:vinniebk@math.uio.no) (V. Ko).

<https://doi.org/10.1016/j.ecosta.2019.01.001>

2452-3062/© 2019 EcoSta Econometrics and Statistics. Published by Elsevier B.V. All rights reserved.

Throughout this paper, we consider the full parametric setup. In this setup, one can simultaneously estimate all parameters of the model (i.e. both copula parameters and margin parameters) by using maximum likelihood (ML) estimation. In this ML estimation framework, one can for instance use  $AIC_{ML}$  (Akaike, 1974) or TIC (also known as model-robust  $AIC_{ML}$ ) (Takeuchi, 1976) as model selection criterion and select the model with the best score. (Note that we denote the AIC under ML estimation as  $AIC_{ML}$  to distinguish it from the two-stage ML based  $AIC_{2ML}$ , which we will derive in Section 2.3.) However, when the dimension of the copula model gets high, the number of parameters increases quickly and the ML estimation is not always feasible in terms of speed and numerical stability. Two-stage maximum likelihood (two-stage ML) estimation, also often referred to as inference functions for margins (IFM), is a popular alternative estimation strategy that is designed to overcome these drawbacks of the ML estimation. In stage 1 of the two-stage ML estimation, the parameter vectors of each marginal distribution are estimated separately by ML. In stage 2, the estimates from stage 1 are plugged into the log-likelihood of the model. Then, the parameters of the copula, which are now the remaining parameters to estimate, are estimated by using ML estimation again. One of the advantages of this multi-stage approach is that it is computation-wise much faster than estimating all parameters simultaneously, because it does not have to search for the global maximum in high-dimensional space. A drawback of the two-stage ML estimation method, however, is that we cannot use the classical results based on ML estimation, which include model selection criteria such as TIC and BIC.

In practice, different sorts of goodness-of-fit testing are often used as substitutes, to choose the best model (Genest and Favre, 2007). Another often used model selection strategy for the two-stage ML is that one first evaluates candidates of each marginal distribution with  $AIC_{ML}$  and consequently chooses the best distribution for each margin. Once the margins are chosen, one fits different copulas and evaluates the copula part with  $AIC_{ML}$ . However, this piecewise model evaluation cannot evaluate the model as a whole.

In this paper, we develop the copula information criterion (CIC) for two-stage ML estimation, which has the form

$$CIC = 2\ell_n(\tilde{\eta}) - 2\tilde{p}_{\eta}^*.$$

Here  $\ell_n(\tilde{\eta})$  is the maximized log-likelihood with the two-stage ML estimation method, in terms of the full parameter vector  $\eta$  of the model in question, and  $\tilde{p}_{\eta}^*$  is a certain penalization factor, worked out in Section 2.2. The main advantage of CIC is that it can evaluate a parametric copula with parametric margins as a whole. CIC is also a model-robust model selection criterion which means that it does not assume that the candidate model contains the true model. As the overlap of the name already suggests, our CIC is analogous to the CIC from Grønneberg and Hjort (2014), which is designed for copulas estimated with pseudo maximum likelihood (PML). In the PML framework, margins are estimated empirically, i.e. nonparametrically, while two-stage ML assumes parametric forms of margins.

Our technical setting, identical to Ko and Hjort (2019), is as follows. Let  $(Y_1, \dots, Y_d)^T$  be a  $d$ -variate continuous stochastic variable originating from a joint density  $g(y_1, \dots, y_d)$  and let  $y_i = (y_{i,1}, \dots, y_{i,d})^T$ , for  $i = 1, \dots, n$ , be independent observations of this variable. The true joint distribution  $g$  is typically unknown. Let  $f(y_1, \dots, y_d, \eta)$  be our parametric approximation of  $g$ , with the parameter vector  $\eta$ , belonging to some connected subset of the appropriate Euclidean space. In addition,  $G$  and  $F(\cdot, \eta)$  indicate cumulative distribution functions corresponding to  $g$  and  $f(\cdot, \eta)$ , respectively. Here  $G_j(y_j)$  and  $F_j(y_j, \alpha_j)$  indicate  $j$ th marginal distribution functions corresponding to  $G$  and  $F(\cdot, \eta)$  respectively, with  $\alpha_j$  as the parameter vector belonging to margin component  $j$ .

According to Sklar's theorem (Sklar, 1959), there always exists a copula  $C(u_1, \dots, u_d, \theta)$  that satisfies

$$F(y_1, \dots, y_d, \eta) = C(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta),$$

where the full parameter vector  $\eta$  is now blocked as

$$\eta = (\alpha^T, \theta^T)^T = (\alpha_1^T, \dots, \alpha_d^T, \theta^T)^T.$$

By assuming the regularity conditions from Ko and Hjort (2019),  $C(\cdot, \theta)$  can be differentiated,

$$f(y_1, \dots, y_d, \eta) = c(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta) \prod_{j=1}^d f_j(y_j, \alpha_j),$$

where  $c(u_1, \dots, u_d) = \partial^d C(u_1, \dots, u_d, \theta) / \partial u_1 \dots \partial u_d$  and  $f_j(y_j, \alpha_j) = \partial F_j(y_j, \alpha_j) / \partial y_j$ . For further details of copula modeling, see Joe (1997) and Nelsen (2006). Analogously, the true density  $g$  can also be decomposed into marginal densities and the copula density

$$g(y_1, \dots, y_d) = c_0(G_1(y_1), \dots, G_d(y_d)) \prod_{j=1}^d g_j(y_j),$$

with  $c_0(\cdot)$  the true copula.

Note that the theories of CIC and  $AIC_{2ML}$ , which we will develop in Section 2, should hold for both continuous and discrete variables since the two-stage ML estimation keeps its properties in case of discrete variables. In this paper, however, we only consider the case of continuous variables for simpler notation and exposition.

The further structure of this paper is as follows. In Section 2.1, we briefly explain Kullback–Leibler divergence and its relationship to TIC and  $AIC_{ML}$ . In Section 2.2, we derive and define our copula information criterion. In Section 2.3, we prove

that the  $AIC_{2ML}$  formula holds under the two-stage ML estimation. In Section 2.4, we summarize the relationship between TIC, CIC,  $AIC_{ML}$  and  $AIC_{2ML}$ . In Section 2.5, we illustrate what CIC looks like in the two-dimensional setting and show how CIC easily can be extended to the conditional copula setting. In Section 3, we study the numerical behavior of those model selection criteria. In our final Section 4, we offer a few concluding remarks and suggestions for future research.

## 2. The copula information criterion for two-stage maximum likelihood estimation

### 2.1. Kullback–Leibler divergence

The Kullback–Leibler (KL) divergence from  $g$  to  $f$  measures how the density  $f$  diverges from  $g$  (Kullback and Leibler, 1951) and is defined as

$$KL(g, f) = \int g(y) \log \frac{g(y)}{f(y)} dy = \int g(y) \log g(y) dy - \int g(y) \log f(y) dy.$$

It is well known that the ML estimator aims for parameter values that minimize the Kullback–Leibler divergence (Akaike, 1998).

Consider a case where one has competing models for certain data and the parameters are estimated by ML. An arbitrary candidate model with ML parameter estimates can be denoted as  $f(y, \hat{\eta})$ . (Throughout this paper, we use ‘ $\hat{\cdot}$ ’ to indicate that a quantity is estimated with ML and ‘ $\sim$ ’ to indicate that a quantity is estimated with two-stage ML.) Since  $g(y)$  is the same across all candidate models, minimizing KL divergence is equal to maximizing  $Q(\hat{\eta}) = \int g(y) \log f(y, \hat{\eta}) dy$ . This quantity is, however, not directly observable since  $g(y)$  is unknown. As an alternative, one may use the empirical equivalent of  $Q(\hat{\eta})$ :

$$\hat{Q}(\hat{\eta}) = \frac{1}{n} \ell(\hat{\eta}) = \frac{1}{n} \sum_{i=1}^n [\log f_1(y_{i,1}, \hat{\alpha}_1) + \cdots + \log f_d(y_{i,d}, \hat{\alpha}_d) + \log c(F_1(y_{i,1}, \hat{\alpha}_1), \dots, F_d(y_{i,d}, \hat{\alpha}_d), \hat{\theta})].$$

Yet, this estimator of  $Q(\hat{\eta})$  is a biased estimator. By identifying and subtracting the bias, we obtain an approximately unbiased estimator of  $Q(\hat{\eta})$ :

$$\hat{Q}^*(\hat{\eta}) = \frac{1}{n} \ell(\hat{\eta}) - \frac{1}{n} \text{tr}(\hat{\mathcal{I}}_{\eta}^{-1} \hat{K}_{\eta}).$$

where  $\hat{\mathcal{I}}_{\eta}$  is the observed information and  $\hat{K}_{\eta}$  is the estimated covariance matrix of the score vector.

The TIC (Takeuchi, 1976) aims for the model that maximizes  $\hat{Q}^*(\hat{\eta})$  and is defined as

$$TIC = 2\ell(\hat{\eta}) - 2\hat{p}_{\eta, TIC}^*, \quad (1)$$

where  $\hat{p}_{\eta, TIC}^* = \text{tr}(\hat{\mathcal{I}}_{\eta}^{-1} \hat{K}_{\eta})$ . This shows that TIC is basically a scaled version of  $\hat{Q}^*(\hat{\eta})$ .

When one boldly makes the assumption that the candidate model is correct, i.e. contains the true data generating mechanism, TIC simplifies to  $AIC_{ML}$ , possibly the most well known model selection criterion in statistics, which is defined as

$$AIC_{ML} = 2\ell(\hat{\eta}) - 2p_{\eta}, \quad (2)$$

where  $p_{\eta}$  is the length of the parameter vector  $\eta$ . Note that the formula of TIC and  $AIC_{ML}$  are only valid if the parameters are estimated by the ML estimator. For more details about TIC and  $AIC_{ML}$ , see chapter 2 of Claeskens and Hjort (2008).

### 2.2. Derivation of the copula information criterion

When the copula model is estimated with the two-stage ML, the bias correction term of TIC, i.e.  $\hat{p}_{\eta, TIC}^*$ , is not valid. We now derive the copula information criterion (CIC) which is analogous to TIC and is made for copula models estimated with the two-stage ML.

When the copula model is estimated with the two-stage ML, the non-constant part of the KL divergence is

$$Q(\tilde{\eta}) = \int g(y) \{ \log f_1(y_1, \tilde{\alpha}_1) + \cdots + \log f_d(y_d, \tilde{\alpha}_d) + \log c(F_1(y_1, \tilde{\alpha}_1), \dots, F_d(y_d, \tilde{\alpha}_d), \tilde{\theta}) \} dy.$$

The empirical equivalent is

$$\tilde{Q}(\tilde{\eta}) = \frac{1}{n} \sum_{i=1}^n [\log f_1(y_{i,1}, \tilde{\alpha}_1) + \cdots + \log f_d(y_{i,d}, \tilde{\alpha}_d) + \log c(F_1(y_{i,1}, \tilde{\alpha}_1), \dots, F_d(y_{i,d}, \tilde{\alpha}_d), \tilde{\theta})].$$

Now we check the bias of  $\tilde{Q}(\tilde{\eta})$ :

$$\begin{aligned} E_G[\tilde{Q}(\tilde{\eta})] - Q(\tilde{\eta}) &= E_{G_1} \left[ \frac{1}{n} \sum_{i=1}^n \log f_1(y_{i,1}, \tilde{\alpha}_1) \right] - \int g(y_1) \log f_1(y_1, \tilde{\alpha}_1) dy_1 \\ &\quad + \cdots \end{aligned}$$

$$\begin{aligned}
& + E_{G_d} \left[ \frac{1}{n} \sum_{i=1}^n \log f_d(y_{i,d}, \tilde{\alpha}_d) \right] - \int g(y_d) \log f_d(y_d, \tilde{\alpha}_d) dy_d \\
& + E_G \left[ \frac{1}{n} \sum_{i=1}^n \log c(F_1(y_{i,1}, \tilde{\alpha}_1), \dots, F_d(y_{i,d}, \tilde{\alpha}_d), \tilde{\theta}) \right] \\
& - \int g(y) \log c(F_1(y_1, \tilde{\alpha}_1), \dots, F_d(y_d, \tilde{\alpha}_d), \tilde{\theta}) dy.
\end{aligned}$$

Since the parameters of marginals from stage 1 ( $\tilde{\alpha}_j$ s) are obtained with ML estimation, we can directly use the results from the derivation of TIC and obtain

$$E_{G_j} \left[ \frac{1}{n} \sum_{i=1}^n \log f_j(y_{i,j}, \tilde{\alpha}_j) \right] - \int g(y_j) \log f(y_j, \tilde{\alpha}_j) dy_j = \frac{1}{n} \text{tr}(\mathcal{I}_{\alpha_j}^{-1} K_{\alpha_j}) + o(n^{-1}) = \frac{1}{n} \tilde{p}_{\alpha_j}^* + o(n^{-1}),$$

where

$$\begin{aligned}
K_{\alpha_j} &= E_{G_j} [U_{\alpha_j}(y_j, \alpha_{0,j}) U_{\alpha_j}(y_j, \alpha_{0,j})^T], \\
U_{\alpha_j}(y_j, \alpha_j) &= \frac{\partial \log f_j(y_j, \alpha_j)}{\partial \alpha_j}
\end{aligned}$$

and

$$\mathcal{I}_{\alpha_j} = -E_{G_j} \left[ \frac{\partial^2 \log f_j(y_j, \alpha_{0,j})}{\partial \alpha_{0,j} \partial \alpha_{0,j}^T} \right].$$

In a nutshell,  $\mathcal{I}_{\alpha_j}$  is the Fisher information of  $j$ th margin and  $K_{\alpha_j}$  is the covariance matrix of the score vector that belongs to  $j$ th margin.

Further, let

$$Q_c(\tilde{\eta}) = \int g(y) \log c(F_1(y_1, \tilde{\alpha}_1), \dots, F_d(y_d, \tilde{\alpha}_d), \tilde{\theta}) dy$$

and

$$\tilde{Q}_c(\tilde{\eta}) = \frac{1}{n} \sum_{i=1}^n \log c(F_1(y_{i,1}, \tilde{\alpha}_1), \dots, F_d(y_{i,d}, \tilde{\alpha}_d), \tilde{\theta}).$$

So, we can write

$$E_G[\tilde{Q}_c(\tilde{\eta})] - Q_c(\tilde{\eta}) = \frac{1}{n} \sum_{j=1}^d \text{tr}(\mathcal{I}_{\alpha_j}^{-1} K_{\alpha_j}) + E_G[\tilde{Q}_c(\tilde{\eta})] - Q_c(\tilde{\eta}) + o(n^{-1}).$$

Now,  $E_G[\tilde{Q}_c(\tilde{\eta})] - Q_c(\tilde{\eta})$  is the only element that should be evaluated. Let

$$\begin{aligned}
Q_c(\eta_0) &= \int g(y) \log c(F_1(y_1, \alpha_{0,1}), \dots, F_d(y_d, \alpha_{0,d}), \theta_0) dy, \\
Z_i &= \log c(F_1(y_{i,1}, \alpha_{0,1}), \dots, F_d(y_{i,d}, \alpha_{0,d}), \theta_0) - Q_c(\eta_0),
\end{aligned}$$

and

$$A_\eta = \sqrt{n}(\tilde{\eta} - \eta_0) = \sqrt{n} \begin{pmatrix} \mathcal{I}_\alpha & 0 \\ \mathcal{I}_{\alpha,\theta}^T & \mathcal{I}_\theta \end{pmatrix}^{-1} \begin{pmatrix} U_{n,\alpha}(\alpha_0) \\ U_{n,\theta}(\alpha_0, \theta_0) \end{pmatrix} + \begin{pmatrix} o_p(1) \\ o_p(1) \end{pmatrix},$$

which stems from Proposition 1 in Ko and Hjort (2019), and furthermore

$$\begin{aligned}
U_{n,\eta}(\eta) &= \frac{1}{n} \sum_{i=1}^n U_\eta(y_i, \eta) = \frac{1}{n} \sum_{i=1}^n \frac{\partial \log c(F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta)}{\partial \eta}, \\
H_{n,\eta}(\eta) &= \frac{1}{n} \sum_{i=1}^n H_\eta(y_i, \eta) = \frac{1}{n} \sum_{i=1}^n \frac{\partial^2 \log c(F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta)}{\partial \eta \partial \eta^T},
\end{aligned}$$

and

$$\mathcal{I}_\eta^* = -E_G[H_\eta(y, \eta_0)] = - \int g(y) H_\eta(y, \eta_0) dy,$$

which, incidentally, should not be confused with  $\mathcal{I}_\eta$  in Proposition 1 of Ko and Hjort (2019). Then we have  $E[\bar{Z}_n] = \frac{1}{n} \sum_{i=1}^n E[Z_i] = 0$ , along with

$$\begin{aligned}\tilde{Q}_c(\tilde{\eta}) &= \frac{1}{n} \sum_{i=1}^n \log c(y_i, \tilde{\eta}) \\ &= \frac{1}{n} \sum_{i=1}^n \left[ \log c(y_i, \eta_0) - Q_c(\eta_0) + Q_c(\eta_0) + (\tilde{\eta} - \eta_0)^T U_\eta(y_i, \eta_0) + \frac{1}{2} (\tilde{\eta} - \eta_0)^T H_\eta(y_i, \eta_0) (\tilde{\eta} - \eta_0) \right] + o_p(n^{-1}) \\ &= \frac{1}{n} \sum_{i=1}^n [Z_i] + Q_c(\eta_0) + (\tilde{\eta} - \eta_0)^T U_{n,\eta}(\eta_0) + \frac{1}{2} (\tilde{\eta} - \eta_0)^T H_{n,\eta}(\eta_0) (\tilde{\eta} - \eta_0) + o_p(n^{-1}) \\ &= Q_c(\eta_0) + \bar{Z}_n + \frac{1}{\sqrt{n}} A_\eta^T U_{n,\eta}(\eta_0) + \frac{1}{2n} A_\eta^T H_{n,\eta}(\eta_0) A_\eta + o_p(n^{-1}),\end{aligned}$$

$$\begin{aligned}Q_c(\tilde{\eta}) &= \int g(y) \log c(F_1(y_1, \tilde{\alpha}_1), \dots, F_d(y_d, \tilde{\alpha}_d), \tilde{\theta}) dy \\ &= \int g(y) \left[ \log c(y, \eta_0) + (\tilde{\eta} - \eta_0)^T U_\eta(y, \eta_0) + \frac{1}{2} (\tilde{\eta} - \eta_0)^T H_\eta(y, \eta_0) (\tilde{\eta} - \eta_0) \right] dy + o_p(n^{-1}) \\ &= \int g(y) \log c(y, \eta_0) dy + (\tilde{\eta} - \eta_0)^T \int g(y) U_\eta(y, \eta_0) dy + \frac{1}{2} (\tilde{\eta} - \eta_0)^T \int g(y) H_\eta(y, \eta_0) dy \cdot (\tilde{\eta} - \eta_0) + o_p(n^{-1}) \\ &= Q_c(\eta_0) + (\tilde{\eta} - \eta_0)^T \cdot 0 + \frac{1}{2n} \sqrt{n} (\tilde{\eta} - \eta_0)^T \int g(y) H_\eta(y, \eta_0) dy \cdot \sqrt{n} (\tilde{\eta} - \eta_0) + o_p(n^{-1}) \\ &= Q_c(\eta_0) - \frac{1}{2n} A_\eta^T \mathcal{I}_\eta^* A_\eta + o_p(n^{-1}),\end{aligned}$$

and

$$n(\tilde{Q}_c(\tilde{\eta}) - Q_c(\tilde{\eta})) = n\bar{Z}_n + \sqrt{n} A_\eta^T U_{n,\eta}(\eta_0) + \frac{1}{2} A_\eta^T H_{n,\eta}(\eta_0) A_\eta + \frac{1}{2} A_\eta^T \mathcal{I}_\eta^* A_\eta + o_p(1).$$

Further, note that

$$U_{n,\eta}(\eta_0) = \frac{1}{n} \sum_{i=1}^n \frac{\partial \log c(F_1(y_{i,1}, \alpha_{0,1}), \dots, F_d(y_{i,d}, \alpha_{0,d}), \theta_0)}{\partial \eta_0}$$

has the following convergence, by the central limit theorem:

$$\sqrt{n} U_{n,\eta}(\eta_0) = \sqrt{n} (U_{n,\eta}(\eta_0) - E[U_\eta(Y, \eta_0)]) \xrightarrow{d} \Lambda_\eta^* \sim N(0, K_\eta^*),$$

where  $K_\eta^* = \text{Var}(U_\eta(y, \eta_0)) = E[U_\eta(y, \eta_0) U_\eta(y, \eta_0)^T]$ . (Here  $\Lambda_\eta^*$  and  $K_\eta^*$  should not be confused with  $\Lambda_\eta$  and  $K_\eta$  in Proposition 1 of Ko and Hjort (2019).)

Now we evaluate  $E_G[n(\tilde{Q}_c(\tilde{\eta}) - Q_c(\tilde{\eta}))]$ :

$$\begin{aligned}E_G[n(\tilde{Q}_c(\tilde{\eta}) - Q_c(\tilde{\eta}))] &= E_G \left[ n\bar{Z}_n + \sqrt{n} A_\eta^T U_{n,\eta}(\eta_0) + \frac{1}{2} A_\eta^T H_{n,\eta}(\eta_0) A_\eta + \frac{1}{2} A_\eta^T \mathcal{I}_\eta^* A_\eta \right] + o(1) \\ &\rightarrow E_G \left[ (\mathcal{I}_\eta^{-1} \Lambda_\eta)^T \Lambda_\eta^* \right] \\ &= E_G \left[ \text{tr}(\mathcal{I}_\eta^{-1} \Lambda_\eta (\Lambda_\eta^*)^T) \right] \\ &= \text{tr}(\mathcal{I}_\eta^{-1} E_G[\Lambda_\eta (\Lambda_\eta^*)^T]) = \text{tr}(\mathcal{I}_\eta^{-1} K_\eta^\circ) = p_\theta^*,\end{aligned}$$

where

$$K_\eta^\circ = E_G[\Lambda_\eta (\Lambda_\eta^*)^T] = E_G \left[ \begin{pmatrix} \Lambda_\alpha (\Lambda_\alpha^*)^T & \Lambda_\alpha \Lambda_\theta^T \\ \Lambda_\theta (\Lambda_\alpha^*)^T & \Lambda_\theta \Lambda_\theta^T \end{pmatrix} \right] = \begin{pmatrix} K_\alpha^\circ & K_{\alpha,\theta} \\ (K_{\alpha,\theta}^*)^T & K_\theta \end{pmatrix}.$$

It is practical to note that

$$\begin{aligned}\text{tr}(\mathcal{I}_\eta^{-1} K_\eta^\circ) &= \text{tr} \left( \begin{pmatrix} \mathcal{I}_\alpha & 0 \\ \mathcal{I}_{\alpha,\theta}^T & \mathcal{I}_\theta \end{pmatrix}^{-1} \begin{pmatrix} K_\alpha^\circ & K_{\alpha,\theta} \\ (K_{\alpha,\theta}^*)^T & K_\theta \end{pmatrix} \right) \\ &= \text{tr} \left( \begin{pmatrix} \mathcal{I}_\alpha^{-1} K_\alpha^\circ & \mathcal{I}_\alpha^{-1} K_{\alpha,\theta} \\ -\mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha,\theta}^T \mathcal{I}_\alpha^{-1} K_\alpha^\circ + \mathcal{I}_\theta^{-1} (K_{\alpha,\theta}^*)^T & -\mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha,\theta}^T \mathcal{I}_\alpha^{-1} K_{\alpha,\theta} + \mathcal{I}_\theta^{-1} K_\theta \end{pmatrix} \right)\end{aligned}$$

$$= \text{tr} \left( \begin{pmatrix} \mathcal{I}_\alpha^{-1} K_\alpha^\circ & 0 \\ 0 & -\mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha, \theta}^\top \mathcal{I}_\alpha^{-1} K_{\alpha, \theta} + \mathcal{I}_\theta^{-1} K_\theta \end{pmatrix} \right).$$

Consistent estimators for  $\mathcal{I}_{\alpha_j}^{-1}$ ,  $K_{\alpha_j}$ ,  $\mathcal{I}_\eta^{-1}$  and  $K_\eta^\circ$  can be obtained by using plug-in sample averages. The consequent estimators are denoted as  $\tilde{\mathcal{I}}_{\alpha_j}^{-1}$ ,  $\tilde{K}_{\alpha_j}$ ,  $\tilde{\mathcal{I}}_\eta^{-1}$ ,  $\tilde{K}_\eta^\circ$ . For regularity conditions that ensure convergence in probability, see Jullum and Hjort (2017).

Thus, the approximately unbiased estimator of  $Q(\tilde{\eta})$  is

$$\tilde{Q}^*(\tilde{\eta}) = \frac{1}{n} \ell_n(\tilde{\eta}) - \frac{1}{n} \left( \sum_{j=1}^d \text{tr}(\tilde{\mathcal{I}}_{\alpha_j}^{-1} \tilde{K}_{\alpha_j}) + \text{tr}(\tilde{\mathcal{I}}_\eta^{-1} \tilde{K}_\eta^\circ) \right).$$

By defining  $\tilde{p}_{\alpha_j}^* = \text{tr}(\tilde{\mathcal{I}}_{\alpha_j}^{-1} \tilde{K}_{\alpha_j})$ ,  $\tilde{p}_\theta^* = \text{tr}(\tilde{\mathcal{I}}_\eta^{-1} \tilde{K}_\eta^\circ)$ ,  $\tilde{p}_\eta^* = \sum_{j=1}^d \tilde{p}_{\alpha_j}^* + \tilde{p}_\theta^*$  and scaling  $\tilde{Q}^*(\tilde{\eta})$ , we can finally define the copula information criterion as

$$\text{CIC} = 2\ell_n(\tilde{\eta}) - 2\tilde{p}_\eta^*. \quad (3)$$

### 2.3. AIC for two-stage maximum likelihood estimator

The CIC, derived in Section 2.2, is a model robust model selection criterion. This means that the CIC does not assume that the parametric model includes the true model that generated data. In this section we show that, if we do make such a true model assumption, the CIC simplifies to the AIC<sub>2ML</sub>. To our knowledge, this is the first time that the validity of the AIC<sub>2ML</sub> formula is proven for the two-stage ML estimator.

**Lemma 1.** Under the assumption that the margins and copula are correctly specified, it holds that  $K_\alpha^\circ = \mathcal{I}_\alpha - K_\alpha$ .

**Proof.** Assume the candidate model worked with contains the true data generating mechanism, i.e. that  $f = g$  at the relevant parameter point. Then

$$\begin{aligned} 0 &= \frac{\partial \text{E}_G[U_\alpha(y, \alpha_0)]^\top}{\partial \alpha_0} \\ &= \frac{\partial}{\partial \alpha_0} \int f \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0^\top} dy \\ &= \int \left( \frac{\partial f}{\partial \alpha_0} \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0^\top} + f \frac{\partial^2 \sum_{j=1}^d \log f_j}{\partial \alpha_0 \partial \alpha_0^\top} \right) dy \\ &= \int \left( f \frac{\partial \log f}{\partial \alpha_0} \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0^\top} + f \frac{\partial^2 \sum_{j=1}^d \log f_j}{\partial \alpha_0 \partial \alpha_0^\top} \right) dy \\ &= \int \left( f \left( \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0} + \frac{\partial \log c}{\partial \alpha_0} \right) \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0^\top} + f \frac{\partial^2 \sum_{j=1}^d \log f_j}{\partial \alpha_0 \partial \alpha_0^\top} \right) dy \\ &= \int f \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0} \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0^\top} dy + \int \frac{\partial \log c}{\partial \alpha_0} \frac{\partial \sum_{j=1}^d \log f_j}{\partial \alpha_0^\top} dy + \int f \frac{\partial^2 \sum_{j=1}^d \log f_j}{\partial \alpha_0 \partial \alpha_0^\top} dy \\ &= \text{E}_G[U_\alpha(y, \alpha_0) U_\alpha(y, \alpha_0)^\top] + \text{E}_G[U_\alpha^*(y, \alpha_0) U_\alpha(y, \alpha_0)^\top] - \text{E}_G[-H_\alpha(y, \alpha_0)] \\ &= K_\alpha + K_\alpha^\circ - \mathcal{I}_\alpha. \quad \square \end{aligned}$$

If we make the true model assumption (i.e.  $f = g$ ), we have from the classical results of maximum likelihood theory that  $\mathcal{I}_{\alpha_j} = K_{\alpha_j}$ . This implies that we have for the bias correction term in the marginal parameters  $p_{\alpha_j}^* = \text{tr}(\mathcal{I}_{\alpha_j}^{-1} K_{\alpha_j}) = \dim(\alpha_j)$ , for each  $j$ .

For the bias correction term in the copula parameters part, the true model assumption results in Lemma 1. Combining Lemma 1 with Lemmas 3 and 5 from Ko and Hjort (2019) gives

$$\begin{aligned} \text{tr}(\mathcal{I}_\eta^{-1} K_\eta^\circ) &= \text{tr} \left( \begin{pmatrix} \mathcal{I}_\alpha^{-1} K_\alpha^\circ & 0 \\ 0 & -\mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha, \theta}^\top \mathcal{I}_\alpha^{-1} K_{\alpha, \theta} + \mathcal{I}_\theta^{-1} K_\theta \end{pmatrix} \right) \\ &= \text{tr} \left( \begin{pmatrix} I - \mathcal{I}_\alpha^{-1} K_\alpha & 0 \\ 0 & I \end{pmatrix} \right) \\ &= \text{tr} \left( \begin{pmatrix} 0 & 0 \\ 0 & I \end{pmatrix} \right) = \dim(\theta). \end{aligned}$$

**Table 1**

An overview of the relationship between model selection criteria discussed in this paper.

|            |     | Model robust |                    |
|------------|-----|--------------|--------------------|
|            |     | Yes          | No                 |
| Estimation | ML  | TIC          | AIC <sub>ML</sub>  |
|            | 2ML | CIC          | AIC <sub>2ML</sub> |

Thus, the unbiased estimator  $\tilde{Q}^*(\tilde{\eta})$  can be simplified as

$$\tilde{Q}^*(\tilde{\eta}) = \frac{1}{n} \ell_n(\tilde{\eta}) - \frac{1}{n} \left( \sum_{j=1}^d \dim(\alpha_j) + \dim(\theta) \right).$$

By defining  $p_\eta = \sum_{j=1}^d \dim(\alpha_j) + \dim(\theta) = \dim(\eta)$  and scaling  $\tilde{Q}^*(\tilde{\eta})$ , we obtain AIC<sub>2ML</sub> for the two-stage ML estimated copula models

$$\text{AIC}_{2\text{ML}} = 2\ell_n(\tilde{\eta}) - 2p_\eta. \quad (4)$$

Compared to AIC<sub>ML</sub> from Section 2.1, the only difference is that the log-likelihood is now estimated under the two-stage ML instead of ML, i.e. we use  $\tilde{\eta}$  instead of  $\hat{\eta}$ .

#### 2.4. Relationship between the model selection criteria

So far, we have discussed four model selection criteria for copula models. Table 1 offers an overview of the relationship between them. When the model (i.e. both copula and margins) is correctly specified, CIC and AIC<sub>2ML</sub> become equal, and the same happens between TIC and AIC<sub>ML</sub>. Thus, one can compare CIC and AIC<sub>2ML</sub> (or TIC and AIC<sub>ML</sub> in case of ML estimation) to check whether the model is correctly specified.

Further, since both CIC and TIC are estimating the same part of the KL divergence under the same model robust environment, they are compatible. This implies that one can compare CIC to TIC to measure how much one loses in terms of KL divergence by using two-stage ML estimation instead of ML estimation. The same can be done by comparing AIC<sub>2ML</sub> to AIC<sub>ML</sub> when one believes in the model. However, one should not compare CIC with AIC<sub>ML</sub> (or TIC with AIC<sub>2ML</sub>) since they are based on two different model beliefs (i.e. presence of the true model assumption). Figuratively speaking, one would in that situation be comparing apples with pears.

#### 2.5. Illustration for two-dimensional case and extension to the conditional copula regression.

To make things more concrete, we now give an example of the two-dimensional case with  $(Y_1, Y_2)$  from the unknown  $g(y)$ . As candidate margins we choose a two-parameter distribution (e.g. normal, gamma, Weibull, etc.) for both  $F_1$  and  $F_2$ . For the copula part, we choose a one-parameter copula (e.g. Gumbel, Frank, Clayton, etc.). The candidate model then has the form

$$f(y_1, y_2, \eta) = c(F_1(y_1, \alpha_1), F_2(y_2, \alpha_2), \theta) \cdot f_1(y_1, \alpha_1) \cdot f_2(y_2, \alpha_2),$$

where  $\eta = (\alpha_{1,1}, \alpha_{1,2}, \alpha_{2,1}, \alpha_{2,2}, \theta)^T$ . The ingredients for the CIC can then be written down by using the block matrix form, in line with Ko and Hjort (2019):

$$K_\eta = \begin{pmatrix} K_{\alpha_1} & K_{\alpha_1, \alpha_2} & K_{\alpha_1, \theta} \\ K_{\alpha_2, \alpha_1} & K_{\alpha_2} & K_{\alpha_2, \theta} \\ K_{\alpha_1, \theta}^T & K_{\alpha_2, \theta}^T & K_\theta \end{pmatrix}, \quad \mathcal{I}_\eta = \begin{pmatrix} \mathcal{I}_{\alpha_1} & 0 & 0 \\ 0 & \mathcal{I}_{\alpha_2} & 0 \\ \mathcal{I}_{\alpha_1, \theta}^T & \mathcal{I}_{\alpha_2, \theta}^T & \mathcal{I}_\theta \end{pmatrix}, \quad K_\eta^\circ = \begin{pmatrix} K_{\alpha_1}^\circ & K_{\alpha_1, \alpha_2}^\circ & K_{\alpha_1, \theta}^\circ \\ K_{\alpha_2, \alpha_1}^\circ & K_{\alpha_2}^\circ & K_{\alpha_2, \theta}^\circ \\ (K_{\alpha_1, \theta}^*)^T & (K_{\alpha_2, \theta}^*)^T & K_\theta \end{pmatrix}.$$

We consequently have

$$\begin{aligned} p_\eta^* &= p_{\alpha_1}^* + p_{\alpha_2}^* + p_\theta^* \\ &= \text{tr}(\mathcal{I}_{\alpha_1}^{-1} K_{\alpha_1}) + \text{tr}(\mathcal{I}_{\alpha_2}^{-1} K_{\alpha_2}) + \text{tr}(\mathcal{I}_\eta^{-1} K_\eta^\circ) \\ &= \text{tr}(\mathcal{I}_{\alpha_1}^{-1} K_{\alpha_1}) + \text{tr}(\mathcal{I}_{\alpha_2}^{-1} K_{\alpha_2}) + \text{tr}(\mathcal{I}_{\alpha_1}^{-1} K_{\alpha_1}^\circ) + \text{tr}(\mathcal{I}_{\alpha_2}^{-1} K_{\alpha_2}^\circ) \\ &\quad + \text{tr}(-\mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha_1, \theta}^T \mathcal{I}_{\alpha_1}^{-1} K_{\alpha_1, \theta} - \mathcal{I}_\theta^{-1} \mathcal{I}_{\alpha_2, \theta}^T \mathcal{I}_{\alpha_2}^{-1} K_{\alpha_2, \theta} + \mathcal{I}_\theta^{-1} K_\theta). \end{aligned}$$

More generally, we can consider the conditional copula regression where all marginal distributions and copula are conditioned on the  $k$ -variate covariate  $X$  parametrically. Patton (2002) extends the existing theories of copula to the conditional copula setting, including the conditional version of Sklar's theorem, which gives conditional copula density

$$f(y_1, \dots, y_d, \eta | x) = c(F_1(y_1, \alpha_1 | x), F_2(y_2, \alpha_2 | x), \theta | x) \cdot f_1(y_1, \alpha_1 | x) \cdot f_2(y_2, \alpha_2 | x).$$

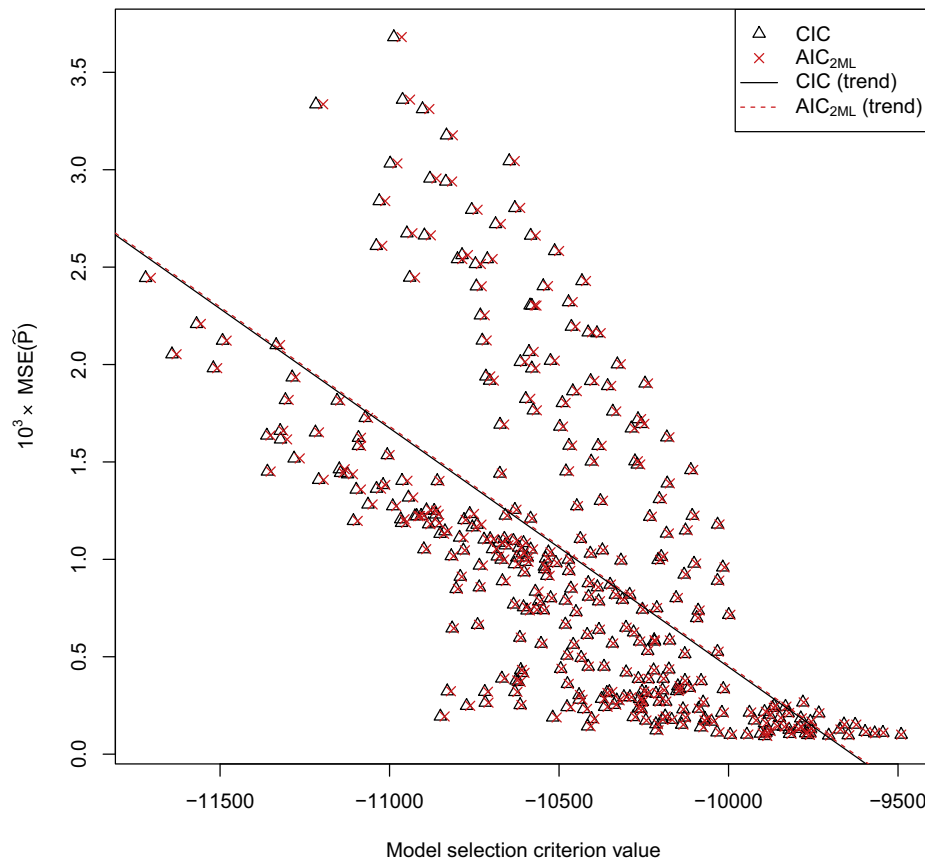


**Fig. 1.** Result from simulation 1. On the x-axis is the value of CIC or  $AIC_{2ML}$ . The 324 different copula models, defined by using the candidate copulas and margins described in Table 3, are fitted to the dataset generated from the data generating model described in Table 2. The y-axis is the mean squared error of  $\tilde{P}(q_{0.7} < Y)$ , which indicates the two-stage ML estimated joint probability that each marginal variable has larger value than its 0.7-quantile value, defined by the data generating model.

For simplicity, we consider the case where the copula parameter  $\theta$  is modeled by the linear calibration function with  $\theta = X\beta$  where  $\beta = (\beta_0, \dots, \beta_k)^T$  is a  $k + 1$  dimensional parameters. We can then consider  $\beta$  as the copula parameter instead of  $\theta$ , which results in  $\eta = (\alpha_{1,1}, \alpha_{1,2}, \alpha_{2,1}, \alpha_{2,2}, \beta_0, \dots, \beta_k)^T$ . One can easily define matrices  $K_\eta$ ,  $\mathcal{I}_\eta$  and  $K_\eta^\circ$  accordingly. For details about conditional copula regression, see Patton (2006), Acar et al. (2013) and Palaro and Hotta (2006).

### 3. Simulation studies

To study the behavior of CIC, we have performed a set of simulation studies. Simulation 1 studies the similarity between CIC and  $AIC_{2ML}$  and whether the models with higher CIC or  $AIC_{2ML}$  scores are indeed ‘better’ models in practice. Simulation 2 focuses on the difference in the behavior of CIC and  $AIC_{2ML}$ . In particular, we analyze whether CIC outperforms  $AIC_{2ML}$  in practice.

#### 3.1. Simulation 1

In simulation 1, we generated a dataset of size  $n = 1000$  with the model described in Table 2. We then, with this dataset, fitted 324 different copula models, which are based on the possible combinations of the candidate copulas and margins described in Table 3. Further,  $P(q_{0.7} < Y)$  was computed from every fitted model. Here,  $q_{0.7}$  is a vector that contains the 0.7-quantile value of each margin according to the true model. i.e.  $q_{0.7} = (G_1^{-1}(0.7), \dots, G_4^{-1}(0.7))$ . We repeated this process 1000 times and averaged the results.

Fig. 1 displays the relationship between mean squared error of estimated  $P(q_{0.7} < Y)$  and CIC and  $AIC_{2ML}$ . We can see that both model selection criteria evaluate the models that have lower mean squared error as better models. The difference between CIC and  $AIC_{2ML}$  in this perspective is minimal. This is because the log-likelihood, the element that is shared by both model selection criteria, has much bigger absolute value than the bias correction term and dominate the criteria. The difference between CIC and  $AIC_{2ML}$  (the misalignment between the black triangles and the red crosses in the figure) is caused by the difference in their bias correction part alone. Further, the model ranks determined by CIC and  $AIC_{2ML}$  are highly similar and both model selection criteria successfully picked the true data generating model as the best model in

**Table 2**

Description of the data generating model used in simulation 1.

|                       | Copula                 | Margin 1                                             | Margin 2                                           | Margin 3                                        | Margin 4                                        |
|-----------------------|------------------------|------------------------------------------------------|----------------------------------------------------|-------------------------------------------------|-------------------------------------------------|
| Data generating model | Gumbel<br>$\theta = 3$ | Weibull<br>$\alpha_1 = (1.5, 4)^T$<br>(shape, scale) | Weibull<br>$\alpha_2 = (2, 3)^T$<br>(shape, scale) | Gamma<br>$\alpha_3 = (2, 1)^T$<br>(shape, rate) | Gamma<br>$\alpha_4 = (3, 1)^T$<br>(shape, rate) |

**Table 3**

List of the candidate copulas and margins used in simulation 1.

|          | Candidates                                |
|----------|-------------------------------------------|
| Copula   | Gumbel, Gaussian, Frank, Survival Clayton |
| Margin 1 | Weibull, Gamma, Log-normal                |
| Margin 2 | Weibull, Gamma, Normal                    |
| Margin 3 | Gamma, Weibull, Log-normal                |
| Margin 4 | Gamma, Weibull, Log-normal                |

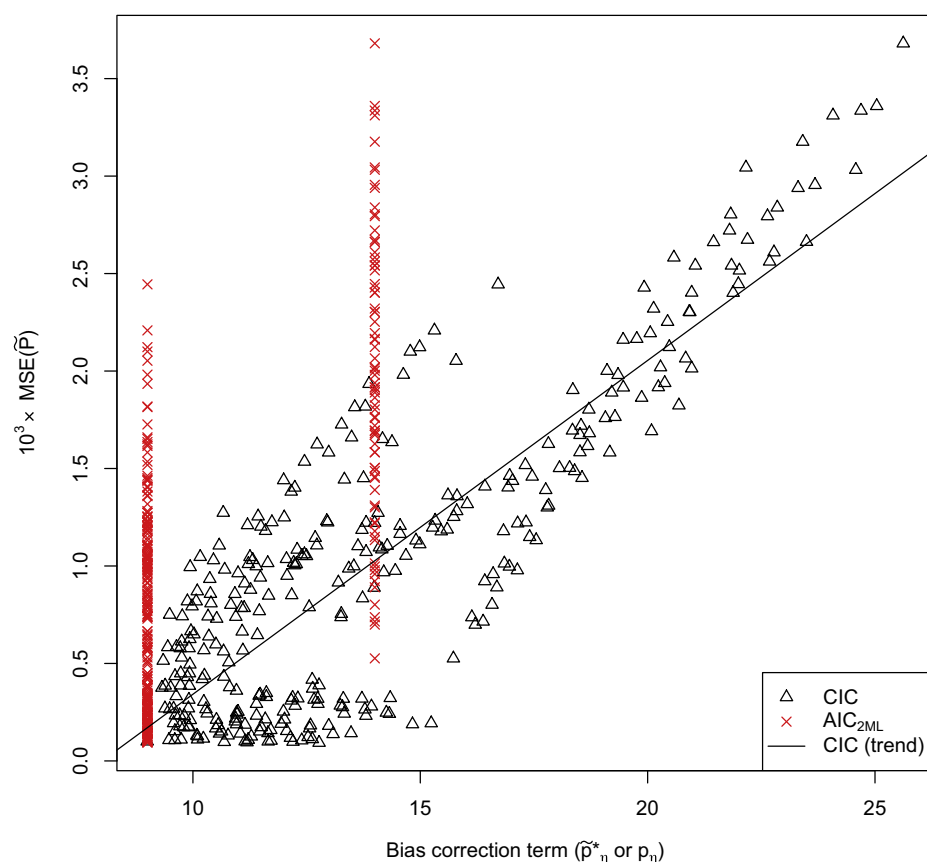


**Fig. 2.** Result from simulation 1. The 324 different copula models, defined by using the candidate copulas and margins described in Table 3, are fitted to the dataset generated from the data generating model described in Table 2. The y-axis is the mean squared error of  $\hat{P}(q_{0.7} < Y)$ . The x-axis is the value of the bias correction terms in model selection criteria ( $\hat{p}_\eta^*$  for CIC and  $p_\eta$  for  $AIC_{2ML}$ ).

all 1000 repetitions. In short, Fig. 1 illustrates that the KL-divergence based model selection criteria that we introduced for two-stage ML estimation (i.e. CIC and  $AIC_{2ML}$ ) show the desired behavior in practice.

Fig. 2 uses the same mean squared error as in Fig. 1, but the x-axis is now the bias correction term ( $\hat{p}_\eta^*$  for CIC and  $p_\eta$  for  $AIC_{2ML}$ ). The difference between CIC and  $AIC_{2ML}$  is now more clear. While  $p_\eta$  (dimension of the parameter vector  $\eta$ ) for  $AIC_{2ML}$  is either 9 or 14, depending on the total number of parameters in the model,  $\hat{p}_\eta^*$  tends to penalize misspecified models more and forms a strong relationship with the mean squared error of estimated  $P(q_{0.7} < Y)$ .

### 3.2. Simulation 2

In simulation 1, we observed that CIC penalizes misspecified models more in its bias correction part, which is a feature that  $AIC_{2ML}$  does not have. Does it imply that CIC should outperform  $AIC_{2ML}$  when they disagree on the model? To find an

**Table 4**  
Description of the models used in simulation 2.

|         | Dependence measure                                  | Copula                  | Margin 1                                                         | Margin 2                                                       |
|---------|-----------------------------------------------------|-------------------------|------------------------------------------------------------------|----------------------------------------------------------------|
| Model 1 | Kendall's $\tau = 0.56$<br>Spearman's $\rho = 0.76$ | Frank<br>$\theta = 7$   | Log-normal<br>$\alpha_1 = (0.9, 0.8)^T$<br>(mean, SD)<br>Weibull | Log-normal<br>$\alpha_2 = (0.3, 0.8)^T$<br>(mean, SD)<br>Gamma |
| Model 2 | Kendall's $\tau = 0.60$<br>Spearman's $\rho = 0.79$ | Clayton<br>$\theta = 3$ | $\alpha_1 = (1.5, 4)^T$<br>(shape, scale)                        | $\alpha_2 = (2, 1)^T$<br>(shape, rate)                         |

answer to this question and to study the difference in the behavior of  $AIC_{2ML}$  and CIC in detail, we performed simulation 2. Although we tried different combinations of copulas and margins, they all led to the same conclusion. Hence, we only display some chosen representative results.

Like we saw in Section 3.1, the difference between CIC and  $AIC_{2ML}$  is fairly small because of the dominance of the log-likelihood part. This means that there are not that many situations in practice where CIC and  $AIC_{2ML}$  disagree, especially when the sample size is large. So, we have to look into situations where the difference between two competing models is relatively small. We mimicked such a situation by letting the data come from a mixture of the two models. Another advantage of this simulation design is that it allows us to study the situation where there is no 'true model' on the candidate model list (when the 'contamination' parameter  $\delta$  is away from its boundaries).

In simulation 2, we generated datasets of 4 different sizes ( $n = 100, 200, 500, 1000$ ) from the two models described in Table 4. The data generating algorithm generated  $(1 - \delta) \cdot 100\%$  of data points from model 1 and  $\delta \cdot 100\%$  of data points from model 2. So,  $\delta$  can be seen as the degree of 'contamination' to model 1 by model 2. Then, we fitted both model 1 and model 2 to these data by using two-stage ML. This process was repeated 10,000 times. We used  $\delta = 0, 0.05, \dots, 0.95, 1$ .

Note that the copulas used in model 1 and 2 are easily distinguishable with reasonable sample size (e.g. 200), since Clayton copula has stronger lower tail dependence than Frank copula. This enables us to observe different degrees of misspecification with varying  $\delta$  values.

From Fig. 3, we can first confirm our theoretical finding from Section 2.3 that CIC and  $AIC_{2ML}$  become equal when the model is correctly specified. (For model 1, it is when  $\delta = 0$  and for model 2, it is when  $\delta = 1$ .) As the degree of model misspecification grows (for model 1, it is when  $\delta$  moves towards 1 and for model 2, it is when  $\delta$  moves towards 0), the two model selection criteria diverge from each other. To be more specific, CIC always has smaller value than  $AIC_{2ML}$ . Since the only difference between the two model selection criteria is the bias correction part and the bias correction part of  $AIC_{2ML}$  is constant for all values of  $\delta$ , this can only imply that  $\tilde{p}_{\eta}^*$  gets larger as the degree of misspecification grows. The divergence between CIC and  $AIC_{2ML}$  gets smaller and smaller as the sample size ( $n$ ) grows. This is because the size of the log-likelihood part of grows linearly with the sample size while the bias correction part does not.

Fig. 4 shows how often each model is chosen as the best model by each model selection criterion. One could intuitively think that CIC should choose the 'more correct' model (which is model 1 for  $\delta < 0.5$  and model 2 for  $\delta > 0.5$ ) more often than  $AIC_{2ML}$  since CIC is a model robust model selection criterion and the misspecified model gets penalized by  $\tilde{p}_{\eta}^*$ . However, the result shows that this is not the case. Fig. 4 shows that CIC always prefers model 1 above model 2, both when  $\delta < 0.5$  and  $\delta > 0.5$ . We tried many different copulas and margins and CIC always preferred a certain model across all range of  $\delta$ . Which model was preferred, depended on the combinations of copula and margins. Fig. 5, where we plot the bias correction term against  $\delta$ , explains this phenomenon. At a first glance, we can observe that  $\tilde{p}_{\eta}^*$  and  $p_{\eta}$  are equal when the model is correctly specified and that  $\tilde{p}_{\eta}^*$  grows as the degree of model misspecification increases. The key thing is that this growth happens in both models and that each model has its own speed of growth for  $\tilde{p}_{\eta}^*$ . Back to  $AIC_{2ML}$ , since both model 1 and 2 have same number of parameters, the model with higher log-likelihood value becomes the 'winning model'. In other words, whether  $AIC_{2ML}$  choose model 1 or 2 as winning model, depends on

$$\text{sgn}(\tilde{\ell}_{n,M1} - \tilde{\ell}_{n,M2}),$$

where  $\tilde{\ell}_{n,Mj}$  indicates the under two-stage ML maximized log-likelihood value of model  $j$ . So, CIC and  $AIC_{2ML}$  disagree on the winning model if and only if

$$\text{sgn}(\tilde{\ell}_{n,M1} - \tilde{\ell}_{n,M2}) \neq \text{sgn}(\tilde{\ell}_{n,M1} - \tilde{\ell}_{n,M2} - (\tilde{p}_{\eta,M1}^* - \tilde{p}_{\eta,M2}^*)), \quad (5)$$

where  $\tilde{p}_{\eta,Mj}^*$  indicates the bias correction term of model  $j$ , as in (3). Thus, if  $\tilde{p}_{\eta,M1}^* < \tilde{p}_{\eta,M2}^*$ , CIC chooses model 2 more often as winning model (relatively to  $AIC_{2ML}$ ). If the inequality holds in opposite direction, CIC chooses model 1 more often as winning model (again, relatively to  $AIC_{2ML}$ ). From Fig. 5, we see that both  $\tilde{p}_{\eta,M1}^*$  and  $\tilde{p}_{\eta,M2}^*$  have value close to 5, when the model is correctly specified, but as the misspecification appears,  $\tilde{p}_{\eta,M2}^*$  grows faster than  $\tilde{p}_{\eta,M1}^*$ . (This is also visible by the fact that the point where  $\tilde{p}_{\eta,M1}^*$ -curve and  $\tilde{p}_{\eta,M2}^*$ -curve cross each other lies far more right to  $\delta = 0.5$ .) Combining this growth speed difference with (5), it is logical that CIC chooses model 1 more often as winning model compared to  $AIC_{2ML}$ .

To sum up, CIC, compared to  $AIC_{2ML}$ , chooses the model with smaller value of  $\tilde{p}_{\eta}^*$  more often as winning model, which in practice is not necessarily more correct model as we noticed in Fig. 4.

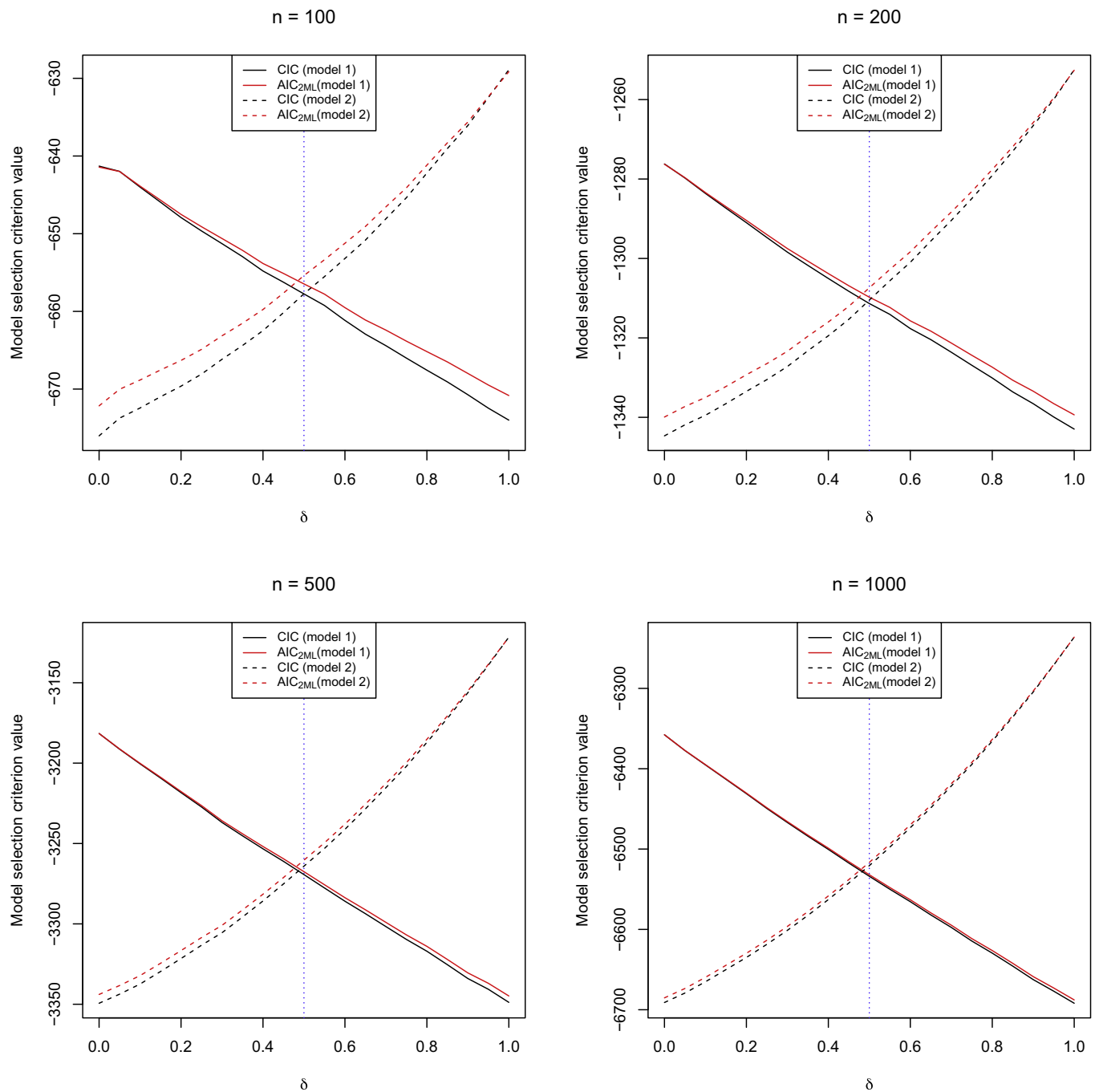


**Fig. 3.** Result of simulation 2. The y-axis is the value of CIC or AIC<sub>2ML</sub>. The x-axis is  $\delta$ , the proportion of data points generated by model 2 described in Table 4. The remaining  $(1 - \delta) \cdot 100\%$  data points were generated by model 1. The simulation was repeated 10000 times and the results were averaged. The blue dashed vertical line is  $\delta = 0.5$ . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Interestingly, there does exist a situation where, theoretically at least, the difference in  $\tilde{p}_\eta^*$  between two competing models always favors the correct model: When one of the two model is the 'true model'. i.e.  $\delta = 0$  or  $1$ . In this case, as one can see in Fig. 5,  $\tilde{p}_\eta^*$  of the correct model will be (almost) equal to  $p_\eta$ , while  $\tilde{p}_\eta^*$  of the misspecified model will outgrow  $p_\eta$ . So, there is no issue of growth speed difference. However, in this situation, the log-likelihood of the 'true model' will be much higher than that of the misspecified model. So, the AIC<sub>2ML</sub> would also be able to point the true model as winning model. (e.g. Winning rate close to 1 in Fig. 4 when  $\delta = 0$  or  $1$ .) So, in practice, there will be no noticeable advantage of CIC in this situation.

Furthermore, the fact that  $\tilde{p}_\eta^*$  gets lower than  $p_\eta$  in lower sample sizes is because of the estimation errors involved in estimating matrices as mentioned in Section 2.3 of Burnham and Anderson (2003).

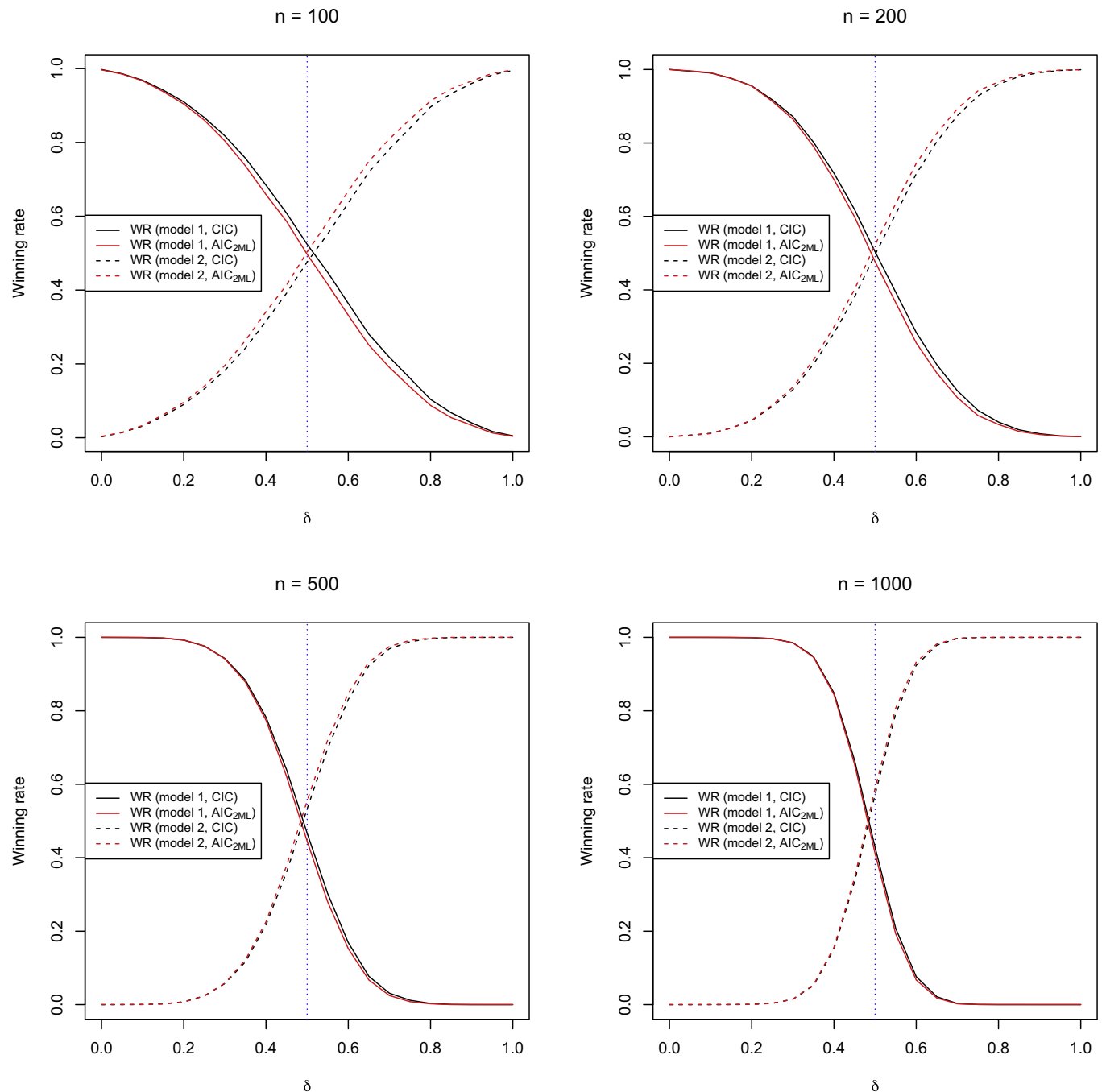


**Fig. 4.** Result of simulation 2. The y-axis is what we call 'winning rate (WR)', the proportion among 10,000 repetitions that the concerning model (model 1 or model 2) is picked as the best model according to the model selection criterion of choice (CIC or  $AIC_{2ML}$ ). The x-axis is  $\delta$ , the proportion of data points generated by model 2 described in Table 4. The remaining  $(1 - \delta) \cdot 100\%$  data points were generated by model 1. The blue dashed vertical line is  $\delta = 0.5$ . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

#### 4. Conclusions and further research

In this paper, we have developed the copula information criterion (CIC), which is a TIC-like model robust model selection criterion for two-stage ML estimated copulas. When we make an assumption that the parametric candidate model contains the true model, CIC becomes equal to  $AIC_{2ML}$ . This validates the use of  $AIC_{2ML}$  for the two-stage ML estimated copula models. To our knowledge, this is the first time that  $AIC_{2ML}$  formula is analytically justified. Further, since both TIC and CIC are estimating the same part of the KL divergence, without the presence of the true model assumption, they are compatible to each other and can be used to check possible disadvantages caused by the two-stage ML estimation. The same can be done by comparing  $AIC_{ML}$  and  $AIC_{2ML}$ , when one believes in the model.

Regarding the assumption that a candidate model is correct, one can compare  $\tilde{p}_\eta^*$  (bias correction term of CIC) and  $p_\eta$  (bias correction term of  $AIC_{2ML}$ ) to check whether the model severely diverges from the data generating model, i.e. as a

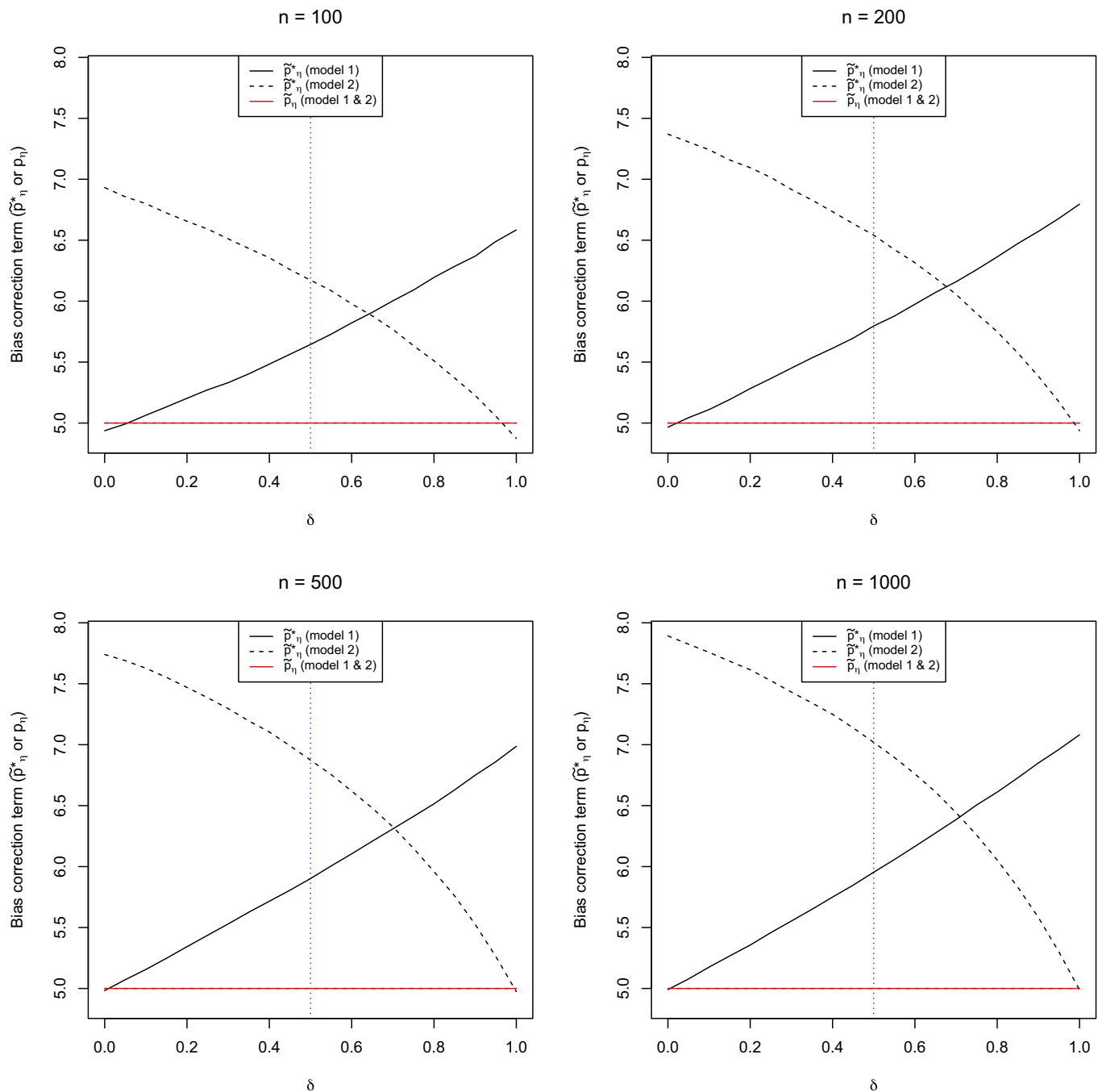


**Fig. 5.** Result of simulation 2. The y-axis is the value of bias correction term ( $\hat{p}_\eta^*$  for CIC and  $p_\eta$  for AIC<sub>2ML</sub>). The x-axis is  $\delta$ , the proportion of data points generated by model 2 described in Table 4. The remaining  $(1 - \delta) \cdot 100\%$  data points were generated by model 1. The simulation was repeated 10,000 times and the results were averaged. The blue dashed vertical line is  $\delta = 0.5$ . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

separate goodness-of-fit test. It may be noted that the job of the CIC is to rank models according to a sensible criterion, and to identify the best ones, but doing well in this ranking is not the same as claiming that the model passes goodness-of-fit tests. In yet other words, the winning model, using the CIC, may still not be a perfect model, perhaps since the list of candidate models has not been the best.

We performed a set of simulation studies. In the first simulation study, we observed that CIC and AIC<sub>2ML</sub> have strong agreement in how they rank the model and that the 'better models' according to CIC and AIC<sub>2ML</sub> show better prediction performance (measured in MSE). In addition,  $\hat{p}_\eta^*$  alone has a strong correlation with the prediction performance. So, one could consider to utilize  $\hat{p}_\eta^*$  as an extra tool to evaluate the model. (We showed earlier that  $\hat{p}_\eta^*$  can be used to measure the divergence of a candidate model from the true model.)

In the second simulation study, we showed that CIC and AIC<sub>2ML</sub> are indeed identical when the model is correctly specified. The two model selection criteria diverge from each other as the degree of misspecification gets larger. The difference

between CIC and  $AIC_{2ML}$  however becomes smaller and smaller as sample size increases, because the log-likelihood part grows much faster than the bias correction part.

Further, we showed that when CIC and  $AIC_{2ML}$  disagree on the winning model, CIC does not necessarily outperform  $AIC_{2ML}$ . CIC has tendency to prefer a model that shows slow growth  $\tilde{p}_\eta^*$  of when the degree of model misspecification increases. Further, CIC should theoretically outperform  $AIC_{2ML}$  in the situation when there is true model on the candidate model list. However, in practice, this effect is shadowed by the log-likelihood term which dominates the criterion because the fit of the true model is much better than the misspecified model. In this regard,  $AIC_{2ML}$  already picks the true model as the best model and there is almost no space for CIC to outperform  $AIC_{2ML}$ . Hence, no real practical advantage of CIC over  $AIC_{2ML}$ . Moreover, CIC requires large sample sizes to estimate its bias correction elements accurately and when sample size is large, the dominance of log-likelihood term gets even larger. In this sense,  $AIC_{2ML}$  is a parsimonious approach to CIC. These practical issues are also the reason why TIC is rarely applied in practice for ML estimated models (Burnham and Anderson, 2003).

Because of the large number of possible models in high dimensional settings, the number of situations that we could examine was limited. (In case of a 4-dimensional copula model with 5 candidate copulas and 3 candidate margins for each variable, there are 324 models that we have to test.) Another problem was that we could not try all copulas and margins on the simulated data since fitting a heavily misspecified copula and margins would cause numerical problems. A further large-scale simulation study that examines the behavior of different types of copulas in variety of situations would be fruitful.

Furthermore, CIC is computationally expensive mainly because  $\tilde{K}_\alpha$ ,  $\tilde{K}_{\alpha,\theta}$  and  $\tilde{K}_\theta$  require score functions for every data point separately. For example, for  $\tilde{I}_\alpha^{-1}$  and  $\tilde{I}_\theta$ , one can avoid this by swapping the order of summation and differentiation, but for  $\tilde{K}_\alpha$ ,  $\tilde{K}_{\alpha,\theta}$  and  $\tilde{K}_\theta$ , this is not possible. A numerical technique that can make CIC less computationally extensive would be appreciated.

Theoretically, CIC can be directly applied to pair-copula constructions as long as one uses two-stage ML for estimation. A pair-copula construction is after all just a special case of high-dimensional copula (Aas et al., 2009). However, in practice, it can be difficult to estimate all the necessary components due to the large number of parameters.

Although CIC and  $AIC_{2ML}$  perform decently well in selecting a good model that fits the data best in terms of KL divergence, there are situations where one is interested in a model that is suitable for specific tasks. The task of interest could be for example estimating tail probabilities, the mean, or the median. The authors of this study has recently developed a model selection criterion that can take this into account, for copula models under the two-stage ML scheme (Ko et al., 2019).

## Acknowledgments

The authors would like to thank Steffen Grønneberg and Ingrid Hobæk Haff for their valuable comments and fruitful discussions. The authors also acknowledge partial funding from the Norwegian Research Council supported research group FocuStat: Focus Driven Statistical Inference With Complex Data, and from the Department of Mathematics at the University of Oslo. We are also grateful to the editor and reviewers for constructive comments which helped improve our presentation.

## References

- Aas, K., Czado, C., Frigessi, A., Bakken, H., 2009. Pair-copula constructions of multiple dependence. *Insur. Math. Econ.* 44 (2), 182–198.
- Acar, E.F., Craiu, R.V., Yao, F., 2013. Statistical testing of covariate effects in conditional copula models. *Electron. J. Stat.* 7, 2822–2850.
- Akaike, H., 1974. A new look at the statistical model identification. *IEEE Trans. Autom. Control* 19 (6), 716–723.
- Akaike, H., 1998. Information theory and an extension of the maximum likelihood principle. In: *Selected Papers of Hirotugu Akaike*. Springer, pp. 199–213.
- Burnham, K.P., Anderson, D.R., 2003. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer Science & Business Media.
- Claeskens, G., Hjort, N.L., 2008. *Model Selection and Model Averaging*. Cambridge University Press Cambridge.
- Genest, C., Favre, A.-C., 2007. Everything you always wanted to know about copula modeling but were afraid to ask. *J. Hydrol. Eng.* 12 (4), 347–368.
- Grønneberg, S., Hjort, N.L., 2014. The copula information criteria. *Scand. J. Stat.* 41 (2), 436–459.
- Joe, H., 1997. *Multivariate Models and Multivariate Dependence Concepts*. CRC Press.
- Jullum, M., Hjort, N.L., 2017. Parametric or nonparametric: the FIC approach. *Statistica Sinica* 27, 951–981.
- Ko, V., Hjort, N.L., 2019. Model robust inference with two-stage maximum likelihood estimation for copulas. *J. Multivar. Anal.* <https://www.sciencedirect.com/science/article/pii/S0047259X18302495>.
- Ko, V., Hjort, N.L., Hobæk Haff, I., 2019. Focused information criterion for copulas. *Scandinavian Journal of Statistics*. (in press).
- Kullback, S., Leibler, R.A., 1951. On information and sufficiency. *Ann. Math. Stat.* 22 (1), 79–86.
- Nelsen, R.B., 2006. *An Introduction to Copulas*. Springer Science & Business Media.
- Palaro, H., Hotta, L., 2006. Using conditional copula to estimate value at risk. *J. Data Sci.* 4 (1), 93–115.
- Patton, A.J., 2002. *Applications of Copula Theory in Financial Econometrics*. University of California, San Diego Ph.D. thesis.
- Patton, A.J., 2006. Modelling asymmetric exchange rate dependence. *Int. Econ. Rev.* 47 (2), 527–556.
- Sklar, A., 1959. Fonctions de répartition à n dimensions et leurs marges. *Publ. Inst. Statist. Univ. Paris* 8, 229–231.
- Takeuchi, K., 1976. The distribution of information statistics and the criterion of goodness of fit of models. *Math. Sci.* 153, 12–18.





# Focused information criteria for copulas

Vinnie Ko  | Nils Lid Hjort | Ingrid Hobæk Haff

Department of Mathematics,  
University of Oslo, Oslo, Norway

## Correspondence

Vinnie Ko, Department of  
Mathematics, University of Oslo, PB  
1053, Blindern, 0316 Oslo, Norway.  
Email: vinniebk@math.uio.no

## Funding information

Norwegian Research Council;  
FocuStat (Focus Driven Statistical  
Inference With Complex Data)

## Abstract

In this paper, we extend the focused information criterion (FIC) to copula models. Copulas are often used for applications where the joint tail behavior of the variables is of particular interest, and selecting a copula that captures this well is then essential. Traditional model selection methods such as the Akaike information criterion (AIC) and the Bayesian information criterion (BIC) aim at finding the overall best-fitting model, which is not necessarily the one best suited for the application at hand. The FIC, on the other hand, evaluates and ranks candidate models based on the precision of their point estimates of a context-given focus parameter. This could be any quantity of particular interest, for example, the mean, a correlation, conditional probabilities, or measures of tail dependence. We derive FIC formulae for the maximum likelihood estimator, the two-stage maximum likelihood estimator, and the so-called pseudo-maximum-likelihood (PML) estimator combined with parametric margins. Furthermore, we confirm the validity of the AIC formula for the PML estimator combined with parametric margins. To study the numerical behavior of FIC, we have carried out a simulation study, and we have also analyzed a multivariate data set pertaining to abalones. The results from the study show that the FIC successfully ranks candidate models in terms of their performance, defined as how well they estimate the focus parameter. In terms of estimation precision, FIC clearly outperforms AIC, especially when the focus

parameter relates to only a specific part of the model, such as the conditional upper-tail probability.

#### KEYWORDS

mean square error, model selection, semiparametric estimation, two-stage maximum likelihood

## 1 | INTRODUCTION AND COPULA MODELS

Model selection is inarguably an important part of modern statistics. Most model selection criteria work by evaluating the “overall” fit of the models, in a suitable sense, as opposed to their performance related to a specific use. For example, the Akaike information criterion (AIC) and the Takeuchi's information criterion (TIC) aim for the model that minimizes the Kullback–Leibler (KL) divergence from the real data-generating mechanism to the candidate model. However, in practice, the model itself is often not the final goal. The model is to be used for some specific tasks such as estimating the mean, a quantile, correlation, or the tail dependence.

Copula models are frequently used for applications within, for instance, finance and insurance, as well as hydrology, where the tail behavior of the joint distribution is particularly important. Measures of interest are then typically the value at risk, the expected shortfall, or the conditional upper quantiles of the distribution of sea and river levels. Selecting a copula that captures the joint tail behavior well is then essential. It is by no means certain that this is obtained with traditional model selection methods. On the other hand, the focused information criterion (FIC), proposed by Claeskens and Hjort (2003), is a model selection method that evaluates candidate models based on the precision of their estimate of a quantity of interest, measured in terms of the mean square error (MSE). This quantity of interest is called the “focus parameter” and hence the name “focused information criterion.” One of the drawbacks of the original FIC (Claeskens & Hjort, 2003) is that it assumes all the models to be parametric and nested. Jullum and Hjort (2017) extend the FIC to a more general setting by using a nonparametric model to estimate the bias part of the decomposed MSE. Their new FIC machinery is designed for maximum likelihood (ML)-estimated parametric models and nonparametric models. Furthermore, the candidate models do not have to be nested. We extend this new FIC to copula models that are estimated with ML, two-stage ML, or pseudo ML combined with parametric margins (PMLpm).

Our technical setting is as follows. Let  $y_1, \dots, y_n$  be independent and identically distributed  $d$ -dimensional observations from a joint density  $g(y_1, \dots, y_d)$ . This data-generating model  $g$  is typically unknown. Suppose  $f(y_1, \dots, y_d, \eta)$  is our parametric approximation of  $g$ , with the parameter vector  $\eta$  belonging to some connected subset of the appropriate Euclidean space. Let  $G$  and  $F(\cdot, \eta)$  denote cumulative distribution functions corresponding to  $g$  and  $f(\cdot, \eta)$ , respectively. Furthermore,  $G_j(y_j)$  and  $F_j(y_j, \alpha_j)$  indicate  $j$ th marginal distribution functions corresponding to  $G$  and  $F(\cdot, \eta)$ , respectively,  $\alpha_j$  being the parameter vector belonging to margin component  $j$ .

According to Sklar's theorem (Sklar, 1959), there always exists a copula  $C(u_1, \dots, u_d, \theta)$  that satisfies

$$F(y_1, \dots, y_d, \eta) = C(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta)$$

with the full parameter vector

$$\eta = (\alpha^T, \theta^T)^T = (\alpha_1^T, \dots, \alpha_d^T, \theta^T)^T.$$

Under the assumption that  $F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d)$  are absolutely continuous and strictly increasing,  $C(\cdot, \theta)$  can be differentiated. This gives

$$f(y_1, \dots, y_d, \eta) = c(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta) \prod_{j=1}^d f_j(y_j, \alpha_j),$$

where  $c(u_1, \dots, u_d) = \partial^d C(u_1, \dots, u_d, \theta) / \partial u_1 \dots \partial u_d$  and  $f_j(y_j, \alpha_j) = \partial F_j(y_j, \alpha_j) / \partial y_j$ . Similarly, the true density  $g$  can also be decomposed into marginal densities and copula density

$$g(y_1, \dots, y_d) = c_0(G_1(y_1), \dots, G_d(y_d)) \prod_{j=1}^d g_j(y_j),$$

with  $c_0(\cdot)$  the density of the true copula. For more details on copula modeling, see Joe (1997, 2014) and Nelsen (2006).

Three of the most commonly used estimation procedures for parametric copulas are full ML, two-stage ML, and PML. When full ML is employed for the copula and margins, the theory of AIC, BIC, and FIC holds directly. For PML-estimated copula, combined with nonparametric margins, Grønneberg and Hjort (2014) develop a copula information criterion (CIC) and Chen and Fan (2005) suggest a pseudolikelihood ratio test. Ko and Hjort (2019a) propose a different CIC for the two-stage ML. This paper concerns extending the FIC theory to these different estimation procedures. This partly follows the line of arguments used in Jullum and Hjort (2017, 2018) but, by necessity, involves certain extra efforts, as we work with more complex estimation strategies than the pure nonparametric and the ML.

The further structure of this paper is as follows. In Section 2, the FIC for copula models under ML, two-stage ML, and the PMLpm estimator is derived and defined. Furthermore, we present the AIC formula for the PMLpm estimator in Section 3 and cover the average weighted FIC (AFIC), designed for the situations where multiple focus parameters are of interest, in Section 4. In Section 5, the behavior of the FIC is investigated in a simulation study. Moreover, we apply the FIC to a data set on multiple measurements on abalones in Section 6. Finally, Section 7 contains concluding remarks and suggestions for future work.

## 2 | FIC FOR COPULA MODELS

### 2.1 | General idea and derivation of the FIC

In this subsection, we briefly summarize the idea behind the FIC of Jullum and Hjort (2017) and reach a general formula. The actual details for implementing this FIC formula depend on the specific estimation scheme, which we return to later in Sections 2.2, 2.3, and 2.4.

Most model selection criteria such as AIC, BIC, and TIC measure how close a candidate model is to the true model in terms of a certain divergence measure. However, in many cases, the model itself is not the goal, but rather a means to estimate a specific quantity, such as the mean or the probability of a certain event. The main idea behind the FIC, first introduced by Claeskens and Hjort (2003) and later modified by Jullum and Hjort (2017, 2018), is that one wants to select a model that is good for a specific task. In our setting, we quantify this specific task as that of estimating a focus parameter, say,  $T(G)$ , a functional of the data-generating distribution. The functional needs to have a clear statistical interpretation across candidate models. With  $F_M$ , the distribution associated with such a candidate model  $M$ ,  $T(F_M)$  is the ensuing approximation to

the  $T_{\text{true}} = T(G)$ . We also need the functional to be smooth, in the sense of admitting an influence function leading to limiting normality with a  $\sqrt{n}$  rate. This involves certain regularity conditions, as (C1)–(C4) of Jullum and Hjort (2017, Section 2).

In our setting, the estimated focus parameter via the use of a candidate model is denoted as  $\hat{T}_{\text{cnd}}$  or  $T(\hat{F})$ , with  $\hat{F}$  as the estimated distribution via this model. We are looking for candidate models such that  $\hat{T}_{\text{cnd}}$  is close to  $T_{\text{true}}$ . As with Jullum and Hjort (2017, 2018), we use a quadratic loss function, leading by definition to the MSE as the criterion of closeness. We hence define the FIC as the estimated MSE

$$\text{FIC} = \widehat{\text{MSE}}(\hat{T}_{\text{cnd}}) = \widehat{\text{Var}}_G(\hat{T}_{\text{cnd}}) + \widehat{\text{bias}}^2(\hat{T}_{\text{cnd}}, T_{\text{true}}). \quad (1)$$

We return to estimates for the variance of  $\hat{T}_{\text{cnd}}$  below, with different recipes for different estimators. For estimating the bias, we use the fact that the nonparametrically estimated  $T$ , which we denote as  $\hat{T}_{\text{np}}$ , is asymptotically unbiased. This suggests using

$$\widehat{\text{bias}}^{2*}(\hat{T}_{\text{cnd}}, T_{\text{true}}) = (\hat{T}_{\text{cnd}} - \hat{T}_{\text{np}})^2,$$

which however tends to overestimate its estimand. By correcting for the overestimation and possible negative values, we obtain the final squared bias estimator

$$\widehat{\text{bias}}^2(\hat{T}_{\text{cnd}}, T_{\text{true}}) = \begin{cases} \max \left\{ 0, (\hat{T}_{\text{cnd}} - \hat{T}_{\text{np}})^2 - \widehat{\text{Var}}_G(\hat{T}_{\text{cnd}}) \right. \\ \left. - \widehat{\text{Var}}_G(\hat{T}_{\text{np}}) + 2\widehat{\text{Cov}}_G(\hat{T}_{\text{np}}, \hat{T}_{\text{cnd}}) \right\}, & \text{if the model is (semi)parametric} \\ 0, & \text{if the model is nonparametric.} \end{cases} \quad (2)$$

Note that this derivation of the FIC is done without specifying the estimation scheme. As long as  $\hat{T}_{\text{cnd}}$ ,  $\hat{T}_{\text{np}}$ ,  $\widehat{\text{Var}}_G(\hat{T}_{\text{cnd}})$ ,  $\widehat{\text{Var}}_G(\hat{T}_{\text{np}})$ , and  $\widehat{\text{Cov}}_G(\hat{T}_{\text{np}}, \hat{T}_{\text{cnd}})$  can be obtained, the FIC can be defined for any model and estimation scheme.

## 2.2 | FIC under ML

The ML estimator  $\hat{\eta}^{\text{ML}}$  is defined as the maximizer of  $\ell_n(\eta)$ , the log-likelihood function of a chosen model with given observations. Its properties are covered in classic references such as White (1982) and Le Cam (1990).

Let  $\hat{T}_{\text{ML}} = T(F(\hat{\eta}^{\text{ML}}))$  denote a focus parameter, estimated based on a given parametric model and using the ML. Furthermore, let  $T_{0,\text{ML}}$  indicate the focus parameter associated with the least false parameter value  $\eta_0^{\text{ML}}$ , which minimizes the KL divergence from the true model to the candidate model. If the candidate model captures the true model, we have  $T_{0,\text{ML}} = T_{\text{true}}$ .

Under the regularity conditions (C1)–(C4) (Jullum & Hjort, 2017), the joint distribution of  $\hat{T}_{\text{ML}}$  and  $\hat{T}_{\text{np}}$  can be written in our copula setting as

$$\sqrt{n} \begin{pmatrix} \hat{T}_{\text{np}} - T_{\text{true}} \\ \hat{T}_{\text{ML}} - T_{0,\text{ML}} \end{pmatrix} \xrightarrow{d} \begin{pmatrix} \Lambda_{\text{np}} \\ \dot{s}_{\eta}^{\text{T}} (\mathcal{I}_{\eta}^{\text{ML}})^{-1} \Lambda_{\eta^{\text{ML}}} \end{pmatrix} \sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} V_{\text{np}} & V_{\text{np,ML}} \\ V_{\text{np,ML}}^{\text{T}} & V_{\text{ML}} \end{pmatrix} \right), \quad (3)$$

where

$$\begin{aligned}
 V_{np} &= \text{Var}_G(\text{IF}_T(y, G)) = E_G [\text{IF}(y, G) \text{IF}(y, G)^T], \\
 V_{np,ML} &= Q_{ML}(\mathcal{I}_\eta^{\text{ML}})^{-1} \dot{s}_\eta, \quad Q_{ML} = E_G \left[ \text{IF}_T(y, G) \cdot \phi_\eta^{\text{ML}}(y, \eta_0^{\text{ML}})^T \right], \quad \dot{s}_\eta = \frac{\partial s_\eta}{\partial \eta} = \frac{\partial T(F(\eta))}{\partial \eta}, \\
 V_{ML} &= \dot{s}_\eta^T (\mathcal{I}_\eta^{\text{ML}})^{-1} K_\eta^{\text{ML}} (\mathcal{I}_\eta^{\text{ML}})^{-1} \dot{s}_\eta, \quad K_\eta^{\text{ML}} = E_G \left[ \phi_\eta^{\text{ML}}(y, \eta_0^{\text{ML}}) \phi_\eta^{\text{ML}}(y, \eta_0^{\text{ML}})^T \right], \\
 \phi_\eta^{\text{ML}}(y, \eta) &= \frac{\partial \left( \sum_{j=1}^d \log f_j(y_j, \alpha_j) + \log c(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta) \right)}{\partial \eta}, \\
 \mathcal{I}_\eta^{\text{ML}} &= -E_G [H_\eta(y, \eta_0^{\text{ML}})] \\
 &= - \int g \frac{\partial^2 \left( \sum_{j=1}^d \log f_j(y_j, \alpha_{0,j}^{\text{ML}}) + \log c(F_1(y_1, \alpha_{0,1}^{\text{ML}}), \dots, F_d(y_d, \alpha_{0,d}^{\text{ML}}), \theta_0^{\text{ML}}) \right)}{\partial \eta \partial \eta^T} dy,
 \end{aligned}$$

and  $\text{IF}_T(y, G)$  is the influence function associated with the focus parameter. Based on (3), the FIC for ML estimated parametric models is defined as

$$\text{FIC}_{ML} = \frac{1}{n} \hat{V}_{ML} + \max \left\{ 0, \left( \hat{T}_{ML} - \hat{T}_{np} \right)^2 - \frac{1}{n} \left( \hat{V}_{ML} + \hat{V}_{np} - 2\hat{V}_{np,ML} \right) \right\}, \quad (4)$$

where

$$\hat{V}_{np,ML} = \hat{Q}_{ML}(\hat{\mathcal{I}}_\eta^{\text{ML}})^{-1} \hat{s}_{\hat{\eta}}, \quad \hat{V}_{ML} = \hat{s}_{\hat{\eta}}^T \hat{V}_\eta^{\text{ML}} \hat{s}_{\hat{\eta}} = \hat{s}_{\hat{\eta}}^T (\hat{\mathcal{I}}_\eta^{\text{ML}})^{-1} \hat{K}_\eta^{\text{ML}} (\hat{\mathcal{I}}_\eta^{\text{ML}})^{-1} \hat{s}_{\hat{\eta}}.$$

Here,  $\hat{V}_{np}$ ,  $\hat{Q}_{ML}$ ,  $\hat{\mathcal{I}}_\eta^{\text{ML}}$ ,  $\hat{K}_\eta^{\text{ML}}$  indicate empirical analogues of  $V_{np}$ ,  $Q_{ML}$ ,  $\mathcal{I}_\eta^{\text{ML}}$ ,  $K_\eta^{\text{ML}}$ , respectively.

### 2.3 | FIC under two-stage ML

When the dimension  $d$  of the model grows, the length of the total parameter vector  $\eta$  increases rapidly and ML estimation is not always feasible. Shih and Louis (1995) proposed two-stage ML (a.k.a. inference functions for margins) as an alternative. This consists in first estimating  $\alpha_j$  separately for each margin by ML. Then, one plugs the resulting  $\hat{\alpha}^{2ML}$  into the log-likelihood and obtains  $\hat{\theta}^{2ML}$  by maximizing the log-likelihood with respect to  $\theta$ . The asymptotic properties of the two-stage ML estimator are given in Ko and Hjort (2019b) for the general case and given in Joe (2005) for the situation where one assumes that the parametric model covers the true data-generating mechanism.

**Proposition 1.** Under regularity conditions (C1)–(C4) from Jullum and Hjort (2017) and (A1)–(A5) from Ko and Hjort (2019b),

$$\sqrt{n} \begin{pmatrix} \hat{T}_{np} - T_{\text{true}} \\ \hat{T}_{2ML} - T_{0,2ML} \end{pmatrix} \xrightarrow{d} \begin{pmatrix} \Lambda_{np} \\ \dot{s}_\eta^T (\mathcal{I}_\eta^{2ML})^{-1} \Lambda_{\eta^{2ML}} \end{pmatrix} \sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} V_{np} & V_{np,2ML} \\ V_{np,2ML}^T & V_{2ML} \end{pmatrix} \right),$$

where

$$\begin{aligned}
 V_{np,2ML} &= Q_\eta^{2ML} \left( (\mathcal{I}_\eta^{2ML})^{-1} \right)^T \dot{s}_\eta, \\
 V_{2ML} &= \dot{s}_\eta^T V_{\eta^{2ML}} \dot{s}_\eta, \\
 V_{\eta^{2ML}} &= (\mathcal{I}_\eta^{2ML})^{-1} K_\eta^{2ML} \left( (\mathcal{I}_\eta^{2ML})^{-1} \right)^T
 \end{aligned}$$

and other quantities are defined in the proof.

The proof is given in the Appendix.

In practice,  $V_{2ML}$  and  $V_{np,2ML}$  can be estimated by taking empirical analogues and using the plug-in principle. We write those estimates as  $\hat{V}_{2ML}$  and  $\hat{V}_{np,2ML}$ . The FIC for two-stage ML-estimated parametric models is then

$$FIC_{2ML} = \frac{1}{n} \hat{V}_{2ML} + \max \left\{ 0, \left( \hat{T}_{2ML} - \hat{T}_{np} \right)^2 - \frac{1}{n} \left( \hat{V}_{2ML} + \hat{V}_{np} - 2\hat{V}_{np,2ML} \right) \right\}. \quad (5)$$

## 2.4 | FIC under pseudo ML combined with parametric margins

One of the drawbacks of (two-stage) ML estimation is that the estimation of the copula parameter  $\theta$  is affected by the choice of parametric distribution for the margins. Even when the copula is correctly specified, possible misspecifications of the margins can lead to suboptimal values of  $\hat{\theta}^{ML}$  and  $\hat{\theta}^{2ML}$ . Genest, Ghoudi, and Rivest (1995) propose the PML estimator that overcomes this shortcoming. In this semiparametric estimation scheme, one first uses the pseudo-observations  $G_{n,j}(y_j) = (n+1)^{-1} \sum_{k=1}^n I\{y_{k,j} \leq y_j\}$ , which are nonparametric probability integral transforms of the data. Subsequently, one plugs these into the copula log-likelihood and maximizes this with respect to  $\theta$  to obtain  $\hat{\theta}^{PML}$ . For details concerning the PML estimator, see Tsukahara (2005), Chen and Fan (2005), and Kim, Silvapulle, and Silvapulle (2007).

Once  $\theta$  is estimated, one can keep the nonparametric margins. However, parametric margins are often used instead, estimating  $\alpha_j$  separately for each margin by ML, just like in stage 1 of two-stage ML estimation (Bevacqua, Maraun, Hobæk Haff, Widmann, & Vrac, 2017; Hobæk Haff, Frigessi, & Maraun, 2015). Below, we derive the FIC formula for this “PML combined with parametric margins” and abbreviate the estimation scheme by “PMLpm” in the rest of our paper.

**Proposition 2.** *Under regularity conditions (C1)–(C5) from Chen and Fan (2005) and (C1)–(C4) from Jullum and Hjort (2017),*

$$\sqrt{n} \begin{pmatrix} \hat{T}_{np} - T_{true} \\ \hat{T}_{PMLpm} - T_{0,PMLpm} \end{pmatrix} \xrightarrow{d} \begin{pmatrix} \Lambda_{np} \\ \dot{s}_\eta^T \left( \mathcal{I}_\eta^{PMLpm} \right)^{-1} \Lambda_\eta^{PMLpm} \end{pmatrix} \sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} V_{np} & V_{np,PMLpm} \\ V_{np,PMLpm}^T & V_{PMLpm} \end{pmatrix} \right),$$

where

$$\begin{aligned} V_{np,PMLpm} &= Q_\eta^{PMLpm} \left( \mathcal{I}_\eta^{PMLpm} \right)^{-1} \dot{s}_\eta, \\ V_{PMLpm} &= \dot{s}_\eta^T V_{\eta^{PMLpm}} \dot{s}_\eta, \\ V_{\eta^{PMLpm}} &= \left( \mathcal{I}_\eta^{PMLpm} \right)^{-1} K_\eta^{PMLpm} \left( \mathcal{I}_\eta^{PMLpm} \right)^{-1}, \end{aligned}$$

and other quantities are defined in the proof.

The proof is given in the Appendix.

In practice,  $V_{PMLpm}$  and  $V_{np,PMLpm}$  can be estimated by plug-in estimators. For  $K_\eta^{PMLpm}$  and  $\mathcal{W}(u, \theta)$ , one can use the consistent estimators suggested in Remark 2 of Chen and Fan (2005). The resulting estimated variance and covariance are written as  $\hat{V}_{PMLpm}$  and  $\hat{V}_{np,PMLpm}$ . Plugging these into (1) and (2) gives the FIC for PMLpm

$$FIC_{PMLpm} = \frac{1}{n} \hat{V}_{PMLpm} + \max \left\{ 0, \left( \hat{T}_{PMLpm} - \hat{T}_{np} \right)^2 - \frac{1}{n} \left( \hat{V}_{PMLpm} + \hat{V}_{np} - 2\hat{V}_{np,PMLpm} \right) \right\}. \quad (6)$$

### 3 | AIC FOR PSEUDO ML WITH PARAMETRIC MARGINS

Because the eventual model form of PMLpm is fully parametric, we can use the KL divergence to evaluate the relative closeness of the candidate model to the true model. Minimizing KL divergence is equal to maximizing

$$Q(\eta) = \int g(y) \{ \log f_1(y_1, \alpha_1) + \cdots + \log f_d(y_d, \alpha_d) + \log c(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta) \} dy.$$

The empirical equivalent is

$$\hat{Q}(\eta) = \frac{1}{n} \sum_{i=1}^n [ \log f_1(y_{i,1}, \alpha_1) + \cdots + \log f_d(y_{i,d}, \alpha_d) + \log c(F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta) ],$$

which is biased. AIC is defined as the bias-corrected and rescaled estimator of  $Q(\eta)$  under the true model assumption. We may now derive AIC for the PMLpm-estimated parametric model.

**Lemma 1.** *Assuming that the parametric model captures the model that generated the data, that is,  $f(\cdot, \eta_0) = g(\cdot)$  for an appropriate  $\eta_0$ , an approximately unbiased empirical estimator of  $2n \cdot Q(\hat{\eta}^{\text{PMLpm}})$ , widely known as AIC, is*

$$\text{AIC} = 2\ell_n(\hat{\eta}^{\text{PMLpm}}) - 2 \dim(\eta).$$

The proof is given in the Appendix.

Because the AIC for the ML, two-stage ML, and PMLpm estimators are all aiming for the same quantity under the model conditions, they are compatible. This means that AIC can be used to compare closeness to the true model for different estimation schemes.

### 4 | AVERAGE WEIGHTED FIC

The FIC apparatus is designed to select the best model for estimating a chosen focus parameter precisely. However, there are cases where one is interested in multiple focus parameters. Jullum and Hjort (2017) developed an averaged weighted FIC (AFIC), which aims for the model that obtains lowest risk for a set of focus parameters  $T_{\text{true}}(t)$ , with  $t$  in some index set. This set of focus parameters is specified along with a cumulative weight function  $W(t)$ . Under this setting, they use the weighted quadratic loss function

$$L = \int \left( \hat{T}_{\text{cnd}}(t) - T_{\text{true}}(t) \right)^2 dW(t),$$

which results in the following risk:

$$E_G[L] = \int E_G \left[ \left( \hat{T}_{\text{cnd}}(t) - T_{\text{true}}(t) \right)^2 \right] dW(t).$$

Then, they define the AFIC as

$$\text{AFIC} = \int \widehat{\text{Var}}_G \left( \hat{T}_{\text{cnd}}(t) \right) dW(t) + \widehat{\text{bias}}_{\text{cnd, true}}^2, \quad (7)$$

where the bias term is defined as

$$\widehat{\text{bias}}_{\text{cnd,true}}^2 = \begin{cases} \max \left\{ 0, \int \left( \widehat{T}_{\text{cnd}}(t) - \widehat{T}_{\text{np}}(t) \right)^2 - \widehat{\text{Var}}_G \left( \widehat{T}_{\text{cnd}}(t) \right) \right. \\ \left. - \widehat{\text{Var}}_G \left( \widehat{T}_{\text{np}}(t) \right) + 2\widehat{\text{Cov}}_G \left( \widehat{T}_{\text{np}}(t), \widehat{T}_{\text{cnd}}(t) \right) \right\} dW(t), & \text{if (semi)parametric} \\ 0, & \text{if nonparametric.} \end{cases} \quad (8)$$

Note that when the set of estimands  $T_{\text{true}}(t)$  is finite and the weight function is not stochastic, the integral with respect to  $W(t)$  simply becomes a weighted sum. For a practical illustration of AFIC, see the supplementary material of Jullum and Hjort (2017).

## 5 | SIMULATION STUDY

To study the behavior of FIC, we have carried out a simulation study in two parts. The performance of the FIC with a joint box probability as focus parameter is assessed in the Part 1 simulations. The purpose of the Part 2 simulations is to evaluate the FIC with other focus parameters, more specifically conditional probabilities, as well as the AFIC.

### 5.1 | Part 1: Evaluation of the FIC

We have generated data sets from a model consisting of a Gumbel copula; two Gamma, one Weibull, and one log-normal margins; and six fitted candidate models for each data set using the ML, two-stage ML, and PMLpm estimators. Furthermore, the sample size is either  $n = 100$  or  $n = 1000$ . The data-generating and candidate models are described in Table 1. With each fitted candidate model, we have computed the focus parameter  $T(G) = P(q_{0.8} < Y)$ , where  $q_{0.8}$  is a vector that contains the 0.8-quantile value of each margin according to the true model. That is,  $q_{0.8} = (G_1^{-1}(0.8), \dots, G_4^{-1}(0.8))$ . In addition, we have estimated the focus parameter nonparametrically, which is necessary for the FIC machinery. We have repeated this process 100 times and the results are averaged. Because the natural empirical estimator for  $\mathcal{W}$  from Chen and Fan (2005), needed for PMLpm, is computationally very expensive for a large sample size, we have estimated the asymptotic variance from Proposition 2 numerically with the jackknife (Efron, 1982). We have

**TABLE 1** Description of the models used in Part 1 of the simulations

|                       | Copula                 | Margin 1                                        | Margin 2                                        | Margin 3                                           | Margin 4                                             |
|-----------------------|------------------------|-------------------------------------------------|-------------------------------------------------|----------------------------------------------------|------------------------------------------------------|
| Data generating model | Gumbel<br>$\theta = 3$ | Gamma<br>$\alpha_1 = (1, 2)^T$<br>(shape, rate) | Gamma<br>$\alpha_2 = (3, 1)^T$<br>(shape, rate) | Weibull<br>$\alpha_3 = (1, 2)^T$<br>(shape, scale) | Log-normal<br>$\alpha_4 = (-1, 0.5)^T$<br>(mean, sd) |
| Model 1               | Gumbel                 | Gamma                                           | Gamma                                           | Weibull                                            | Log-normal                                           |
| Model 2               | Gumbel                 | Weibull                                         | Weibull                                         | Gamma                                              | Gamma                                                |
| Model 3               | Survival Clayton       | Gamma                                           | Gamma                                           | Weibull                                            | Log-normal                                           |
| Model 4               | Survival Clayton       | Weibull                                         | Weibull                                         | Gamma                                              | Gamma                                                |
| Model 5               | Frank                  | Gamma                                           | Gamma                                           | Weibull                                            | Log-normal                                           |
| Model 6               | Frank                  | Weibull                                         | Weibull                                         | Gamma                                              | Gamma                                                |

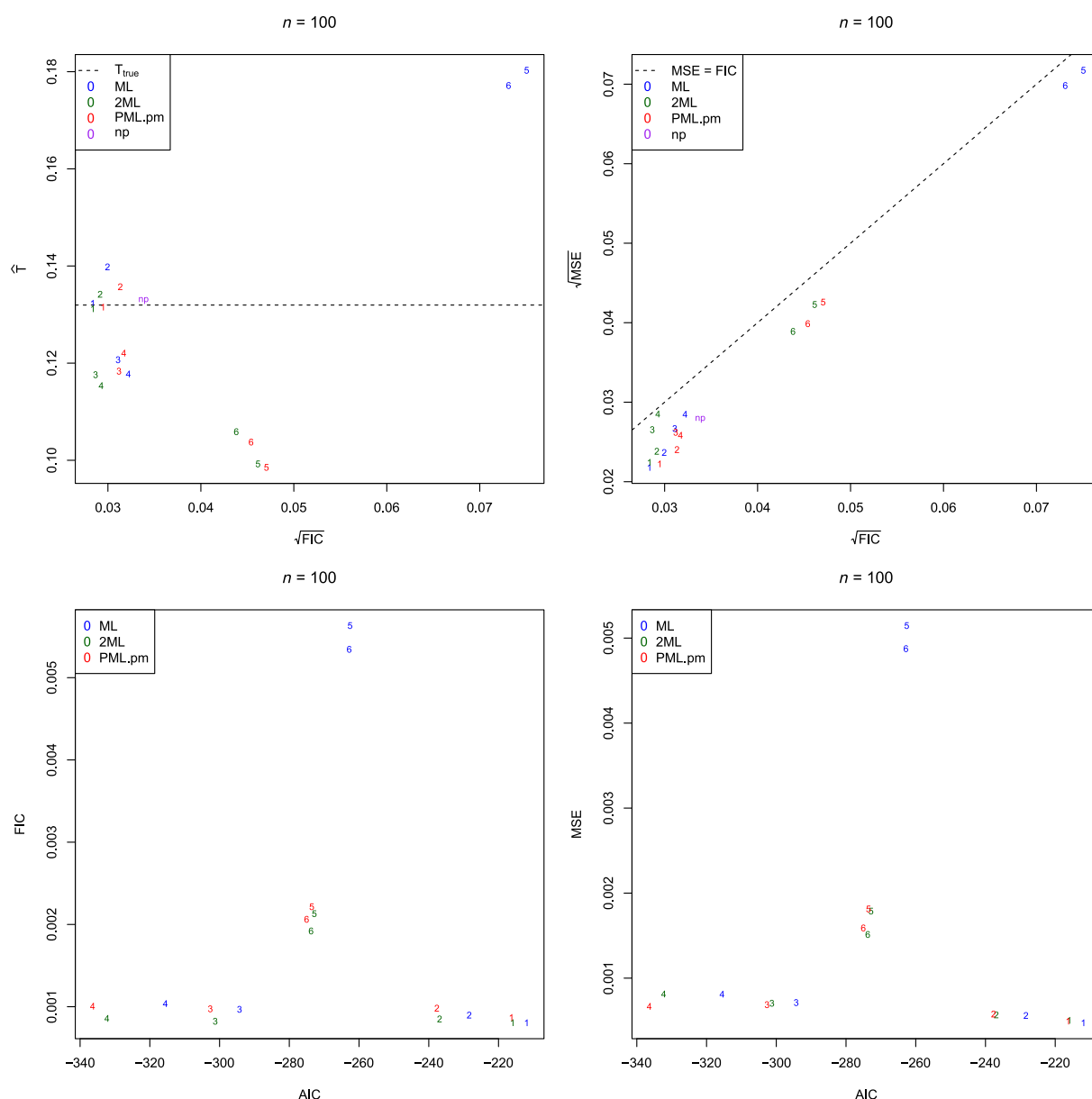


**FIGURE 1** Results from Part 1 of the simulations for  $n = 100$ . The colored numbers refer to the model numbers in Table 1. The nonparametric model is denoted by “np.” ML = maximum likelihood; PML = pseudo maximum likelihood; MSE = mean square error; FIC = focused information criterion; AIC = Akaike information criterion [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

compared the jackknife-estimated variance to the estimate on the basis of Chen and Fan (2005) for small sample sizes, and they produced comparable results.

The results from the simulations can be found in Tables 1 and 2 from the supplementary material (Ko, Hjort, & Hobæk Haff, 2019). The result is also visualized in Figure 1 for  $n = 100$  and in Figure 2 for  $n = 1000$ .

The top left panel of Figure 1 is a plot of the estimated focus parameter  $\hat{T}$  against  $\sqrt{\text{FIC}}$ , which is on the same scale as  $\hat{T}$ . We see that the models with smaller  $\sqrt{\text{FIC}}$  value tend to give focus parameter estimates that are closer to the true focus parameter value. This tendency becomes clearer as the sample size increases (see Figure 2).

In the top right panel of Figure 1, we have plotted  $\sqrt{\text{MSE}}$  against  $\sqrt{\text{FIC}}$ . We can observe that the FIC provides a good estimate of the MSE. As the sample size increases (Figure 2), the correspondence is even better.

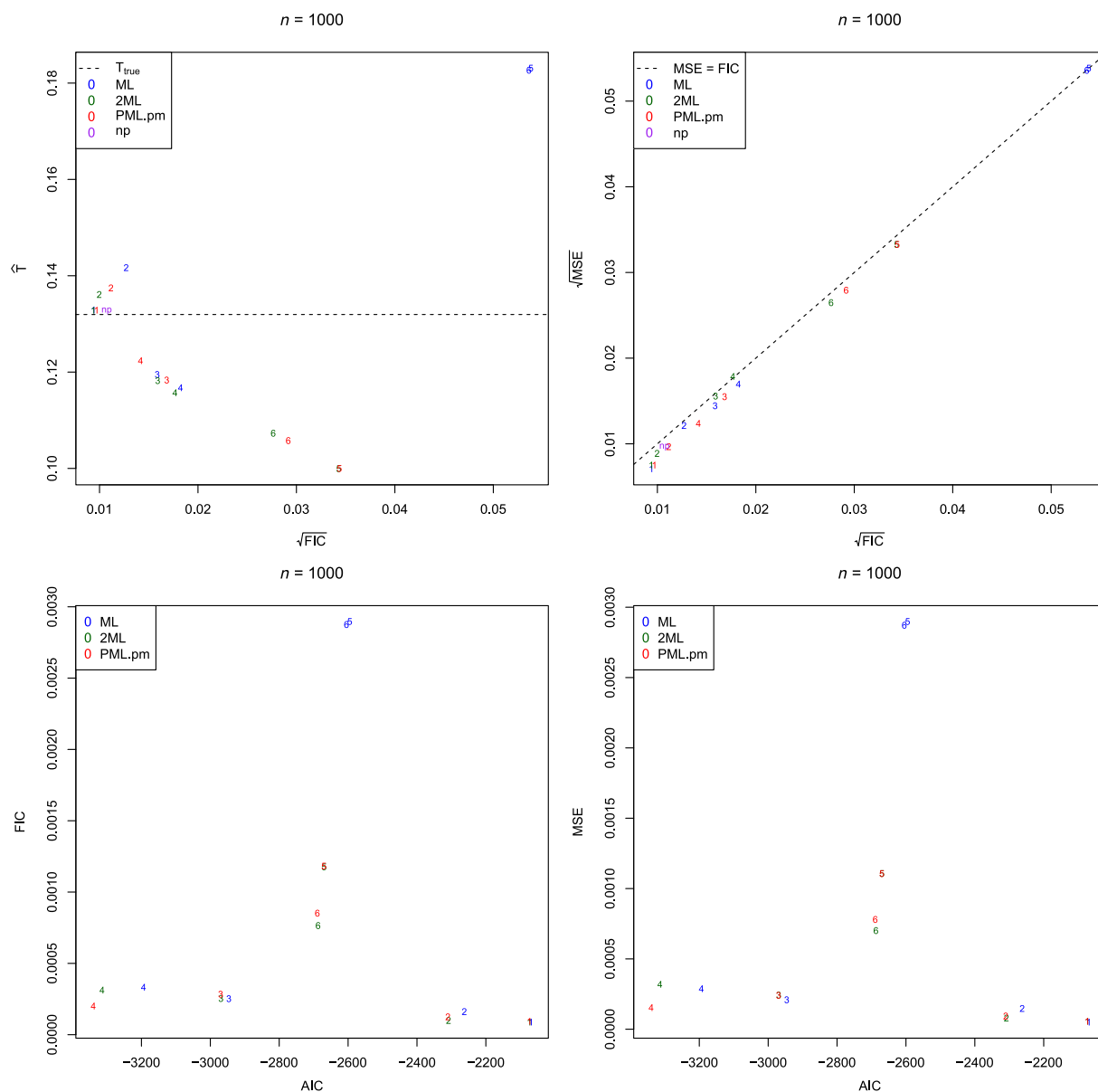


**FIGURE 2** Results from Part 1 of the simulations for  $n = 1000$ . See Figure 1 for an explanation of the symbols. ML = maximum likelihood; PML = pseudo maximum likelihood; MSE = mean square error; FIC = focused information criterion; AIC = Akaike information criterion [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

The two panels at the bottom of Figure 1 show the AIC plotted against the FIC (left panel) and the MSE (right panel) for  $n = 100$ . Note that models with large scores are preferred when using the AIC. We see that both the AIC and the FIC pick the correctly specified model (model 1) as the best model. The AIC rank for models 5 and 6 implies that those are overall rather close to the true model (relatively to other candidates). However, the FIC suggests that they do not estimate  $P(q_{0.8} < Y)$  well (again in comparison to the other candidate models). The estimated focus parameter values and MSEs indicate that this is indeed the case. This illustrates that the FIC does choose models that are good at a specific task, as intended. Figure 2 shows that the same holds when the sample size increases.

One of the noticeable changes as the sample size increases from  $n = 100$  to  $n = 1000$  is that the rank of the nonparametric model decreases from 13 to 5. The reason is that the (co)variance terms shrink toward 0 such that the asymptotic unbiasedness of the nonparametric

model becomes a stronger advantage. When  $n$  is large enough, the (co)variance terms will be negligible and the nonparametric model will be the overall “winning” model.

Furthermore, the focus parameter values from models estimated with two-stage ML and PMLpm tend to be better than the ones obtained with ML when the model is misspecified, with correspondingly lower FIC scores. The differences are less systematic when we compare two-stage ML and PMLpm.

## 5.2 | Part 2: Evaluation of the AFIC

In Part 2 of the simulations, we have generated data sets from a two-dimensional model consisting of a Gaussian copula and log-normal margins and fitted 12 candidate models by using ML, two-stage ML, and PMLpm estimators. The data-generating and candidate models are described in Table 2, and the sample size is  $n = 1000$ . With each fitted candidate model, we have computed a set of focus parameters  $T(F) = P(G_1^{-1}(p) < Y_1 \mid G_2^{-1}(p) < Y_2)$  for  $p = 0.90, 0.91, \dots, 0.95$ . When  $p$  is close to 1, a reasonable precision of the nonparametric estimate of this conditional probability requires a very large sample size to ensure that there are enough observations that satisfy the inequality conditions. Therefore, we did not include probabilities that are even closer to 1. In addition, we have computed the AFIC by assigning equal weights to each  $p$ . The whole process was repeated 100 times and the results are averaged. As for the Part 1 simulations, the asymptotic variance of PMLpm is estimated with jackknife.

The results from the simulations with the AFIC, as well as with the FIC with a single focus parameter for  $p = 0.90$  and  $p = 0.95$ , can be found in Tables 3, 4, and 5 from the supplementary material (Ko et al., 2019). The results are also visualized in Figures 3 and 4, together with Figure 1 from the supplementary material (Ko et al., 2019). The top left panel of Figure 3 is a plot of  $\sqrt{\text{FIC}}$  against the focus parameter estimates  $\hat{T}$  for the FIC with  $p = 0.9$ , and the top right panel shows  $\sqrt{\text{FIC}}$  against  $\sqrt{\text{MSE}}$ . The bottom left and right panels display AIC against FIC and AIC against MSE, respectively. Figures 4 show corresponding results for the FIC with  $p = 0.95$ . The corresponding result for AFIC is displayed in Figure 1 from the supplementary material (Ko et al., 2019).

**TABLE 2** Description of the models used in Part 2 of the simulations

|                       | Copula                     | Margin 1                                                    | Margin 2                                                      |
|-----------------------|----------------------------|-------------------------------------------------------------|---------------------------------------------------------------|
| Data generating model | Gaussian<br>$\theta = 0.6$ | Log-normal<br>$\alpha_1 = (1, 0.8)^T$<br>(mean, <i>SD</i> ) | Log-normal<br>$\alpha_2 = (0.4, 0.7)^T$<br>(mean, <i>SD</i> ) |
| Model 1               | Gaussian                   | Log-normal                                                  | Log-normal                                                    |
| Model 2               | Gaussian                   | Gamma                                                       | Gamma                                                         |
| Model 3               | Gaussian                   | Weibull                                                     | Weibull                                                       |
| Model 4               | Frank                      | Log-normal                                                  | Log-normal                                                    |
| Model 5               | Frank                      | Gamma                                                       | Gamma                                                         |
| Model 6               | Frank                      | Weibull                                                     | Weibull                                                       |
| Model 7               | Gumbel                     | Log-normal                                                  | Log-normal                                                    |
| Model 8               | Gumbel                     | Gamma                                                       | Gamma                                                         |
| Model 9               | Gumbel                     | Weibull                                                     | Weibull                                                       |
| Model 10              | Survival Clayton           | Log-normal                                                  | Log-normal                                                    |
| Model 11              | Survival Clayton           | Gamma                                                       | Gamma                                                         |
| Model 12              | Survival Clayton           | Weibull                                                     | Weibull                                                       |

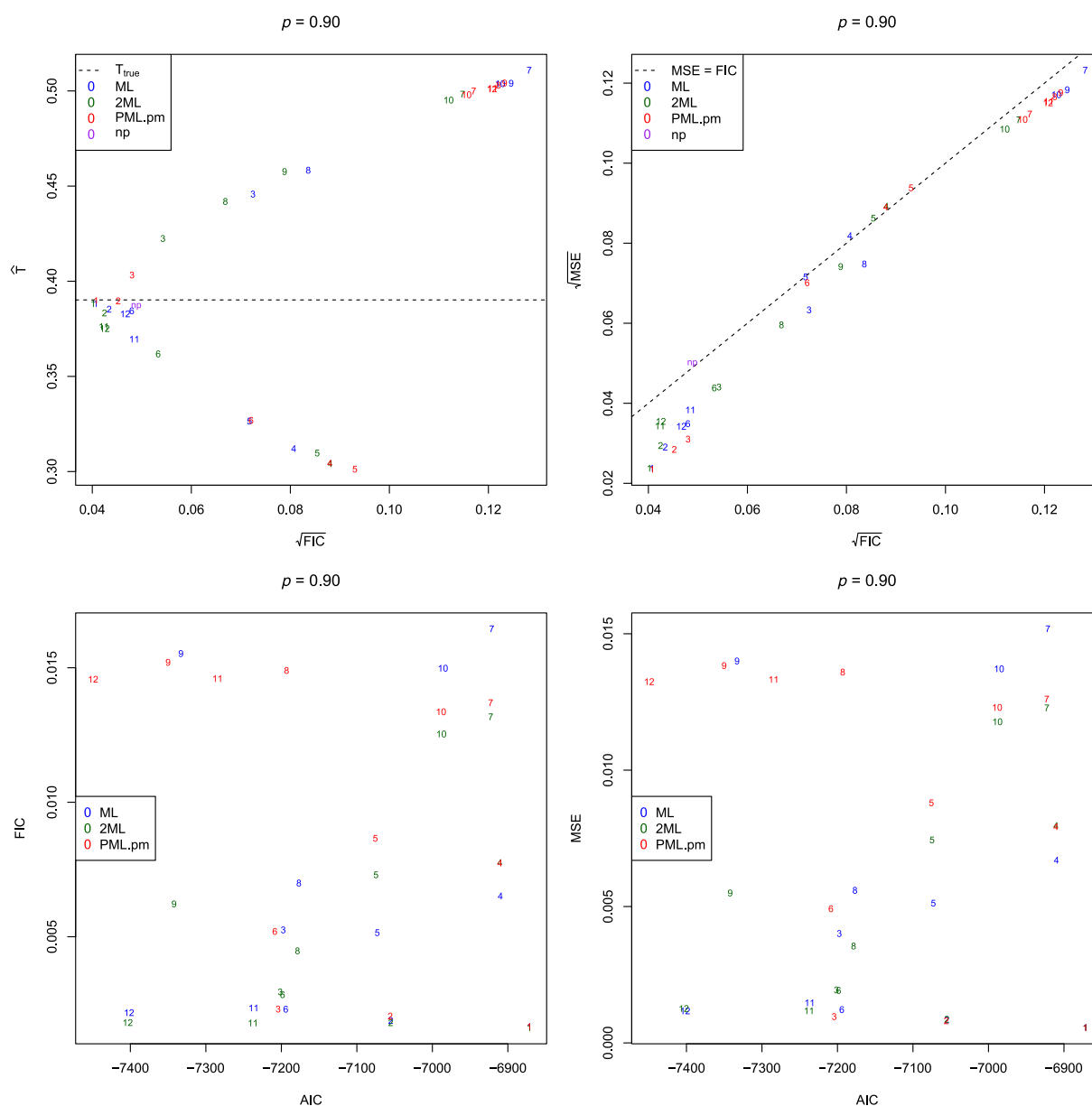


**FIGURE 3** Results from Part 2 of the simulations for the focused information criterion (FIC) with  $p = 0.90$ . The colored numbers refer to the model numbers in Table 2. ML = maximum likelihood; PML = pseudo maximum likelihood; MSE = mean square error; AIC = Akaike information criterion [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

When we consider a single focus parameter with  $p = 0.90$ , the models with smaller  $\sqrt{\text{FIC}}$  score tend to have  $\hat{T}$  values closer to the true focus parameter value, as anticipated. Moreover, the FIC is overall a good estimate of the MSE. When  $p = 0.95$ , the patterns are the same. However, the incorrectly specified models now have larger FIC and MSE values. This is logical because the upper conditional probability, which becomes upper-tail dependence when  $p \rightarrow 1$ , differs highly from model to model as  $p$  increases. Furthermore, there is little correspondence between the models preferred by the AIC and the FIC, except that they both pick the true model as the best model. Again, the models that are favored by the AIC are not necessarily good at estimating the upper-tail conditional probability, though they may be overall closer to the true model. This confirms that there is a clear gain in using the FIC when the model is to be used for estimating a specific quantity. The results for the AFIC are in line with the previous results with a single focus parameter.

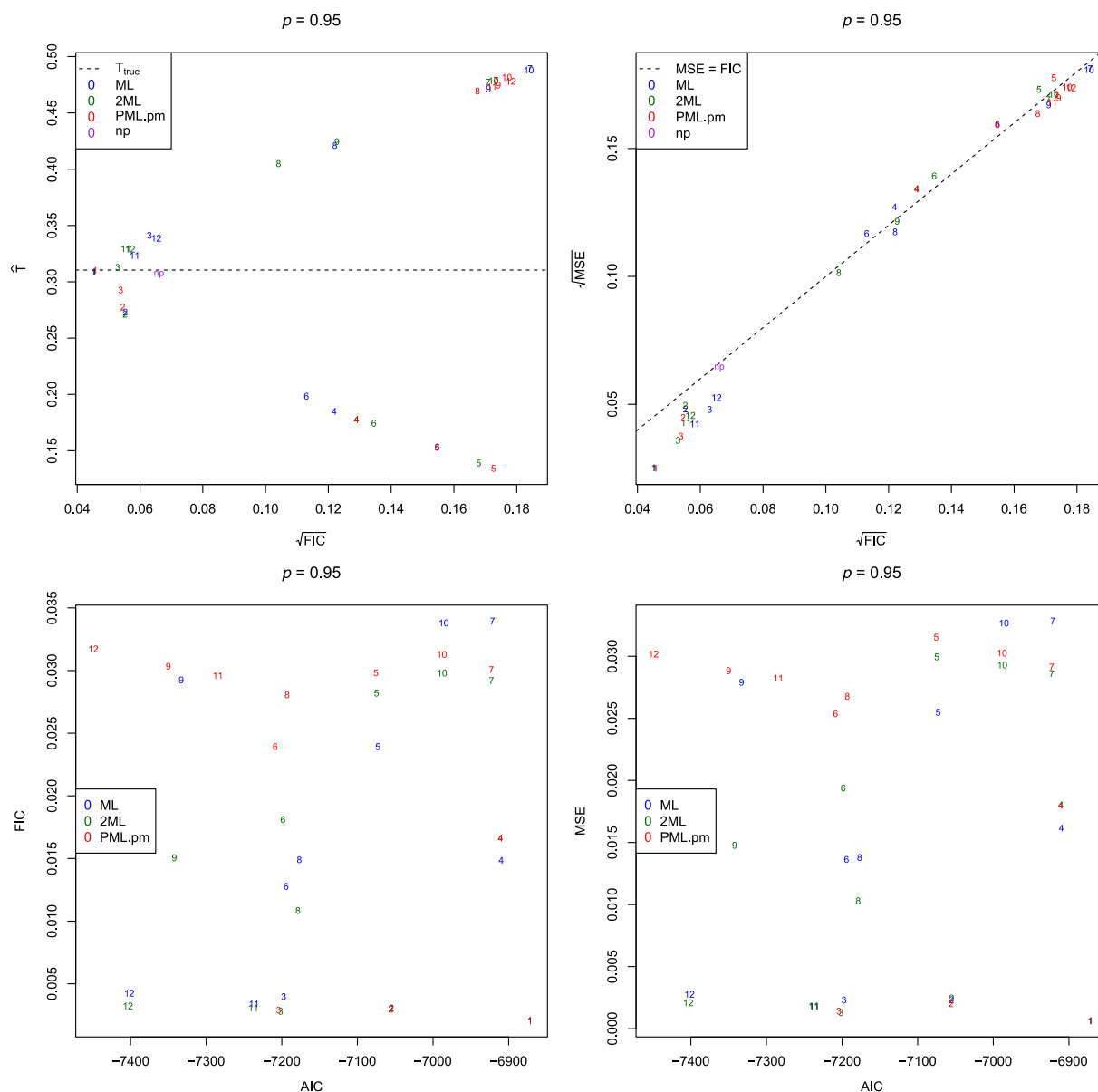


**FIGURE 4** Results from Part 2 of the simulations for the focused information criterion (FIC) with  $p = 0.95$ . The colored numbers refer to the model numbers in Table 2. ML = maximum likelihood; PML = pseudo maximum likelihood; MSE = mean square error; AIC = Akaike information criterion [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

## 6 | EXAMPLE: THE ABALONE DATA

As a real-life data example, we applied our FIC model selection method to the abalone data set (Asuncion & Newman, 2011), which has previously been used to illustrate different dependence modeling methods (e.g., Hobæk Haff, Aas, Frigessi, & Lacal, 2016; Ma, Sun, Chen, & Liu, 2012). Abalone is an edible sea snail, the harvest of which is subject to quotas. The quotas should reflect the age distribution of abalones and is partly based on information about their sizes. As the age of an abalone is cumbersome to determine, one would like to estimate the age based on some physical measurements such as weight and height. The data set consists of 4177 samples of nine variables. We focus on the dependence relations among the four variables “diameter,” “height,” “shell weight,” and “rings.” According to the description of the data set, the age in years is given

**TABLE 3** Description of the models used to fit the abalone data set

|         | Copula   | Margin 1 | Margin 2 | Margin 3 | Margin 4   |
|---------|----------|----------|----------|----------|------------|
| Model 1 | t        | Weibull  | Weibull  | Weibull  | NIG        |
| Model 2 | t        | Weibull  | Weibull  | Weibull  | Log-normal |
| Model 3 | t        | Normal   | Normal   | Normal   | Log-normal |
| Model 4 | Gaussian | Weibull  | Weibull  | Weibull  | NIG        |
| Model 5 | Gaussian | Weibull  | Weibull  | Weibull  | Log-normal |
| Model 6 | Gaussian | Normal   | Normal   | Normal   | Log-normal |
| Model 7 | Clayton  | Weibull  | Weibull  | Weibull  | NIG        |
| Model 8 | Clayton  | Weibull  | Weibull  | Weibull  | Log-normal |
| Model 9 | Clayton  | Normal   | Normal   | Normal   | Log-normal |

Note. NIG = normal-inverse Gaussian distribution.

by the number of rings + 1.5. Therefore, we convert the variable “ring” into “age” by adding this constant. Furthermore, we make the “age” variable continuous by adding Gaussian noise  $N(0, 0.01^2)$ .

Because the harvest quotas are related to the age of abalones as well as their size, an interesting focus parameter to consider is the probability that an abalone is under a certain age given that physical measurements are smaller than certain bounds. More specifically, we used the focus parameter

$$T(F) = P(\text{age} < 8 \mid \text{diameter} < 0.325, \text{height} < 0.105, \text{shellweight} < 0.109).$$

Here, 0.325, 0.105, and 0.109 were sample 20%-quantile values of “diameter,” “height,” and “shell weight,” respectively, whereas 8 is the 27%-quantile of “age.”

We have tried the nine candidate models described in Table 3. To choose adequate candidate margins, we fitted a set of well-known univariate distributions with the ML estimator. We then evaluated them by AIC. For copula, we fitted different copula models with the PML estimator and looked at maximized likelihood values. We composed Table 3 by choosing certain combinations of copula and margins. Each model in Table 3 is estimated with the two-stage ML and the PMLpm estimators. The ML estimator encountered numeric problems for some models.

The results are shown in Table 6 from the supplementary material (Ko et al., 2019) and visualized in Figure 5. We see a similar pattern to the ones from the simulation study. One of the notable differences is that the nonparametric model is now the winning model. This seems reasonable, considering that the nonparametric model becomes quite precise when the sample size is large enough. This is also in line with results from Part 1 of the simulations, where the FIC rank of the nonparametric model decreased from 13 to 5 when the sample size increased from 100 to 1000. Furthermore, scatter plots of the pseudo-observations (not shown here) indicate that the copula is rather different from well-known four-dimensional parametric copulas, with both asymmetric dependence and large differences between pairs. This also explains the preference for the nonparametric model. The winning models among the parametric ones, according to the FIC, are the ones based on the Gaussian or the t copula, which, as opposed to the Clayton, allow for different dependence between different pairs. Finally, the right panel shows a moderate agreement between the AIC and the FIC.

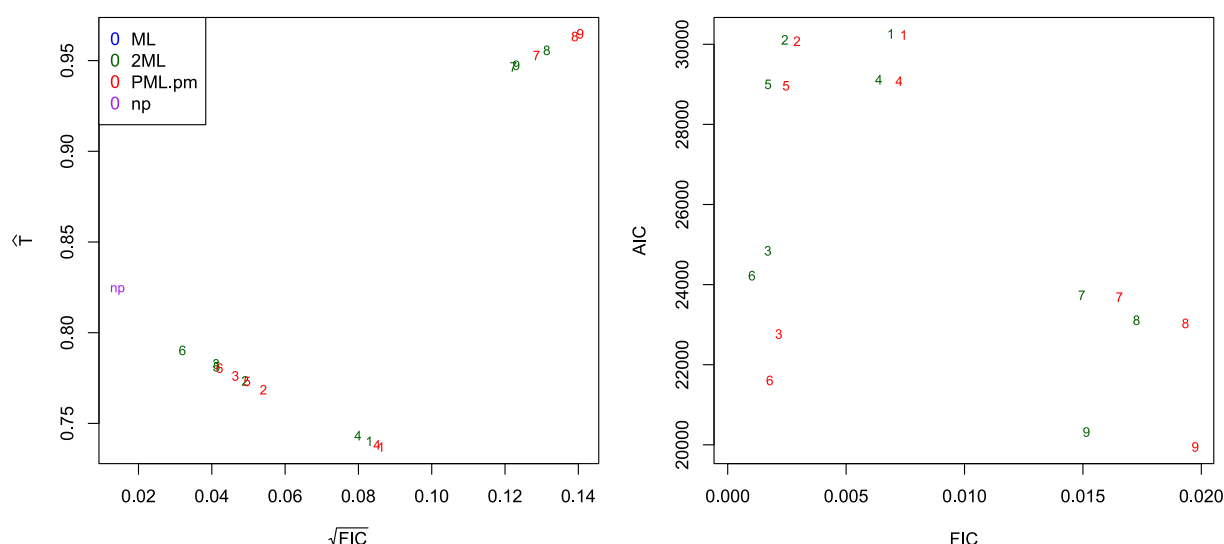


**FIGURE 5** Results from the models fitted to the abalone data. The colored numbers refer to the model numbers in Table 3. ML = maximum likelihood; PML = pseudo maximum likelihood; FIC = focused information criterion; AIC = Akaike information criterion [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

## 7 | CONCLUSIONS AND FURTHER RESEARCH

In this paper, we have developed the FIC for copula models, partly based on the general methods of Jullum and Hjort (2017), for three different estimators, namely, the ML, two-stage ML, and PMLpm. This is a model selection criterion that aims for the model that minimizes the estimated MSE of a chosen focus parameter and has the advantage that it can be used to compare models that are estimated under different estimation schemes, including the nonparametric one. It can also easily be extended to other estimation schemes as long as the joint asymptotic distribution with the nonparametric focus parameter estimate can be derived. In addition, it is a model-robust model selection criterion because it does not assume that any of the candidate model captures the data-generating model.

We performed a series of simulations to study the behavior of the FIC. We also applied our method to the abalone data set. The results show that models with lower FIC values give focus parameter estimates that are closer to the true parameter value, which is the aim of the method. Furthermore, it turns out that FIC also fulfills its role as an estimator of the MSE. Moreover, when the focus parameter is heavily based on a specific part of the distribution, particularly the tails, the models favored by the AIC do not necessarily give good estimates of the focus parameter. This seems natural because AIC aims for the model that minimizes overall closeness to the true model in terms of KL divergence and demonstrates that there is a clear advantage in using the FIC instead of the AIC in such situations.

Even though the FIC machinery can be applied to many different kinds of focus parameters, a limitation of the method is that it needs the focus parameter to be a smooth functional of the full distribution  $G$  so that an influence function can be derived. For some functionals, such as the density itself, or the copula function  $c_0$  in a given location, there is no influence function, and more elaborate estimators are called for, involving smoothing parameters. Versions of FIC may still be put up but requires further elaboration; see Jullum and Hjort (2017, Section 7).

As we see from the FIC plots (Figures 1, 2, 3, and 4), often the estimates from the best models are in reasonable agreement and a final estimate can be based on a suitable average over

these. Claeskens and Hjort (2008, Chapter 7) propose and develop machinery for such model average estimators and suggest using weights proportional to  $\exp(-\lambda \cdot \text{FIC})$ , with  $\lambda$  being a tuning parameter. We do not pursue that theme in this paper, however.

In this paper, we have only considered parametric copula models with parametric margins and i.i.d. variables. One can for example extend the FIC framework to the PML estimator where margins are nonparametric, as well as to time series and vine models. Furthermore, it is possible to derive another variant of FIC by using another loss function than the quadratic loss function (e.g., absolute loss).

## ACKNOWLEDGEMENTS

The authors would like to thank Steffen Grønneberg for his valuable comments and fruitful discussions. The authors also acknowledge partial funding from the Norwegian Research Council-supported research group FocuStat (Focus Driven Statistical Inference With Complex Data) and from the Department of Mathematics at the University of Oslo.

## ORCID

Vinnie Ko  <https://orcid.org/0000-0002-0680-5518>

## REFERENCES

- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19, 716–723.
- Asuncion, A., & Newman, D. (2011). UCI machine learning repository. Irvine, CA: School of Information and Computer Sciences, University of California. Retrieved from <https://archive.ics.uci.edu/ml/datasets.html>
- Bevacqua, E., Maraun, D., Hobæk Haff, I., Widmann, M., & Vrac, M. (2017). Multivariate statistical modelling of compound events via pair-copula constructions: Analysis of floods in Ravenna (Italy). *Hydrology and Earth System Sciences*, 21, 2701–2723.
- Chen, X., & Fan, Y. (2005). Pseudo-likelihood ratio tests for semiparametric multivariate copula model selection. *The Canadian Journal of Statistics*, 33, 389–414.
- Claeskens, G., & Hjort, N. L. (2003). The focused information criterion [with discussion contributions and a rejoinder]. *Journal of the American Statistical Association*, 98, 900–916.
- Claeskens, G., & Hjort, N. L. (2008). *Model selection and model averaging*. Cambridge, UK: Cambridge University Press.
- Efron, B. (1982). *The jackknife, the bootstrap, and other resampling plans* (Vol. 38). Philadelphia, PA: SIAM.
- Genest, C., Ghoudi, K., & Rivest, L.-P. (1995). A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika*, 82, 543–552.
- Grønneberg, S., & Hjort, N. L. (2014). The copula information criteria. *Scandinavian Journal of Statistics*, 41, 436–459.
- Hardin, J. W. (2003). The sandwich estimate of variance. In T. B. Fomby & R. Carter Hill (Eds.), *Maximum likelihood estimation of misspecified models: Twenty years later* (pp. 45–73). Bingley, UK: Emerald Group Publishing.
- Hobæk Haff, I., Aas, K., Frigessi, A., & Lacal, V. (2016). Structure learning in Bayesian networks using regular vines. *Computational Statistics & Data Analysis*, 101, 186–208.
- Hobæk Haff, I., Frigessi, A., & Maraun, D. (2015). How well do regional climate models simulate the spatial dependence of precipitation? An application of pair-copula constructions. *Journal of Geophysical Research: Atmospheres*, 120, 2624–2646.
- Joe, H. (1997). *Multivariate models and multivariate dependence concepts*. Boca Raton, FL: CRC Press.
- Joe, H. (2005). Asymptotic efficiency of the two-stage estimation method for copula-based models. *Journal of Multivariate Analysis*, 94, 401–419.
- Joe, H. (2014). *Dependence modeling with copulas*. New York, NY: CRC Press.

- Jullum, M., & Hjort, N. L. (2017). Parametric or nonparametric: The FIC approach. *Statistica Sinica*, 27, 951–981.
- Jullum, M., & Hjort, N. L. (2018). What price semiparametric Cox regression?. *Lifetime Data Analysis*, 1, 1–33. <https://doi.org/10.1007/s10985-018-9450-7>
- Kim, G., Silvapulle, M. J., & Silvapulle, P. (2007). Comparison of semiparametric and parametric methods for estimating copulas. *Computational Statistics & Data Analysis*, 51, 2836–2850.
- Ko, V., & Hjort, N. L. (2019a). Copula information criterion for model selection with two-stage maximum likelihood estimation. *Econometrics and Statistics*, (in press).
- Ko, V., & Hjort, N. L. (2019b). Model robust inference with two-stage maximum likelihood estimation for copulas. *Journal of Multivariate Analysis*, 171, 362–381.
- Ko, V., Hjort, N. L., & Hobæk Haff, I. (2019). Supplement to “Focused information criteria for copulas”. *Scandinavian Journal of Statistics*.
- Le Cam, L. (1990). Maximum likelihood: An introduction. *International Statistical Review*, 58, 153–171.
- Ma, J., Sun, Z.-Q., Chen, S., & Liu, H.-H. (2012). Dependence tree structure estimation via copula. *International Journal of Automation and Computing*, 9, 113–121.
- Nelsen, R. B. (2006). *An introduction to copulas*. New York, NY: Springer Science & Business Media.
- Shao, J. (2003). *Mathematical statistics*. New York, NY: Springer Science & Business Media.
- Shih, J. H., & Louis, T. A. (1995). Inferences on the association parameter in copula models for bivariate survival data. *Biometrics*, 51, 1384–1399.
- Sklar, A. (1959). Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut Statistique de l'Université de Paris*, 8, 229–231.
- Tsukahara, H. (2005). Semiparametric estimation in copula models. *The Canadian Journal of Statistics*, 33, 357–375.
- White, H. (1982). Maximum likelihood estimation of misspecified models. *Econometrica: Journal of the Econometric Society*, 1–25.

## SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of the article.

**How to cite this article:** Ko V, Hjort NL, Hobæk Haff I. Focused information criteria for copulas. *Scand J Statist.* 2019;1–24. <https://doi.org/10.1111/sjos.12387>

## APPENDIX

### PROOFS

#### Proof of Proposition 1

*Proof.* Under the assumed regularity conditions, we have from Theorem 5.5 in Shao (2003) and Proposition 1 in Ko and Hjort (2019b) that

$$\hat{T}_{\text{np}} - T_{\text{true}} = \frac{1}{n} \sum_{i=1}^n \text{IF}_T(y_i, G) + o_p(n^{-1/2}),$$

$$\hat{\eta}^{2\text{ML}} - \eta_0^{2\text{ML}} = (\mathcal{I}_{\eta}^{2\text{ML}})^{-1} \phi_{n,\eta}^{2\text{ML}}(\eta_0^{2\text{ML}}) + o_p(n^{-1/2}),$$

where

$$\phi_{n,\eta}^{2\text{ML}}(\eta) = \frac{1}{n} \sum_{i=1}^n \phi_{\eta}^{2\text{ML}}(y_i, \eta),$$

$$\phi_{n,\alpha}^{2\text{ML}}(\alpha) = \frac{1}{n} \sum_{i=1}^n \phi_{\alpha}^{2\text{ML}}(y_i, \alpha),$$

$$\phi_{n,\theta}^{2\text{ML}}(\theta) = \frac{1}{n} \sum_{i=1}^n \phi_{\theta}^{2\text{ML}}(y_i, \theta),$$

$$\phi_{\eta}^{2\text{ML}}(\eta) = \begin{pmatrix} \phi_{\alpha}^{2\text{ML}}(\alpha) \\ \phi_{\theta}^{2\text{ML}}(\theta) \end{pmatrix},$$

$$\phi_{\alpha}^{2\text{ML}}(y, \alpha) = \begin{pmatrix} \phi_{\alpha_1}^{2\text{ML}}(y_1, \alpha_1) \\ \vdots \\ \phi_{\alpha_d}^{2\text{ML}}(y_d, \alpha_d) \end{pmatrix} = \begin{pmatrix} \partial \log f_1(y_1, \alpha_1) / \partial \alpha_1 \\ \vdots \\ \partial \log f_d(y_d, \alpha_d) / \partial \alpha_d \end{pmatrix},$$

$$\phi_{\theta}^{2\text{ML}}(y, \eta) = \frac{\partial \log c(F_1(y_1, \alpha_1), \dots, F_d(y_d, \alpha_d), \theta)}{\partial \theta},$$

$$\mathcal{I}_{\eta}^{2\text{ML}} = \begin{pmatrix} \mathcal{I}_{\alpha}^{2\text{ML}} & 0 \\ \left(\mathcal{I}_{\alpha,\theta}^{2\text{ML}}\right)^{\text{T}} & \mathcal{I}_{\theta}^{2\text{ML}} \end{pmatrix},$$

$$\mathcal{I}_{\alpha}^{2\text{ML}} = \begin{pmatrix} \mathcal{I}_{\alpha_1}^{2\text{ML}} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathcal{I}_{\alpha_d}^{2\text{ML}} \end{pmatrix},$$

$$\mathcal{I}_{\alpha_j}^{2\text{ML}} = -\text{E}_{G_j} \left[ H_{\alpha_j}^{2\text{ML}}(y_j, \alpha_{0,j}^{2\text{ML}}) \right] = - \int g_j \frac{\partial^2 \log f_j(y_j, \alpha_{0,j}^{2\text{ML}})}{\partial \alpha_j \partial \alpha_j^{\text{T}}} dy_j,$$

$$\mathcal{I}_{\theta}^{2\text{ML}} = -\text{E}_G \left[ H_{\theta}^{2\text{ML}}(y, \eta_0^{2\text{ML}}) \right] = - \int g \frac{\partial^2 \log c(F_1(y_1, \alpha_{0,1}^{2\text{ML}}), \dots, F_d(y_d, \alpha_{0,d}^{2\text{ML}}), \theta_0^{2\text{ML}})}{\partial \theta \partial \theta^{\text{T}}} dy,$$

$$\mathcal{I}_{\alpha,\theta}^{2\text{ML}} = -\text{E}_G \left[ H_{\alpha,\theta}^{2\text{ML}}(y, \eta_0^{2\text{ML}}) \right] = - \int g \frac{\partial^2 \log c(F_1(y_1, \alpha_{0,1}^{2\text{ML}}), \dots, F_d(y_d, \alpha_{0,d}^{2\text{ML}}), \theta_0^{2\text{ML}})}{\partial \alpha \partial \theta^{\text{T}}} dy.$$

Applying the multivariate central limit theorem to the summand yields the joint distribution

$$\sqrt{n} \begin{pmatrix} \hat{T}_{\text{np}} - T_{\text{true}} \\ \hat{\eta}^{2\text{ML}} - \eta_0^{2\text{ML}} \end{pmatrix} \xrightarrow{d} \begin{pmatrix} \Lambda_{\text{np}} \\ (\mathcal{I}_{\eta}^{2\text{ML}})^{-1} \Lambda_{\eta^{2\text{ML}}} \end{pmatrix} \sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} V_{\text{np}} & V_{\text{np},\eta^{2\text{ML}}} \\ (V_{\text{np},\eta^{2\text{ML}}})^{\text{T}} & V_{\eta^{2\text{ML}}} \end{pmatrix} \right),$$

where

$$V_{\text{np},\eta^{2\text{ML}}} = Q_{\eta}^{2\text{ML}} \left( (\mathcal{I}_{\eta}^{2\text{ML}})^{-1} \right)^{\text{T}},$$

$$V_{\eta^{2\text{ML}}} = (\mathcal{I}_{\eta}^{2\text{ML}})^{-1} K_{\eta}^{2\text{ML}} \left( (\mathcal{I}_{\eta}^{2\text{ML}})^{-1} \right)^{\text{T}},$$

$$Q_{\eta}^{2\text{ML}} = \text{Cov}_G \left( \text{IF}_T(y, G), \phi_{\eta}^{2\text{ML}}(y, \eta_0^{2\text{ML}}) \right) = \text{E}_G \left[ \text{IF}_T(y, G) \cdot \phi_{\eta}^{2\text{ML}}(y, \eta_0^{2\text{ML}})^{\text{T}} \right],$$

$$K_{\eta}^{2\text{ML}} = \text{Var}_G \left( \phi_{\eta}^{2\text{ML}}(y, \eta_0^{2\text{ML}}) \right) = \text{E}_G \left[ \phi_{\eta}^{2\text{ML}}(y, \eta_0^{2\text{ML}}) \phi_{\eta}^{2\text{ML}}(y, \eta_0^{2\text{ML}})^{\text{T}} \right].$$

Now, by applying the delta method with transformation function  $s_{x,\eta}(x, \eta) = (x, s_\eta)^T = (x, T(F(\eta)))^T$ , we obtain

$$\sqrt{n} \begin{pmatrix} \hat{T}_{np} - T_{\text{true}} \\ \hat{T}_{2ML} - T_{0,2ML} \end{pmatrix} \xrightarrow{d} \begin{pmatrix} \Lambda_{np} \\ \dot{s}_\eta^T (\mathcal{I}_\eta^{2ML})^{-1} \Lambda_{\eta^{2ML}} \end{pmatrix} \sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} V_{np} & V_{np,2ML} \\ V_{np,2ML}^T & V_{2ML} \end{pmatrix} \right).$$

□

## Proof of Proposition 2

*Proof.* Under the assumed regularity conditions, we have from Theorem 5.5 in Shao (2003), Lemma 1 in Ko and Hjort (2019b), and Proposition 2 from Chen and Fan (2005) that

$$\begin{aligned} \hat{T}_{np} - T_{\text{true}} &= \frac{1}{n} \sum_{i=1}^n \text{IF}_T(y_i, G) + o_p(n^{-1/2}), \\ \hat{\alpha}^{2ML} - \alpha_0^{2ML} &= (\mathcal{I}_\alpha^{2ML})^{-1} \phi_{n,\alpha}^{2ML}(\alpha_0^{2ML}) + o_p(n^{-1/2}), \\ \hat{\theta}^{PML} - \theta_0^{PML} &= (\mathcal{I}_\theta^{PML})^{-1} \phi_{n,\theta}^{PML}(\theta_0^{PML}) + o_p(n^{-1/2}), \end{aligned}$$

where

$$\begin{aligned} \phi_{n,\theta}^{PML}(\theta) &= \frac{1}{n} \sum_{i=1}^n \phi_\theta^{PML}(G_{n,1}(y_{i,1}), \dots, G_{n,d}(y_{i,d}), \theta), \\ G_{n,j}(y_j) &= \frac{1}{n+1} \sum_{k=1}^n I\{y_{k,j} \leq y_j\}, \\ \phi_\theta^{PML}(u, \theta) &= \frac{\partial \log c(u_1, \dots, u_d), \theta}{\partial \theta}, \\ \mathcal{I}_\theta^{PML} &= -E_{C_0} [H_\theta^{PML}(u, \theta_0^{PML})] = - \int_{[0,1]^d} c_0(u) \frac{\partial^2 \log c(u, \theta_0^{PML})}{\partial \theta \partial \theta^T} du, \\ u &= (u_1, \dots, u_d) = (G_1(y_1), \dots, G_d(y_d)). \end{aligned}$$

Applying Lemma 2 from Chen and Fan (2005), which becomes the central limit theorem when one holds the empirical process in the marginal parts constant, yields the joint distribution

$$\begin{aligned} \sqrt{n} \begin{pmatrix} \hat{T}_{np} - T_{\text{true}} \\ \hat{\eta}^{PMLpm} - \eta_0^{PMLpm} \end{pmatrix} &\xrightarrow{d} \begin{pmatrix} \Lambda_{np} \\ (\mathcal{I}_\eta^{PMLpm})^{-1} \Lambda_{\eta^{PMLpm}} \end{pmatrix} \\ &\sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} V_{np} & V_{np,\eta^{PMLpm}} \\ (V_{np,\eta^{PMLpm}})^T & V_{\eta^{PMLpm}} \end{pmatrix} \right), \end{aligned}$$

where

$$\begin{aligned} \hat{\eta}^{PMLpm} &= ((\hat{\alpha}^{2ML})^T, (\hat{\theta}^{PML})^T)^T, \\ \eta_0^{PMLpm} &= ((\alpha_0^{2ML})^T, (\theta_0^{PML})^T)^T, \\ \mathcal{I}_\eta^{PMLpm} &= \begin{pmatrix} \mathcal{I}_\alpha^{2ML} & 0 \\ 0 & \mathcal{I}_\theta^{PML} \end{pmatrix}, \\ V_{np,\eta^{PMLpm}} &= Q_\eta^{PMLpm} (\mathcal{I}_\eta^{PMLpm})^{-1}, \end{aligned}$$

$$\begin{aligned}
V_{\eta^{\text{PMLpm}}} &= \left( \mathcal{I}_{\eta}^{\text{PMLpm}} \right)^{-1} K_{\eta}^{\text{PMLpm}} \left( \mathcal{I}_{\eta}^{\text{PMLpm}} \right)^{-1}, \\
Q_{\eta}^{\text{PMLpm}} &= \text{COV}_G \left( \text{IF}_T(y, G), \phi_{\eta}^{\text{PMLpm}} \left( y, \eta_0^{\text{PMLpm}} \right) + \mathcal{W}^* \left( y, \theta_0^{\text{PML}} \right) \right) \\
&= E_G \left[ \text{IF}_T(y, G) \cdot \left( \phi_{\eta}^{\text{PMLpm}} \left( y, \eta_0^{\text{PMLpm}} \right) + \mathcal{W}^* \left( y, \theta_0^{\text{PML}} \right) \right)^T \right], \\
K_{\eta}^{\text{PMLpm}} &= \text{Var}_G \left( \phi_{\eta}^{\text{PMLpm}} \left( y, \eta_0^{\text{PMLpm}} + \mathcal{W}^* \left( y, \theta_0^{\text{PML}} \right) \right) \right) \\
&= E_G \left[ \left( \phi_{\eta}^{\text{PMLpm}} \left( y, \eta_0^{\text{PMLpm}} \right) + \mathcal{W}^* \left( y, \theta_0^{\text{PML}} \right) \right) \right. \\
&\quad \left. \cdot \left( \phi_{\eta}^{\text{PMLpm}} \left( y, \eta_0^{\text{PMLpm}} \right) + \mathcal{W}^* \left( y, \theta_0^{\text{PML}} \right) \right)^T \right], \\
\phi_{\eta}^{\text{PMLpm}}(y, \eta) &= \left( \begin{array}{c} \phi_{\alpha}^{2\text{ML}}(y, \alpha) \\ \phi_{\theta}^{\text{PML}}(\mathcal{W}^*((G_1(y_1), \dots, G_d(y_d)), \theta), \theta) \end{array} \right), \\
\mathcal{W}^*(y, \theta) &= \left( \begin{array}{c} 0_{\dim(\alpha)} \\ \mathcal{W}((G_1(y_1), \dots, G_d(y_d)), \theta) \end{array} \right), \\
\mathcal{W}(u, \theta) &= \sum_{j=1}^d \mathcal{W}_j(u_j, \theta) = \sum_{j=1}^d \int_{[0,1]^d} c_0(v) \frac{\partial^2 \log c(v, \theta)}{\partial \theta \partial v_j^T} (I(u_j \leq v_j) - v_j) \, dv.
\end{aligned}$$

Now, by applying the delta method with transformation function  $s_{x,\eta}(x, \eta)$ , we obtain

$$\begin{aligned}
\sqrt{n} \left( \begin{array}{c} \hat{T}_{\text{np}} - T_{\text{true}} \\ \hat{T}_{\text{PMLpm}} - T_{0,\text{PMLpm}} \end{array} \right) &\xrightarrow{d} \left( \begin{array}{c} \Lambda_{\text{np}} \\ \dot{s}_{\eta}^T \left( \mathcal{I}_{\eta}^{\text{PMLpm}} \right)^{-1} \Lambda_{\eta^{\text{PMLpm}}} \end{array} \right) \\
&\sim N \left( \left( \begin{array}{c} 0 \\ 0 \end{array} \right), \left( \begin{array}{cc} V_{\text{np}} & V_{\text{np,PMLpm}} \\ V_{\text{np,PMLpm}}^T & V_{\text{PMLpm}} \end{array} \right) \right).
\end{aligned}$$

□

### Proof of Lemma 3

*Proof.* When the copula model is estimated with the pseudo ML with parametric margins, the nonconstant part of the KL divergence is

$$\begin{aligned}
Q(\hat{\eta}^{\text{PMLpm}}) &= \int g(y) \left\{ \log f_1(y_1, \hat{\alpha}_1^{2\text{ML}}) + \dots + \log f_d(y_d, \hat{\alpha}_d^{2\text{ML}}) \right. \\
&\quad \left. + \log c \left( F_1(y_1, \hat{\alpha}_1^{2\text{ML}}), \dots, F_d(y_d, \hat{\alpha}_d^{2\text{ML}}), \hat{\theta}^{\text{PML}} \right) \right\} dy.
\end{aligned}$$

The empirical equivalent is

$$\begin{aligned}
\hat{Q}(\hat{\eta}^{\text{PMLpm}}) &= \frac{1}{n} \sum_{i=1}^n \left[ \log f_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}) + \dots + \log f_d(y_{i,d}, \hat{\alpha}_d^{2\text{ML}}) \right. \\
&\quad \left. + \log c \left( F_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}), \dots, F_d(y_{i,d}, \hat{\alpha}_d^{2\text{ML}}), \hat{\theta}^{\text{PML}} \right) \right].
\end{aligned}$$

Now, we check the bias of  $\hat{Q}(\hat{\eta}^{\text{PMLpm}})$ :

$$\begin{aligned} E_G \left[ \hat{Q}(\hat{\eta}^{\text{PMLpm}}) \right] - Q(\hat{\eta}^{\text{PMLpm}}) &= E_{G_1} \left[ \frac{1}{n} \sum_{i=1}^n \log f_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}) \right] \\ &\quad - \int g(y_1) \log f_1(y_1, \hat{\alpha}_1^{2\text{ML}}) dy_1 + \dots \\ &\quad + E_{G_d} \left[ \frac{1}{n} \sum_{i=1}^n \log f_d(y_{i,d}, \hat{\alpha}_d^{2\text{ML}}) \right] \\ &\quad - \int g(y_d) \log f_d(y_d, \hat{\alpha}_d^{2\text{ML}}) dy_d \\ &\quad + E_G \left[ \frac{1}{n} \sum_{i=1}^n \log c \left( F_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}), \dots, F_d(y_{i,d}, \hat{\alpha}_d^{2\text{ML}}), \hat{\theta}^{\text{PML}} \right) \right] \\ &\quad - \int g(y) \log c \left( F_1(y_1, \hat{\alpha}_1^{2\text{ML}}), \dots, F_d(y_d, \hat{\alpha}_d^{2\text{ML}}), \hat{\theta}^{\text{PML}} \right) dy. \end{aligned}$$

Because the parameter estimates  $\hat{\alpha}_j^{2\text{ML}}$  for the margins are obtained with ML estimation, we can use the results from the derivation of the TIC (Claeskens & Hjort, 2008, Chapter 2) directly and obtain

$$E_{G_j} \left[ \frac{1}{n} \sum_{i=1}^n \log f_j(y_{i,j}, \hat{\alpha}_j^{2\text{ML}}) \right] - \int g(y_j) \log f_j(y_j, \hat{\alpha}_j^{2\text{ML}}) dy_j = \frac{1}{n} \text{tr} \left( I_{\alpha_j}^{-1} K_{\alpha_j} \right) + o_p(n^{-1}).$$

In a nutshell,  $I_{\alpha_j}$  is the Fisher information of the  $j$ th margin and  $K_{\alpha_j}$  is the covariance matrix of the score vector that belongs to the  $j$ th margin. If one assumes that the model is correctly specified (i.e.,  $f = g$ ), we have  $I_{\alpha_j} = K_{\alpha_j}$  (Hardin, 2003; Le Cam, 1990; White, 1982). Therefore,  $\text{tr}(I_{\alpha_j}^{-1} K_{\alpha_j}) = \dim(\alpha_j)$ .

Furthermore, let

$$Q_c(\hat{\eta}^{\text{PMLpm}}) = \int g(y) \log c \left( F_1(y_1, \hat{\alpha}_1^{2\text{ML}}), \dots, F_d(y_d, \hat{\alpha}_d^{2\text{ML}}), \hat{\theta}^{\text{PML}} \right) dy$$

and

$$\hat{Q}_c(\hat{\eta}^{\text{PMLpm}}) = E \left[ \frac{1}{n} \sum_{i=1}^n \log c \left( F_1(y_{i,1}, \hat{\alpha}_1^{2\text{ML}}), \dots, F_d(y_{i,d}, \hat{\alpha}_d^{2\text{ML}}), \hat{\theta}^{\text{PML}} \right) \right].$$

Therefore, we can write

$$\begin{aligned} E_G \left[ \hat{Q}(\hat{\eta}^{\text{PMLpm}}) \right] - Q(\hat{\eta}^{\text{PMLpm}}) &= \frac{1}{n} \sum_{j=1}^d \text{tr} \left( I_{\alpha_j}^{-1} K_{\alpha_j} \right) \\ &\quad + E_G \left[ \hat{Q}_c(\hat{\eta}^{\text{PMLpm}}) \right] - Q_c(\hat{\eta}^{\text{PMLpm}}) + o(n^{-1}). \end{aligned}$$

Now,  $E_G[\hat{Q}_c(\hat{\eta}^{\text{PMLpm}})] - Q_c(\hat{\eta}^{\text{PMLpm}})$  is the only element that has to be evaluated. Let

$$\begin{aligned} Q_c(\eta_0^{\text{PMLpm}}) &= \int g(y) \log c(F_1(y_1, \alpha_{0,1}^{2\text{ML}}, \dots, F_d(y_d, \alpha_{0,d}^{2\text{ML}}, \theta_0^{\text{PML}}) dy, \\ Z_i &= \log c(F_1(y_{i,1}, \alpha_{0,1}^{2\text{ML}}, \dots, F_d(y_{i,d}, \alpha_{0,d}^{2\text{ML}}, \theta_0^{\text{PML}}) - Q_c(\eta_0^{\text{PMLpm}}), \\ A_\eta &= \sqrt{n}(\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}}) = \sqrt{n}(\mathcal{I}_\eta^{\text{PMLpm}})^{-1} \phi_\eta^{\text{PMLpm}}(y, \eta_0^{\text{PMLpm}}) + \begin{pmatrix} o_p(1) \\ o_p(1) \end{pmatrix}, \end{aligned}$$

which stems from the proof of Proposition 2, and furthermore,

$$\begin{aligned} \phi_{n,\eta}^*(\eta) &= \frac{1}{n} \sum_{i=1}^n \phi_\eta^*(y_i, \eta) = \frac{1}{n} \sum_{i=1}^n \frac{\partial \log c(F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta)}{\partial \eta}, \\ H_{n,\eta}^*(\eta) &= \frac{1}{n} \sum_{i=1}^n H_\eta^*(y_i, \eta) = \frac{1}{n} \sum_{i=1}^n \frac{\partial^2 \log c(F_1(y_{i,1}, \alpha_1), \dots, F_d(y_{i,d}, \alpha_d), \theta)}{\partial \eta \partial \eta^T}, \\ \mathcal{I}_\eta^* &= -E_G[H_\eta^*(y, \eta_0^{\text{PMLpm}})] = - \int g(y) H_\eta^*(y, \eta_0^{\text{PMLpm}}) dy. \end{aligned}$$

Then, we have  $E[\bar{Z}_n] = \frac{1}{n} \sum_{i=1}^n E[Z_i] = 0$ , along with

$$\begin{aligned} \hat{Q}_c(\hat{\eta}^{\text{PMLpm}}) &= \frac{1}{n} \sum_{i=1}^n \log c(y_i, \hat{\eta}^{\text{PMLpm}}) \\ &= \frac{1}{n} \sum_{i=1}^n \left[ \log c(y_i, \eta_0^{\text{PMLpm}}) - Q_c(\eta_0^{\text{PMLpm}}) + Q_c(\eta_0^{\text{PMLpm}}) \right. \\ &\quad \left. + (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}})^T \phi_\eta^*(y_i, \eta_0^{\text{PMLpm}}) \right. \\ &\quad \left. + \frac{1}{2} (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}})^T H_\eta^*(y_i, \eta_0^{\text{PMLpm}}) (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}}) \right] + o_p(n^{-1}) \\ &= \frac{1}{n} \sum_{i=1}^n [Z_i] + Q_c(\eta_0^{\text{PMLpm}}) + (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}})^T \phi_{n,\eta}^*(\eta_0^{\text{PMLpm}}) \\ &\quad + \frac{1}{2} (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}})^T H_{n,\eta}^*(\eta_0^{\text{PMLpm}}) (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}}) + o_p(n^{-1}) \\ &= Q_c(\eta_0^{\text{PMLpm}}) + \bar{Z}_n + \frac{1}{\sqrt{n}} A_\eta^T \phi_{n,\eta}^*(\eta_0^{\text{PMLpm}}) \\ &\quad + \frac{1}{2n} A_\eta^T H_{n,\eta}^*(\eta_0^{\text{PMLpm}}) A_\eta + o_p(n^{-1}), \end{aligned}$$

$$\begin{aligned} Q_c(\hat{\eta}^{\text{PMLpm}}) &= \int g(y) \log c(F_1(y_1, \hat{\alpha}_1^{2\text{ML}}, \dots, F_d(y_d, \hat{\alpha}_d^{2\text{ML}}, \hat{\theta}^{\text{PML}}) dy \\ &= \int g(y) \left[ \log c(y, \eta_0^{\text{PMLpm}}) + (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}})^T \phi_\eta^*(y, \eta_0^{\text{PMLpm}}) \right. \\ &\quad \left. + \frac{1}{2} (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}})^T H_\eta^*(y, \eta_0^{\text{PMLpm}}) (\hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}}) \right] dy + o_p(n^{-1}) \end{aligned}$$

$$\begin{aligned}
&= \int g(y) \log c(y, \eta_0^{\text{PMLpm}}) dy + \left( \hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}} \right)^T \int g(y) \phi_{\eta}^*(y, \eta_0^{\text{PMLpm}}) dy \\
&\quad + \frac{1}{2} \left( \hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}} \right)^T \int g(y) H_{\eta}^*(y, \eta_0^{\text{PMLpm}}) dy \\
&\quad \cdot \left( \hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}} \right) + o_p(n^{-1}), \\
&= Q_c(\eta_0^{\text{PMLpm}}) + \left( \hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}} \right)^T \cdot 0 \\
&\quad + \frac{1}{2n} \sqrt{n} \left( \hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}} \right)^T \int g(y) H_{\eta}^*(y, \eta_0^{\text{PMLpm}}) dy \\
&\quad \cdot \sqrt{n} \left( \hat{\eta}^{\text{PMLpm}} - \eta_0^{\text{PMLpm}} \right) + o_p(n^{-1}) \\
&= Q_c(\eta_0^{\text{PMLpm}}) - \frac{1}{2n} A_{\eta}^T \mathcal{I}_{\eta}^* A_{\eta} + o_p(n^{-1}),
\end{aligned}$$

and

$$\begin{aligned}
n \left\{ \hat{Q}_c(\hat{\eta}^{\text{PMLpm}}) - Q_c(\hat{\eta}^{\text{PMLpm}}) \right\} &= n\bar{Z}_n + \sqrt{n} A_{\eta}^T \phi_{n,\eta}^*(\eta_0^{\text{PMLpm}}) \\
&\quad + \frac{1}{2} A_{\eta}^T H_{n,\eta}^*(\eta_0^{\text{PMLpm}}) A_{\eta} + \frac{1}{2} A_{\eta}^T \mathcal{I}_{\eta}^* A_{\eta} + o_p(1).
\end{aligned}$$

We now assume that the margins are correctly specified, that is,  $F_j(y_j, \alpha_{0,j}^{2\text{ML}}) = G_j(y_j)$ . This implies that  $\theta_0^{\text{PML}} = \theta_0^{2\text{ML}}$  and, consequently,  $\eta_0^{\text{PMLpm}} = \eta_0^{2\text{ML}}$ . Then, the central limit theorem gives

$$\begin{aligned}
\sqrt{n} \phi_{n,\eta}^*(\eta_0^{\text{PMLpm}}) &= \sqrt{n} \phi_{n,\eta}^*(\eta_0^{2\text{ML}}) \\
&= \sqrt{n} \left\{ \phi_{n,\eta}^*(\eta_0^{2\text{ML}}) - E[\phi_{\eta}^*(Y, \eta_0^{2\text{ML}})] \right\} \xrightarrow{d} \Lambda_{\eta}^* \sim N(0, K_{\eta}^*),
\end{aligned}$$

where  $K_{\eta}^* = \text{Var}(\phi_{\eta}^*(y, \eta_0^{2\text{ML}})) = E[\phi_{\eta}(y, \eta_0^{2\text{ML}}) \phi_{\eta}(y, \eta_0^{2\text{ML}})^T]$ .

Now, we evaluate  $E_G[n\{\hat{Q}_c(\hat{\eta}^{\text{PMLpm}}) - Q_c(\hat{\eta}^{\text{PMLpm}})\}]$ :

$$\begin{aligned}
&E_G \left[ n \left( \hat{Q}_c(\hat{\eta}^{\text{PMLpm}}) - Q_c(\hat{\eta}^{\text{PMLpm}}) \right) \right] \\
&= E_G \left[ n\bar{Z}_n + \sqrt{n} A_{\eta}^T \phi_{n,\eta}^*(\eta_0^{\text{PMLpm}}) + \frac{1}{2} A_{\eta}^T H_{n,\eta}^*(\eta_0^{\text{PMLpm}}) A_{\eta} + \frac{1}{2} A_{\eta}^T \mathcal{I}_{\eta}^* A_{\eta} \right] + o_p(1) \\
&\xrightarrow{p} E_G \left[ \left( \left( \mathcal{I}_{\eta}^{\text{PMLpm}} \right)^{-1} \Lambda_{\eta^{2\text{ML}}} \right)^T \Lambda_{\eta}^* \right] \\
&= E_G \left[ \text{tr} \left( \left( \mathcal{I}_{\eta}^{\text{PMLpm}} \right)^{-1} \Lambda_{\eta^{\text{PMLpm}}} (\Lambda_{\eta}^*)^T \right) \right] \\
&= \text{tr} \left( \left( \mathcal{I}_{\eta}^{\text{PMLpm}} \right)^{-1} E_G \left[ \Lambda_{\eta^{\text{PMLpm}}} (\Lambda_{\eta}^*)^T \right] \right) \\
&= \text{tr} \left( \left( \mathcal{I}_{\eta}^{\text{PMLpm}} \right)^{-1} K_{\eta}^{\circ} \right),
\end{aligned}$$

where

$$K_{\eta}^{\circ} = E_G \left[ \Lambda_{\eta^{\text{PMLpm}}} (\Lambda_{\eta}^*)^T \right] = E_G \left[ \begin{pmatrix} \Lambda_{\alpha^{2\text{ML}}} (\Lambda_{\alpha}^*)^T & \Lambda_{\alpha^{2\text{ML}}} (\Lambda_{\theta}^*)^T \\ \Lambda_{\theta^{\text{PML}}} (\Lambda_{\alpha}^*)^T & \Lambda_{\theta^{\text{PML}}} (\Lambda_{\theta}^*)^T \end{pmatrix} \right] = \begin{pmatrix} K_{\alpha}^{\circ} & K_{\alpha,\theta}^{\circ} \\ K_{\theta,\alpha}^{\circ} & K_{\theta}^{\circ} \end{pmatrix}.$$

Furthermore, we assume that  $c = c_0$  such that the whole model is correctly specified. According to Genest et al. (1995), the true model assumption yields  $\text{Cov}_{C_0}(\phi_\theta^{\text{PML}}(u, \theta_0^{\text{PML}}), \mathcal{W}(u, \theta_0^{\text{PML}})) = 0$  and  $\mathcal{I}_\theta^{\text{PML}} = \Sigma_\theta^{\text{PML}}$ , where  $\Sigma_\theta^{\text{PML}} = E_{C_0}[(\phi_\eta^{\text{PMLpm}}(u, \eta_0^{\text{PMLpm}})) \cdot (\phi_\eta^{\text{PMLpm}}(u, \eta_0^{\text{PMLpm}}))^T]$ . Because we also assumed true margins, we have

$$\log c \left( F_1(y_1, \alpha_{0,1}^{2\text{ML}}), \dots, F_d(y_d, \alpha_{0,d}^{2\text{ML}}), \theta \right) = \log c(G_1(y_1), \dots, G_d(y_d), \theta).$$

This leads to  $K_\theta^\circ = \mathcal{I}_\theta^{\text{PML}}$ . In addition, Lemma 1 of Ko and Hjort (2019a) gives  $K_\alpha^\circ = \mathcal{I}_\alpha^{2\text{ML}} - K_\alpha^{2\text{ML}}$ .

By using those results, we have

$$\begin{aligned} \text{tr} \left( \left( \mathcal{I}_\eta^{\text{PMLpm}} \right)^{-1} K_\eta^\circ \right) &= \text{tr} \left( \begin{pmatrix} \mathcal{I}_\alpha^{2\text{ML}} & 0 \\ 0 & \mathcal{I}_\theta^{\text{PML}} \end{pmatrix}^{-1} \begin{pmatrix} K_\alpha^\circ & K_{\alpha,\theta}^\circ \\ K_{\theta,\alpha}^\circ & K_\theta^\circ \end{pmatrix} \right) \\ &= \text{tr} \left( \begin{pmatrix} (\mathcal{I}_\alpha^{2\text{ML}})^{-1} K_\alpha^\circ & (\mathcal{I}_\alpha^{2\text{ML}})^{-1} K_{\alpha,\theta}^\circ \\ (\mathcal{I}_\theta^{\text{PML}})^{-1} K_{\theta,\alpha}^\circ & (\mathcal{I}_\theta^{\text{PML}})^{-1} K_\theta^\circ \end{pmatrix} \right) \\ &= \text{tr} \left( \begin{pmatrix} (\mathcal{I}_\alpha^{2\text{ML}})^{-1} (\mathcal{I}_\alpha^{2\text{ML}} - K_\alpha^{2\text{ML}}) & 0 \\ 0 & (\mathcal{I}_\theta^{\text{PML}})^{-1} K_\theta^\circ \end{pmatrix} \right) \\ &= \text{tr} \left( \begin{pmatrix} 0 & 0 \\ 0 & I_{\dim(\theta)} \end{pmatrix} \right) \\ &= \dim(\theta). \end{aligned}$$

Thus, the unbiased estimator of  $Q(\hat{\eta}^{\text{PMLpm}})$  is

$$\hat{Q}(\hat{\eta}^{\text{PMLpm}}) = \frac{1}{n} \ell_n(\hat{\eta}^{\text{PMLpm}}) - \frac{1}{n} \dim(\eta).$$

By scaling this unbiased estimator with  $2n$  such that it is on the same scale as the AIC for ML (Akaike, 1974), we define the AIC for the copula model estimated with PML with parametric margins

$$\text{AIC} = 2\ell_n(\hat{\eta}^{\text{PMLpm}}) - 2 \dim(\eta).$$

□

---

# Supplementary Material:

## Focused information criteria for copulas

Vinnie Ko<sup>1</sup> | Nils Lid Hjort<sup>1</sup> | Ingrid Hobæk Haff<sup>1</sup>

<sup>1</sup>Department of Mathematics, University of Oslo, PB 1053, Blindern, NO-0316 Oslo, Norway

### Correspondence

Vinnie Ko, Department of Mathematics,  
University of Oslo, PB 1053, Blindern,  
NO-0316 Oslo, Norway  
Email: vinniebk@math.uio.no

### Funding information

This is a supplement to Ko et al. (2019). Sections 1, 2, 3 provide tables referred to in Sections 5.1, 5.2, 5.3 in that paper. Section 4 provides a figure referred to in Section 5.2 in that paper.

# 1 | TABLES FROM SECTION 5.1

**TABLE 1** Results from Part 1 of the simulations with  $n = 100$ . The results are averaged.

| Model no. | Estimation | $\hat{T}$ | $10^{-4} \cdot \text{MSE}(\hat{T})$ | $10^{-4} \cdot \text{FIC}$ | AIC     | Rank <sub>FIC</sub> | Rank <sub>FIC</sub> <sup>np</sup> | Rank <sub>AIC</sub> | Copula       | Margin 1 | Margin 2 | Margin 3 | Margin 4   |
|-----------|------------|-----------|-------------------------------------|----------------------------|---------|---------------------|-----------------------------------|---------------------|--------------|----------|----------|----------|------------|
| 1         | ML         | 0.1323    | 4.7453                              | 8.0417                     | -211.93 | 1                   | 1                                 | 1                   | Gumbel       | Gamma    | Gamma    | Weibull  | Log-normal |
| 1         | 2ML        | 0.1312    | 5.0293                              | 8.0484                     | -215.97 | 2                   | 2                                 | 2                   | Gumbel       | Gamma    | Gamma    | Weibull  | Log-normal |
| 3         | 2ML        | 0.1176    | 7.0372                              | 8.2209                     | -301.24 | 3                   | 3                                 | 14                  | Surv.Clayton | Gamma    | Gamma    | Weibull  | Log-normal |
| 2         | 2ML        | 0.1342    | 5.6744                              | 8.5046                     | -236.90 | 4                   | 4                                 | 5                   | Gumbel       | Weibull  | Weibull  | Gamma    | Gamma      |
| 4         | 2ML        | 0.1154    | 8.1284                              | 8.5721                     | -332.28 | 5                   | 5                                 | 17                  | Surv.Clayton | Weibull  | Weibull  | Gamma    | Gamma      |
| 1         | PML-pm     | 0.1315    | 4.9445                              | 8.6823                     | -216.31 | 6                   | 6                                 | 3                   | Gumbel       | Gamma    | Gamma    | Weibull  | Log-normal |
| 2         | ML         | 0.1398    | 5.5992                              | 8.9757                     | -228.40 | 7                   | 7                                 | 4                   | Gumbel       | Weibull  | Weibull  | Gamma    | Gamma      |
| 3         | ML         | 0.1207    | 7.1126                              | 9.6686                     | -294.28 | 8                   | 8                                 | 13                  | Surv.Clayton | Gamma    | Gamma    | Weibull  | Log-normal |
| 3         | PML-pm     | 0.1183    | 6.8651                              | 9.7290                     | -302.66 | 9                   | 9                                 | 15                  | Surv.Clayton | Gamma    | Gamma    | Weibull  | Log-normal |
| 2         | PML-pm     | 0.1357    | 5.7744                              | 9.8151                     | -237.68 | 10                  | 10                                | 6                   | Gumbel       | Weibull  | Weibull  | Gamma    | Gamma      |
| 4         | PML-pm     | 0.1221    | 6.6745                              | 10.0568                    | -336.40 | 11                  | 11                                | 18                  | Surv.Clayton | Weibull  | Weibull  | Gamma    | Gamma      |
| 4         | ML         | 0.1178    | 8.1122                              | 10.3724                    | -315.53 | 12                  | 12                                | 16                  | Surv.Clayton | Weibull  | Weibull  | Gamma    | Gamma      |
| np        | np         | 0.1329    | 7.7549                              | 11.4463                    | -       | 13                  | -                                 | -                   | np           | np       | np       | np       | np         |
| 6         | 2ML        | 0.1059    | 15.1220                             | 19.1944                    | -273.76 | 14                  | 13                                | 11                  | Frank        | Weibull  | Weibull  | Gamma    | Gamma      |
| 6         | PML-pm     | 0.1037    | 15.8880                             | 20.6127                    | -275.04 | 15                  | 14                                | 12                  | Frank        | Weibull  | Weibull  | Gamma    | Gamma      |
| 5         | 2ML        | 0.0992    | 17.8678                             | 21.2891                    | -272.82 | 16                  | 15                                | 9                   | Frank        | Gamma    | Gamma    | Weibull  | Log-normal |
| 5         | PML-pm     | 0.0985    | 18.1207                             | 22.1375                    | -273.56 | 17                  | 16                                | 10                  | Frank        | Gamma    | Gamma    | Weibull  | Log-normal |
| 6         | ML         | 0.1771    | 48.7422                             | 53.4333                    | -262.81 | 18                  | 17                                | 8                   | Frank        | Weibull  | Weibull  | Gamma    | Gamma      |
| 5         | ML         | 0.1803    | 51.4541                             | 56.3131                    | -262.59 | 19                  | 18                                | 7                   | Frank        | Gamma    | Gamma    | Weibull  | Log-normal |

**TABLE 2** Results from Part 1 of the simulations with  $n = 1000$ . The results are averaged.

| Model no. | Estimation | $\hat{T}$ | $10^{-4} \cdot \text{MSE}(\hat{T})$ | $10^{-4} \cdot \text{FIC}$ | AIC      | Rank <sub>FIC</sub> | Rank <sub>FIC</sub> <sup>np</sup> | Rank <sub>AIC</sub> | Copula       | Margin 1 | Margin 2 | Margin 3 | Margin 4   |
|-----------|------------|-----------|-------------------------------------|----------------------------|----------|---------------------|-----------------------------------|---------------------|--------------|----------|----------|----------|------------|
| 1         | 2ML        | 0.1328    | 0.5652                              | 0.8786                     | -2073.55 | 1                   | 1                                 | 2                   | Gumbel       | Gamma    | Gamma    | Weibull  | Log-normal |
| 1         | ML         | 0.1329    | 0.5018                              | 0.8813                     | -2069.36 | 2                   | 2                                 | 1                   | Gumbel       | Gamma    | Gamma    | Weibull  | Log-normal |
| 1         | PML.pm     | 0.1328    | 0.5645                              | 0.9402                     | -2073.61 | 3                   | 3                                 | 3                   | Gumbel       | Gamma    | Gamma    | Weibull  | Log-normal |
| 2         | 2ML        | 0.1361    | 0.7877                              | 0.9937                     | -2308.32 | 4                   | 4                                 | 5                   | Gumbel       | Weibull  | Weibull  | Gamma    | Gamma      |
| np        | np         | 0.1327    | 0.9250                              | 1.1499                     | -        | 5                   | -                                 | -                   | np           | np       | np       | np       | np         |
| 2         | PML.pm     | 0.1375    | 0.9330                              | 1.2455                     | -2310.28 | 6                   | 5                                 | 6                   | Gumbel       | Weibull  | Weibull  | Gamma    | Gamma      |
| 2         | ML         | 0.1417    | 1.4708                              | 1.6146                     | -2262.54 | 7                   | 6                                 | 4                   | Gumbel       | Weibull  | Weibull  | Gamma    | Gamma      |
| 4         | PML.pm     | 0.1224    | 1.5327                              | 2.0082                     | -3339.62 | 8                   | 7                                 | 18                  | Surv.Clayton | Weibull  | Weibull  | Gamma    | Gamma      |
| 3         | ML         | 0.1195    | 2.0818                              | 2.5168                     | -2946.18 | 9                   | 8                                 | 13                  | Surv.Clayton | Gamma    | Gamma    | Weibull  | Log-normal |
| 3         | 2ML        | 0.1182    | 2.4213                              | 2.5317                     | -2969.28 | 10                  | 9                                 | 14                  | Surv.Clayton | Gamma    | Gamma    | Weibull  | Log-normal |
| 3         | PML.pm     | 0.1183    | 2.3987                              | 2.8321                     | -2970.06 | 11                  | 10                                | 15                  | Surv.Clayton | Gamma    | Gamma    | Weibull  | Log-normal |
| 4         | 2ML        | 0.1157    | 3.1902                              | 3.1229                     | -3314.20 | 12                  | 11                                | 17                  | Surv.Clayton | Weibull  | Weibull  | Gamma    | Gamma      |
| 4         | ML         | 0.1167    | 2.8793                              | 3.3248                     | -3193.82 | 13                  | 12                                | 16                  | Surv.Clayton | Weibull  | Weibull  | Gamma    | Gamma      |
| 6         | 2ML        | 0.1073    | 7.0037                              | 7.6417                     | -2687.17 | 14                  | 13                                | 11                  | Frank        | Weibull  | Weibull  | Gamma    | Gamma      |
| 6         | PML.pm     | 0.1057    | 7.7933                              | 8.5109                     | -2689.18 | 15                  | 14                                | 12                  | Frank        | Weibull  | Weibull  | Gamma    | Gamma      |
| 5         | 2ML        | 0.0999    | 11.0561                             | 11.7567                    | -2669.70 | 16                  | 15                                | 9                   | Frank        | Gamma    | Gamma    | Weibull  | Log-normal |
| 5         | PML.pm     | 0.1000    | 11.0542                             | 11.8092                    | -2670.15 | 17                  | 16                                | 10                  | Frank        | Gamma    | Gamma    | Weibull  | Log-normal |
| 6         | ML         | 0.1827    | 28.7108                             | 28.7463                    | -2604.74 | 18                  | 17                                | 8                   | Frank        | Weibull  | Weibull  | Gamma    | Gamma      |
| 5         | ML         | 0.1831    | 28.9728                             | 28.9566                    | -2595.25 | 19                  | 18                                | 7                   | Frank        | Gamma    | Gamma    | Weibull  | Log-normal |

## 2 | TABLES FROM SECTION 5.2

**TABLE 3** Results from Part 2 of the simulations for the FIC with  $p = 0.90$ . The results are averaged.

| Model no. | Estimation | $\hat{T}$ | $10^{-4} \cdot \text{MSE}(\hat{T})$ | $10^{-4} \cdot \text{FIC}$ | AIC      | $\text{Rank}_{\text{FIC}}$ | $\text{Rank}_{\text{FIC}}^{\text{np}}$ | $\text{Rank}_{\text{AIC}}$ | Copula       | Margin 1   | Margin 2   |
|-----------|------------|-----------|-------------------------------------|----------------------------|----------|----------------------------|----------------------------------------|----------------------------|--------------|------------|------------|
| 1         | 2ML        | 0.3882    | 5.6478                              | 16.1254                    | -6871.78 | 1                          | 1                                      | 2                          | Gaussian     | Log-normal | Log-normal |
| 1         | ML         | 0.3882    | 5.6478                              | 16.5589                    | -6871.78 | 2                          | 2                                      | 1                          | Gaussian     | Log-normal | Log-normal |
| 1         | PML.pm     | 0.3898    | 5.5264                              | 16.6036                    | -6871.80 | 3                          | 3                                      | 3                          | Gaussian     | Log-normal | Log-normal |
| 11        | 2ML        | 0.3762    | 11.7738                             | 17.9116                    | -7237.69 | 4                          | 4                                      | 29                         | Surv.Clayton | Gamma      | Gamma      |
| 2         | 2ML        | 0.3833    | 8.6586                              | 18.0289                    | -7055.26 | 5                          | 5                                      | 14                         | Gaussian     | Gamma      | Gamma      |
| 12        | 2ML        | 0.3751    | 12.5833                             | 18.0651                    | -7403.53 | 6                          | 6                                      | 35                         | Surv.Clayton | Weibull    | Weibull    |
| 2         | ML         | 0.3853    | 8.4380                              | 18.8414                    | -7055.18 | 7                          | 7                                      | 13                         | Gaussian     | Gamma      | Gamma      |
| 2         | PML.pm     | 0.3896    | 8.1054                              | 20.4592                    | -7055.79 | 8                          | 8                                      | 15                         | Gaussian     | Gamma      | Gamma      |
| 12        | ML         | 0.3828    | 11.7219                             | 21.7556                    | -7401.77 | 9                          | 9                                      | 34                         | Surv.Clayton | Weibull    | Weibull    |
| 6         | ML         | 0.3844    | 12.1560                             | 23.0150                    | -7194.21 | 10                         | 10                                     | 22                         | Frank        | Weibull    | Weibull    |
| 3         | PML.pm     | 0.4032    | 9.6180                              | 23.0546                    | -7204.27 | 11                         | 11                                     | 26                         | Gaussian     | Weibull    | Weibull    |
| 11        | ML         | 0.3696    | 14.6960                             | 23.4908                    | -7236.84 | 12                         | 12                                     | 28                         | Surv.Clayton | Gamma      | Gamma      |
| np        | np         | 0.3868    | 24.8964                             | 23.8918                    | -        | 13                         | -                                      | -                          | np           | np         | np         |
| 6         | 2ML        | 0.3617    | 19.1963                             | 28.4171                    | -7198.43 | 14                         | 13                                     | 24                         | Frank        | Weibull    | Weibull    |
| 3         | 2ML        | 0.4224    | 19.3920                             | 29.4495                    | -7201.46 | 15                         | 14                                     | 25                         | Gaussian     | Weibull    | Weibull    |
| 8         | 2ML        | 0.4419    | 35.4811                             | 44.7175                    | -7178.64 | 16                         | 15                                     | 20                         | Gumbel       | Gamma      | Gamma      |
| 5         | ML         | 0.3264    | 51.1712                             | 51.4565                    | -7073.03 | 17                         | 16                                     | 16                         | Frank        | Gamma      | Gamma      |
| 6         | PML.pm     | 0.3269    | 49.1468                             | 51.9458                    | -7208.65 | 18                         | 17                                     | 27                         | Frank        | Weibull    | Weibull    |
| 3         | ML         | 0.4457    | 40.0698                             | 52.4882                    | -7197.51 | 19                         | 18                                     | 23                         | Gaussian     | Weibull    | Weibull    |
| 9         | 2ML        | 0.4575    | 54.9298                             | 62.1409                    | -7342.12 | 20                         | 19                                     | 32                         | Gumbel       | Weibull    | Weibull    |
| 4         | ML         | 0.3121    | 67.0025                             | 65.1113                    | -6909.88 | 21                         | 20                                     | 4                          | Frank        | Log-normal | Log-normal |
| 8         | ML         | 0.4583    | 55.9058                             | 69.9129                    | -7176.83 | 22                         | 21                                     | 19                         | Gumbel       | Gamma      | Gamma      |
| 5         | 2ML        | 0.3097    | 74.3431                             | 72.9594                    | -7074.76 | 23                         | 22                                     | 17                         | Frank        | Gamma      | Gamma      |
| 4         | PML.pm     | 0.3044    | 79.2754                             | 77.3873                    | -6910.60 | 24                         | 23                                     | 6                          | Frank        | Log-normal | Log-normal |
| 4         | 2ML        | 0.3042    | 79.6418                             | 77.5446                    | -6910.55 | 25                         | 24                                     | 5                          | Frank        | Log-normal | Log-normal |
| 5         | PML.pm     | 0.3012    | 87.9706                             | 86.5350                    | -7075.58 | 26                         | 25                                     | 18                         | Frank        | Gamma      | Gamma      |
| 10        | 2ML        | 0.4951    | 117.6930                            | 125.3261                   | -6988.12 | 27                         | 26                                     | 11                         | Surv.Clayton | Log-normal | Log-normal |
| 7         | 2ML        | 0.4983    | 122.7807                            | 131.7212                   | -6922.68 | 28                         | 27                                     | 8                          | Gumbel       | Log-normal | Log-normal |
| 10        | PML.pm     | 0.4979    | 122.9565                            | 133.5784                   | -6988.51 | 29                         | 28                                     | 12                         | Surv.Clayton | Log-normal | Log-normal |
| 7         | PML.pm     | 0.4999    | 126.0077                            | 136.9588                   | -6922.81 | 30                         | 29                                     | 9                          | Gumbel       | Log-normal | Log-normal |
| 12        | PML.pm     | 0.5011    | 132.4155                            | 145.6955                   | -7449.37 | 31                         | 30                                     | 36                         | Surv.Clayton | Weibull    | Weibull    |
| 11        | PML.pm     | 0.5013    | 133.2260                            | 145.9652                   | -7284.64 | 32                         | 31                                     | 30                         | Surv.Clayton | Gamma      | Gamma      |
| 8         | PML.pm     | 0.5030    | 135.9110                            | 148.9554                   | -7193.07 | 33                         | 32                                     | 21                         | Gumbel       | Gamma      | Gamma      |
| 10        | ML         | 0.5038    | 137.1085                            | 149.7742                   | -6985.99 | 34                         | 33                                     | 10                         | Surv.Clayton | Log-normal | Log-normal |
| 9         | PML.pm     | 0.5041    | 138.2748                            | 151.9813                   | -7350.06 | 35                         | 34                                     | 33                         | Gumbel       | Weibull    | Weibull    |
| 9         | ML         | 0.5040    | 139.9657                            | 155.2434                   | -7333.02 | 36                         | 35                                     | 31                         | Gumbel       | Weibull    | Weibull    |
| 7         | ML         | 0.5108    | 151.7796                            | 164.4415                   | -6921.32 | 37                         | 36                                     | 7                          | Gumbel       | Log-normal | Log-normal |

**TABLE 4** Results from Part 2 of the simulations for the FIC with  $p = 0.95$ . The results are averaged.

| Model no. | Estimation | $\hat{T}$ | $10^{-4} \cdot \text{MSE}(\hat{T})$ | $10^{-4} \cdot \text{FIC}$ | AIC      | Rank <sub>FIC</sub> | Rank <sub>FIC</sub> <sup>np</sup> | Rank <sub>AIC</sub> | Copula       | Margin 1   | Margin 2   |
|-----------|------------|-----------|-------------------------------------|----------------------------|----------|---------------------|-----------------------------------|---------------------|--------------|------------|------------|
| 1         | 2ML        | 0.3086    | 6.3295                              | 20.3392                    | -6871.78 | 1                   | 1                                 | 2                   | Gaussian     | Log-normal | Log-normal |
| 1         | ML         | 0.3086    | 6.3295                              | 20.6200                    | -6871.78 | 2                   | 2                                 | 1                   | Gaussian     | Log-normal | Log-normal |
| 1         | PML.pm     | 0.3102    | 6.2141                              | 20.9331                    | -6871.80 | 3                   | 3                                 | 3                   | Gaussian     | Log-normal | Log-normal |
| 3         | 2ML        | 0.3127    | 12.8982                             | 27.9917                    | -7201.46 | 4                   | 4                                 | 25                  | Gaussian     | Weibull    | Weibull    |
| 3         | PML.pm     | 0.2926    | 14.0941                             | 28.9935                    | -7204.27 | 5                   | 5                                 | 26                  | Gaussian     | Weibull    | Weibull    |
| 2         | PML.pm     | 0.2776    | 20.1145                             | 29.7261                    | -7055.79 | 6                   | 6                                 | 15                  | Gaussian     | Gamma      | Gamma      |
| 2         | 2ML        | 0.2712    | 24.6047                             | 30.5281                    | -7055.26 | 7                   | 7                                 | 14                  | Gaussian     | Gamma      | Gamma      |
| 2         | ML         | 0.2731    | 23.2402                             | 30.5653                    | -7055.18 | 8                   | 8                                 | 13                  | Gaussian     | Gamma      | Gamma      |
| 11        | 2ML        | 0.3293    | 18.3423                             | 30.7280                    | -7237.69 | 9                   | 9                                 | 29                  | Surv.Clayton | Gamma      | Gamma      |
| 12        | 2ML        | 0.3290    | 20.8156                             | 32.3012                    | -7403.53 | 10                  | 10                                | 35                  | Surv.Clayton | Weibull    | Weibull    |
| 11        | ML         | 0.3233    | 17.9161                             | 33.9353                    | -7236.84 | 11                  | 11                                | 28                  | Surv.Clayton | Gamma      | Gamma      |
| 3         | ML         | 0.3412    | 23.0161                             | 39.6736                    | -7197.51 | 12                  | 12                                | 23                  | Gaussian     | Weibull    | Weibull    |
| 12        | ML         | 0.3386    | 27.7021                             | 42.4621                    | -7401.77 | 13                  | 13                                | 34                  | Surv.Clayton | Weibull    | Weibull    |
| np        | np         | 0.3065    | 41.2571                             | 43.6136                    | -        | 14                  | -                                 | -                   | np           | np         | np         |
| 8         | 2ML        | 0.4049    | 102.8917                            | 108.4323                   | -7178.64 | 15                  | 14                                | 20                  | Gumbel       | Gamma      | Gamma      |
| 6         | ML         | 0.1983    | 136.2531                            | 127.7135                   | -7194.21 | 16                  | 15                                | 22                  | Frank        | Weibull    | Weibull    |
| 4         | ML         | 0.1850    | 161.6738                            | 148.5580                   | -6909.88 | 17                  | 16                                | 4                   | Frank        | Log-normal | Log-normal |
| 8         | ML         | 0.4210    | 137.7982                            | 148.9792                   | -7176.83 | 18                  | 17                                | 19                  | Gumbel       | Gamma      | Gamma      |
| 9         | 2ML        | 0.4243    | 147.6641                            | 150.5689                   | -7342.12 | 19                  | 18                                | 32                  | Gumbel       | Weibull    | Weibull    |
| 4         | 2ML        | 0.1776    | 180.3702                            | 166.3138                   | -6910.55 | 20                  | 19                                | 5                   | Frank        | Log-normal | Log-normal |
| 4         | PML.pm     | 0.1778    | 179.9556                            | 166.3638                   | -6910.60 | 21                  | 20                                | 6                   | Frank        | Log-normal | Log-normal |
| 6         | 2ML        | 0.1744    | 193.7918                            | 180.9205                   | -7198.43 | 22                  | 21                                | 24                  | Frank        | Weibull    | Weibull    |
| 5         | ML         | 0.1529    | 254.7066                            | 239.1068                   | -7073.03 | 23                  | 22                                | 16                  | Frank        | Gamma      | Gamma      |
| 6         | PML.pm     | 0.1532    | 253.6577                            | 239.2535                   | -7208.65 | 24                  | 23                                | 27                  | Frank        | Weibull    | Weibull    |
| 8         | PML.pm     | 0.4694    | 267.6935                            | 280.5936                   | -7193.07 | 25                  | 24                                | 21                  | Gumbel       | Gamma      | Gamma      |
| 5         | 2ML        | 0.1389    | 299.3806                            | 281.8677                   | -7074.76 | 26                  | 25                                | 17                  | Frank        | Gamma      | Gamma      |
| 7         | 2ML        | 0.4771    | 285.8960                            | 291.9153                   | -6922.68 | 27                  | 26                                | 8                   | Gumbel       | Log-normal | Log-normal |
| 9         | ML         | 0.4713    | 279.0023                            | 292.4440                   | -7333.02 | 28                  | 27                                | 31                  | Gumbel       | Weibull    | Weibull    |
| 11        | PML.pm     | 0.4732    | 282.4024                            | 295.9925                   | -7284.64 | 29                  | 28                                | 30                  | Surv.Clayton | Gamma      | Gamma      |
| 10        | 2ML        | 0.4784    | 292.8752                            | 297.8630                   | -6988.12 | 30                  | 29                                | 11                  | Surv.Clayton | Log-normal | Log-normal |
| 5         | PML.pm     | 0.1342    | 315.3396                            | 298.0236                   | -7075.58 | 31                  | 30                                | 18                  | Frank        | Gamma      | Gamma      |
| 7         | PML.pm     | 0.4788    | 291.2749                            | 300.6144                   | -6922.81 | 32                  | 31                                | 9                   | Gumbel       | Log-normal | Log-normal |
| 9         | PML.pm     | 0.4746    | 288.3063                            | 303.3531                   | -7350.06 | 33                  | 32                                | 33                  | Gumbel       | Weibull    | Weibull    |
| 10        | PML.pm     | 0.4815    | 302.4989                            | 312.7326                   | -6988.51 | 34                  | 33                                | 12                  | Surv.Clayton | Log-normal | Log-normal |
| 12        | PML.pm     | 0.4781    | 301.7405                            | 317.2543                   | -7449.37 | 35                  | 34                                | 36                  | Surv.Clayton | Weibull    | Weibull    |
| 10        | ML         | 0.4880    | 326.6800                            | 337.9244                   | -6985.99 | 36                  | 35                                | 10                  | Surv.Clayton | Log-normal | Log-normal |
| 7         | ML         | 0.4893    | 328.3196                            | 339.4458                   | -6921.32 | 37                  | 36                                | 7                   | Gumbel       | Log-normal | Log-normal |

**TABLE 5** Results from Part 2 of the simulations for the AFIC based on  $p \in \{0.90, 0.91, \dots, 0.95\}$ . The results are averaged.

| Model no. | Estimation | $\hat{T}$ | $10^{-4} \cdot \text{AMSE}(\hat{T})$ | $10^{-4} \cdot \text{AFIC}$ | AAIC     | Rank <sub>AFIC</sub> | Rank <sub>AFIC</sub> <sup>-np</sup> | Rank <sub>AAIC</sub> | Copula       | Margin 1   | Margin 2   |
|-----------|------------|-----------|--------------------------------------|-----------------------------|----------|----------------------|-------------------------------------|----------------------|--------------|------------|------------|
| 1         | 2ML        | 0.3506    | 6.0039                               | 13.4946                     | -6871.78 | 1                    | 1                                   | 2                    | Gaussian     | Log-normal | Log-normal |
| 1         | ML         | 0.3506    | 6.0039                               | 13.8241                     | -6871.78 | 2                    | 2                                   | 1                    | Gaussian     | Log-normal | Log-normal |
| 1         | PML.pm     | 0.3522    | 5.8839                               | 13.9772                     | -6871.80 | 3                    | 3                                   | 3                    | Gaussian     | Log-normal | Log-normal |
| 11        | 2ML        | 0.3535    | 13.2986                              | 17.7845                     | -7237.69 | 4                    | 4                                   | 29                   | Surv.Clayton | Gamma      | Gamma      |
| 2         | 2ML        | 0.3309    | 14.7284                              | 18.1211                     | -7055.26 | 5                    | 5                                   | 14                   | Gaussian     | Gamma      | Gamma      |
| 12        | 2ML        | 0.3525    | 14.6442                              | 18.1680                     | -7403.53 | 6                    | 6                                   | 35                   | Surv.Clayton | Weibull    | Weibull    |
| 2         | ML         | 0.3328    | 13.9672                              | 18.5785                     | -7055.18 | 7                    | 7                                   | 13                   | Gaussian     | Gamma      | Gamma      |
| 2         | PML.pm     | 0.3372    | 12.3515                              | 19.0526                     | -7055.79 | 8                    | 8                                   | 15                   | Gaussian     | Gamma      | Gamma      |
| 3         | PML.pm     | 0.3517    | 10.4913                              | 20.1153                     | -7204.27 | 9                    | 9                                   | 26                   | Gaussian     | Weibull    | Weibull    |
| 11        | ML         | 0.3472    | 14.6029                              | 21.4949                     | -7236.84 | 10                   | 10                                  | 28                   | Surv.Clayton | Gamma      | Gamma      |
| 3         | 2ML        | 0.3714    | 15.4486                              | 22.1663                     | -7201.46 | 11                   | 11                                  | 25                   | Gaussian     | Weibull    | Weibull    |
| 12        | ML         | 0.3612    | 17.0221                              | 24.0154                     | -7401.77 | 12                   | 12                                  | 34                   | Surv.Clayton | Weibull    | Weibull    |
| np        | np         | 0.3490    | 30.6032                              | 32.1526                     | -        | 13                   | -                                   | -                    | np           | np         | np         |
| 3         | ML         | 0.3973    | 31.9702                              | 40.0586                     | -7197.51 | 14                   | 13                                  | 23                   | Gaussian     | Weibull    | Weibull    |
| 6         | ML         | 0.2970    | 55.9162                              | 52.9555                     | -7194.21 | 15                   | 14                                  | 22                   | Frank        | Weibull    | Weibull    |
| 8         | 2ML        | 0.4237    | 63.5797                              | 65.4443                     | -7178.64 | 16                   | 15                                  | 20                   | Gumbel       | Gamma      | Gamma      |
| 6         | 2ML        | 0.2731    | 87.3558                              | 80.1738                     | -7198.43 | 17                   | 16                                  | 24                   | Frank        | Weibull    | Weibull    |
| 9         | 2ML        | 0.4407    | 93.2461                              | 93.9918                     | -7342.12 | 18                   | 17                                  | 32                   | Gumbel       | Weibull    | Weibull    |
| 8         | ML         | 0.4399    | 90.5960                              | 97.8883                     | -7176.83 | 19                   | 18                                  | 19                   | Gumbel       | Gamma      | Gamma      |
| 4         | ML         | 0.2512    | 110.5606                             | 100.5473                    | -6909.88 | 20                   | 19                                  | 4                    | Frank        | Log-normal | Log-normal |
| 4         | PML.pm     | 0.2436    | 126.1196                             | 115.5180                    | -6910.60 | 21                   | 20                                  | 6                    | Frank        | Log-normal | Log-normal |
| 4         | 2ML        | 0.2434    | 126.5241                             | 115.6843                    | -6910.55 | 22                   | 21                                  | 5                    | Frank        | Log-normal | Log-normal |
| 6         | PML.pm     | 0.2441    | 136.3455                             | 127.0975                    | -7208.65 | 23                   | 22                                  | 27                   | Frank        | Weibull    | Weibull    |
| 5         | ML         | 0.2432    | 138.9328                             | 128.6879                    | -7073.03 | 24                   | 23                                  | 16                   | Frank        | Gamma      | Gamma      |
| 5         | 2ML        | 0.2273    | 174.5574                             | 162.4928                    | -7074.76 | 25                   | 24                                  | 17                   | Frank        | Gamma      | Gamma      |
| 5         | PML.pm     | 0.2205    | 190.6471                             | 178.6701                    | -7075.58 | 26                   | 25                                  | 18                   | Frank        | Gamma      | Gamma      |
| 10        | 2ML        | 0.4868    | 193.7567                             | 197.3935                    | -6988.12 | 27                   | 26                                  | 11                   | Surv.Clayton | Log-normal | Log-normal |
| 7         | 2ML        | 0.4876    | 193.4383                             | 198.1757                    | -6922.68 | 28                   | 27                                  | 8                    | Gumbel       | Log-normal | Log-normal |
| 8         | PML.pm     | 0.4866    | 193.6220                             | 203.1375                    | -7193.07 | 29                   | 28                                  | 21                   | Gumbel       | Gamma      | Gamma      |
| 7         | PML.pm     | 0.4893    | 197.6532                             | 205.2236                    | -6922.81 | 30                   | 29                                  | 9                    | Gumbel       | Log-normal | Log-normal |
| 10        | PML.pm     | 0.4897    | 201.0107                             | 209.2503                    | -6988.51 | 31                   | 30                                  | 12                   | Surv.Clayton | Log-normal | Log-normal |
| 11        | PML.pm     | 0.4878    | 199.1494                             | 209.4146                    | -7284.64 | 32                   | 31                                  | 30                   | Surv.Clayton | Gamma      | Gamma      |
| 9         | ML         | 0.4877    | 199.6092                             | 210.3643                    | -7333.02 | 33                   | 32                                  | 31                   | Gumbel       | Weibull    | Weibull    |
| 9         | PML.pm     | 0.4892    | 202.2458                             | 212.8958                    | -7350.06 | 34                   | 33                                  | 33                   | Gumbel       | Weibull    | Weibull    |
| 12        | PML.pm     | 0.4895    | 205.2491                             | 216.5513                    | -7449.37 | 35                   | 34                                  | 36                   | Surv.Clayton | Weibull    | Weibull    |
| 10        | ML         | 0.4959    | 219.7997                             | 228.7910                    | -6985.99 | 36                   | 35                                  | 10                   | Surv.Clayton | Log-normal | Log-normal |
| 7         | ML         | 0.5000    | 228.7504                             | 237.6216                    | -6921.32 | 37                   | 36                                  | 7                    | Gumbel       | Log-normal | Log-normal |

3 | TABLE FROM SECTION 5.3

TABLE 6 Results for the models fitted to the abalone data.

| Model no. | Estimation | $\hat{\tau}$ | $10^{-4} \cdot \text{FIC}$ | AIC      | Rank <sub>FIC</sub> | Rank <sub>FIC</sub> <sup>np</sup> | Rank <sub>AIC</sub> | Copula   | Margin 1 | Margin 2 | Margin 3 | Margin 4   |
|-----------|------------|--------------|----------------------------|----------|---------------------|-----------------------------------|---------------------|----------|----------|----------|----------|------------|
| np        | np         | 0.8235       | 2.0354                     | -        | 1                   | -                                 | -                   | np       | np       | np       | np       | np         |
| 6         | 2ML        | 0.7902       | 10.2261                    | 24219.02 | 2                   | 1                                 | 10                  | Gaussian | Normal   | Normal   | Normal   | Log-normal |
| 3         | 2ML        | 0.7827       | 16.9604                    | 24839.29 | 3                   | 2                                 | 9                   | t        | Normal   | Normal   | Normal   | Log-normal |
| 5         | 2ML        | 0.7811       | 17.0115                    | 28994.80 | 4                   | 3                                 | 7                   | Gaussian | Weibull  | Weibull  | Weibull  | Log-normal |
| 6         | PML.pm     | 0.7804       | 17.7474                    | 21602.45 | 5                   | 4                                 | 16                  | Gaussian | Normal   | Normal   | Normal   | Log-normal |
| 3         | PML.pm     | 0.7761       | 21.5499                    | 22761.26 | 6                   | 5                                 | 15                  | t        | Normal   | Normal   | Normal   | Log-normal |
| 2         | 2ML        | 0.7734       | 24.1082                    | 30108.31 | 7                   | 6                                 | 3                   | t        | Weibull  | Weibull  | Weibull  | Log-normal |
| 5         | PML.pm     | 0.7730       | 24.6204                    | 28958.97 | 8                   | 7                                 | 8                   | Gaussian | Weibull  | Weibull  | Weibull  | Log-normal |
| 2         | PML.pm     | 0.7685       | 29.2406                    | 30074.78 | 9                   | 8                                 | 4                   | t        | Weibull  | Weibull  | Weibull  | Log-normal |
| 4         | 2ML        | 0.7432       | 63.7877                    | 29114.80 | 10                  | 9                                 | 5                   | Gaussian | Weibull  | Weibull  | Weibull  | NIG        |
| 1         | 2ML        | 0.7401       | 68.8440                    | 30257.06 | 11                  | 10                                | 1                   | t        | Weibull  | Weibull  | Weibull  | NIG        |
| 4         | PML.pm     | 0.7381       | 72.3696                    | 29079.53 | 12                  | 11                                | 6                   | Gaussian | Weibull  | Weibull  | Weibull  | NIG        |
| 1         | PML.pm     | 0.7369       | 74.2642                    | 30230.77 | 13                  | 12                                | 2                   | t        | Weibull  | Weibull  | Weibull  | NIG        |
| 7         | 2ML        | 0.9465       | 149.5032                   | 23729.98 | 14                  | 13                                | 11                  | Clayton  | Weibull  | Weibull  | Weibull  | NIG        |
| 9         | 2ML        | 0.9473       | 151.4692                   | 20314.83 | 15                  | 14                                | 17                  | Clayton  | Normal   | Normal   | Normal   | Log-normal |
| 7         | PML.pm     | 0.9528       | 165.3779                   | 23683.19 | 16                  | 15                                | 12                  | Clayton  | Weibull  | Weibull  | Weibull  | NIG        |
| 8         | 2ML        | 0.9556       | 172.6200                   | 23107.05 | 17                  | 16                                | 13                  | Clayton  | Weibull  | Weibull  | Weibull  | Log-normal |
| 8         | PML.pm     | 0.9632       | 193.2900                   | 23026.99 | 18                  | 17                                | 14                  | Clayton  | Weibull  | Weibull  | Weibull  | Log-normal |
| 9         | PML.pm     | 0.9647       | 197.4406                   | 19942.89 | 19                  | 18                                | 18                  | Clayton  | Normal   | Normal   | Normal   | Log-normal |

## 4 | FIGURE FROM SECTION 5.2

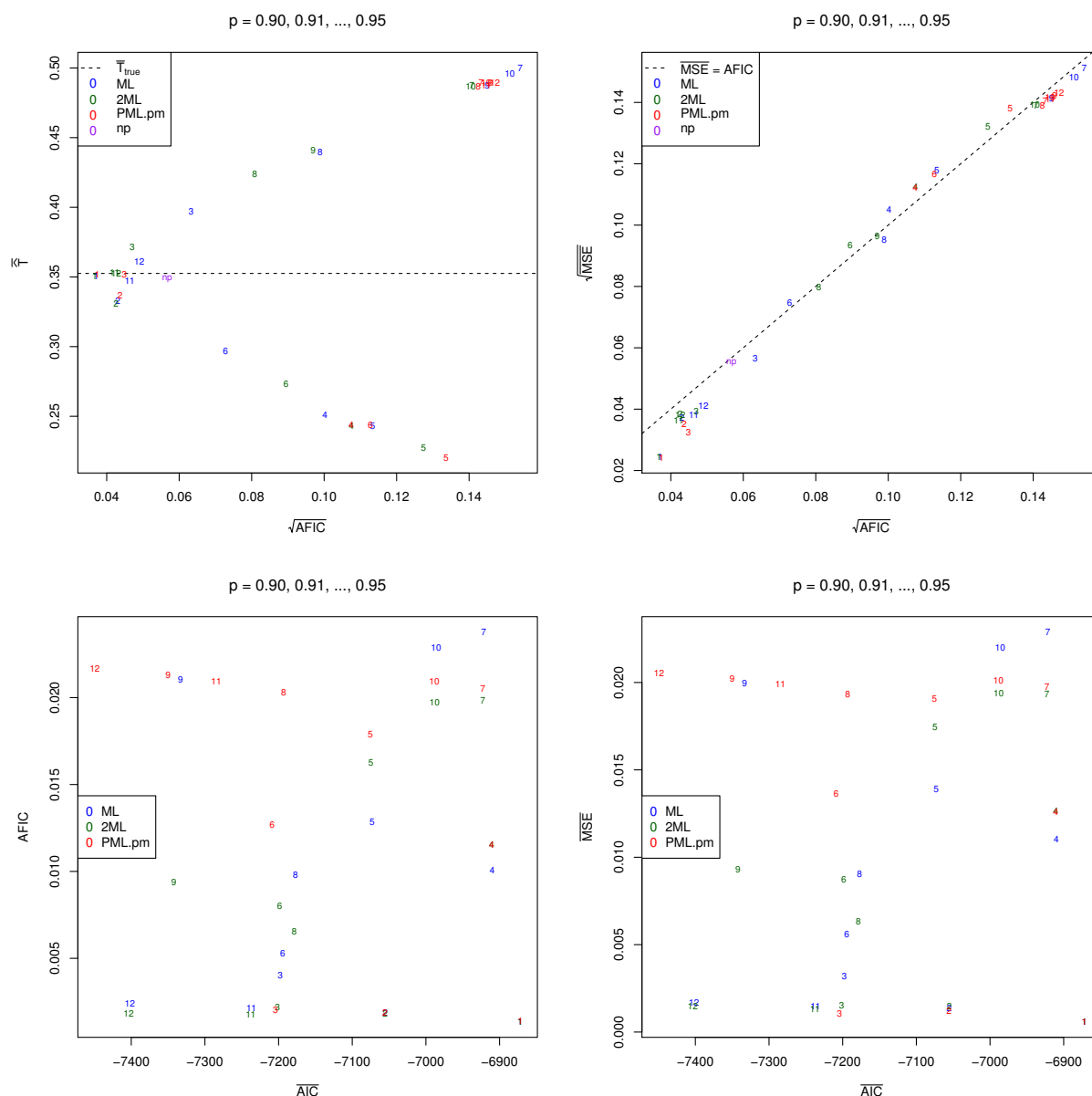


**FIGURE 1** Results from Part 2 of the simulations for the AFIC. The colored numbers refer to the model numbers in Table 2 of Ko et al. (2019).  $\overline{AIC}$  indicates the average of AIC across focus parameters in the set. Similarly,  $\overline{MSE}$  indicates the average MSE.

## REFERENCES

Ko, V., Hjort, N. L. and Hobæk Haff, I. (2019) Focused information criterion for copulas. *Scandinavian Journal of Statistics*.





# Magnitude and Uncertainty Pruning Criterion for Neural Networks

Vinnie Ko  
University of Oslo  
vinniebk@math.uio.no

Stefan Oehmcke  
University of Copenhagen  
stefan.oehmcke@di.ku.dk

Fabian Gieseke  
University of Copenhagen  
fabian.gieseke@di.ku.dk

## Abstract

*Neural networks have achieved dramatic improvements in recent years and depict the state-of-the-art methods for many real-world tasks nowadays. One drawback is, however, that many of these models are overparameterized, which makes them both computationally and memory intensive. In addition, overparameterization can also lead to undesired overfitting side-effects. Inspired by recently proposed magnitude-based pruning schemes and the Wald test from the field of statistics, we introduce a novel magnitude and uncertainty (M&U) pruning criterion. In addition, we present a “pseudo bootstrap” scheme, which can efficiently estimate the uncertainty of the weights by using their update information during training. Our experimental evaluation, which is based on various neural network architectures and datasets, shows that our new criterion leads to more compressed models compared to solely using a magnitude-based pruning criterion, with less loss in predictive power.*

## 1. Introduction

In recent years, deep neural networks have achieved state-of-the-art performance in a broad range of tasks and have gained big popularity in many fields like computer vision and natural language processing [19]. Typically, these powerful models have millions or even billions of parameters, which require significant computational and memory resources when deployed. For the training phase, this computational bottleneck can usually be overcome by using powerful hardware such as graphics processing units, which renders it possible to train big models based on a lot of data. However, during the inference phase, once the trained model has been deployed, one is often faced with very limited computational and memory resources. For instance, mobile or edge computing devices have little to no hardware acceleration available. In addition, many applications of neural networks—such as face recognition in video and autonomous driving—require the inference to be done with minimal latency.

A prominent approach to deal with these scenarios is *pruning*. Here, one usually generates a powerful model without any restrictions on its size and on computational requirements during the training phase. Afterwards, one “compresses” the model such that it fits into the (main) memory of the device that should be used for the inference phase. Pruning neural networks this way is a critical task for many real-world applications. In addition, Denil *et al.* [11] show that there is significant redundancy in the parameterization of many deep learning models. This overparameterization not only requires unnecessary computational resources, but can also lead to an overfitting of the models. Various approaches have been suggested in the literature that aim at decreasing the size and the computational footprint of neural networks [21, 39, 7, 47, 51]; see Cheng *et al.* [8] for an overview. One of the most successful works for pruning neural networks is provided by Han *et al.* [21], who prune network parameters based on their magnitude, measured as absolute value.

*Contribution:* Inspired by the *Wald test* from the field of statistics [46], we introduce a novel magnitude and uncertainty (M&U) pruning criterion, which depicts a generalization of both, the magnitude-based criterion used by Han *et al.* [21] and the Wald test statistic. We also propose a new “pseudo bootstrap” scheme that efficiently estimates the weight uncertainties based on update information obtained during the training process. Our experiments show that using our M&U pruning criterion outperforms the previous pruning criterion [21] and results in compressed models with less loss in predictive power.

## 2. Related Work

The attempt to prune neural networks started with LeCun *et al.* [36] and Hassibi *et al.* [23], who used a second-order Taylor expansion to determine the saliency of each parameter. Their methods require the computation of (the inverse of) the Hessian matrix regarding the loss function. Anders and Korn [1] consider pruning as a statistical model selection problem and apply diverse model selection tools from the field of maximum likelihood theory.

Since the recent success of deep learning, there has been

a number of new approaches for reducing the computation and memory footprint of neural networks. One approach, quantization, aims to reduce the numerical precision required to represent each parameter (or neuron). For example, Vanhoucke *et al.* [53], Hwang and Sung [25], and Gong *et al.* [17] compress networks by using 8-bit precision, 3-bit precision, and vector quantizations, respectively. There are more quantization based methods that expanded on these [9, 24, 38, 43]. Other strategies to reduce the computational burden resort to low-rank decompositions of tensors [12, 33, 26].

The strategy that received the most attention in recent years is probably the pruning of parameters based on certain criteria. Particularly, Han *et al.* [21] achieve state-of-the-art results by pruning weights based on their magnitudes (*i.e.*, based on the absolute values of the parameters) and by retraining the network afterwards. The same authors further improve on their results by combining their absolute value based pruning scheme with quantizing and Huffman encoding techniques [20]. Extensions of this pruning strategy have also been proposed for neural machine translation (NMT) [44] and recurrent neural networks (RNN) [40], respectively. Yet, those two follow-up works keep the same criterion as the one originally proposed by Han *et al.* [21] for ranking the weights to be pruned, *i.e.*, they use the absolute values of the weights. Finally, there are also attempts to prune specific classes of weights [2, 6, 34, 37]. This approach has an advantage that the resulting pruned networks have less irregular network connections and hence is more suitable for parallel computation.

### 3. Background

In this section, we first briefly recap the theory behind statistical hypothesis testing. Then, we look into the similarities between the Wald test statistic and the magnitude based criterion proposed by Han *et al.* [21].

#### 3.1. Asymptotic Normality

Consider an arbitrary regression model  $g$  with independent and dependent variable pairs  $(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_n, \mathbf{y}_n)$ . And let  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T$  denote a tensor containing all  $n$  observations of independent variables. The arbitrary regression model can then be written as

$$\mathbb{E}[\mathbf{Y}] = g(\mathbf{X}, \mathbf{w}), \quad (1)$$

where  $\mathbf{w} = [w_1, \dots, w_p]^T$  is a parameter vector of the model, which is commonly called weights in case of neural networks. The parameter  $p$  indicates the total number of parameters of the model.

For now, we assume that there exists a unique set of optimum parameters  $\mathbf{w}_*$ , which will give the best approximation of  $\mathbf{y}$ . We can then obtain  $\hat{\mathbf{w}}$ , our estimate of  $\mathbf{w}_*$ , by

minimizing a loss function with the optimization technique of choice (*e.g.*, stochastic gradient descent, mini-batch gradient descent, momentum based methods, etc.).

When the loss function is the negative log-likelihood, or an equivalent quantity such as cross entropy, gradient descent can be seen as a numerical realization of maximum likelihood (ML) estimation. In this case, we can apply the results from classic literature on ML theory [5, 32, 55] and we have for  $n \rightarrow \infty$ , the following convergence in distribution property:

$$\sqrt{n}(\hat{\mathbf{w}} - \mathbf{w}_*) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \mathbf{V}_{\text{ML}}). \quad (2)$$

Here  $\mathbf{V}_{\text{ML}}$  is the covariance matrix as defined in ML theory. In the field of statistics, this type of convergence to a normal distribution is called *asymptotic normality* [52]. In the model robust case,  $\mathbf{V}_{\text{ML}}$  is equal to the so-called sandwich estimator

$$\mathbf{V}_{\text{ML}} = \mathcal{I}_{\mathbf{w}}^{-1} \mathbf{K}_{\mathbf{w}} \mathcal{I}_{\mathbf{w}}^{-1}, \quad (3)$$

where  $\mathcal{I}_{\mathbf{w}}$  is the Fisher information and  $\mathbf{K}_{\mathbf{w}}$  is the covariance matrix of the score function [55]. In the field of statistics, the score function is defined as the first derivate of the log-likelihood with respect to its parameters. It indicates how sensitive a likelihood function is w.r.t. its parameters. For exact definitions of the quantities in Equation (3), we refer to the literature in maximum likelihood theory, for instance White [55]. When one assumes that the model is true, in other words, that the current candidate model is the true model that generated the data, we have  $\mathbf{K}_{\mathbf{w}} = \mathcal{I}_{\mathbf{w}}$  and Equation (3) simplifies to

$$\mathbf{V}_{\text{ML}} = \mathcal{I}_{\mathbf{w}}^{-1}. \quad (4)$$

For more details on the impact of this ‘true model’ assumption and the comparison between Equations (3) and (4), we refer to the corresponding literature [22, 27, 28].

Even though stochastic gradient descent (SGD) and its modified versions can be considered as numerical realization of ML estimation, the details of SGD lead to different properties than ML. The recent work by Toulis and Airolidi [50] obtains, for the first time, a full analytical characterization of the asymptotic behavior of SGD procedures, including an asymptotic normality and analytical expression for the covariance matrix, which is different from that of ML estimators. They also show the loss of asymptotic statistical efficiency for SGD estimators. Although the work of Toulis and Airolidi [50] is a great step forward towards understanding the asymptotic properties of SGD estimators, its result cannot be directly applied to non-vanilla versions of SGD without necessary extension work. For example, combining the SGD estimator with training strategies like Dropout [48] and Dropconnect [54] or using momentum based modifications such as RMSprop [49] or Adam [29]

would invalidate the asymptotic covariance matrix from Toulis and Airolidi [50].

Based on White [55] and Toulis and Airolidi [50], it is not unreasonable to assume that the modified versions of the SGD estimator are still consistent and have asymptotic normality with a finite, but unknown covariance matrix  $\mathbf{V}$ , *i.e.*, that we have

$$\sqrt{n}(\hat{\mathbf{w}} - \mathbf{w}_*) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \mathbf{V}). \quad (5)$$

### 3.2. Statistical Hypothesis Testing

With the asymptotic normality property as defined in Section 3.1, one can perform statistical hypothesis testing on the parameters  $\mathbf{w}$ . The natural null hypothesis would be

$$H_0 : w_j = 0, \quad (6)$$

*i.e.*, is the  $j$ -th parameter of the model significantly different from 0? The three commonly used statistical hypothesis testings are the *Wald test* [46], the *likelihood ratio test* [41], and the *score test* [3]. Note that all three tests are asymptotically equivalent [16]. Yet, the Wald test has a practical advantage over the other two since it does not require nested models for testing the null hypothesis on individual weights. Here, “nested” means that one can obtain one model by constraining some of the parameters of another model.

Given Equations (5) and (6), the Wald test statistic for the parameter  $w_j$  is given by

$$z = \frac{\hat{w}_j}{\text{SE}(\hat{w}_j)} = \frac{\hat{w}_j}{\sqrt{\hat{\mathbf{V}}_{j,j}}}, \quad (7)$$

where  $\text{SE}(\hat{w}_j)$  indicates the standard error of  $\hat{w}_j$  and  $\hat{\mathbf{V}}$  the estimate of the covariance matrix from Equation (5). (Since we defined  $\mathbf{V}$  as a finite, but unknown covariance matrix, its estimate  $\hat{\mathbf{V}}$  is an unknown and arbitrary estimate.) The test statistic  $z$  follows the standard normal distribution [16]. The corresponding  $p$ -value can be obtained via  $p = 2(1 - \Phi(|z|))$ , where  $\Phi$  is the cumulative distribution function of the standard normal distribution.

### 3.3. Wald Test and Absolute Value Based Pruning

In statistics, one of the most canonical ways of removing redundant parameters is by performing null hypothesis testing as described above and by dropping the parameters that have a  $p$ -value above certain threshold (*e.g.*, 0.05, 0.01). Since  $p = 2(1 - \Phi(|z|))$  is a monotonic function of  $|z|$ , one can skip the step of converting  $|z|$  into a  $p$ -value. Instead, one can directly use

$$|z| = \frac{|\hat{w}_j|}{\text{SE}(\hat{w}_j)} = \frac{|\hat{w}_j|}{\sqrt{\hat{\mathbf{V}}_{j,j}}}, \quad (8)$$

and compare it with the threshold values in  $|z|$ -scale, which can be obtained by transforming the threshold  $p$ -values via the function  $\Phi^{-1}(1 - p/2)$ .

Here, we can realize two things. Firstly, Equation (8) is highly similar to the magnitude based pruning criterion suggested by Han *et al.* [21] and also used by Han *et al.* [20], See *et al.* [44], and Narang *et al.* [40]. In those works, only the absolute value of each parameter is considered and the parameters with absolute values below a certain threshold are removed. We denote this criterion as the ABS pruning criterion from now on. The only difference between the Wald test statistic (Equation (8)) and the ABS pruning criterion is the standard error term in the denominator.

In this sense, the Wald test statistic can actually be interpreted as evaluating parameters based on their absolute value, while compensating for their uncertainty. For example, if  $\hat{w}_j$  has high uncertainty, there is higher chance that we observe a large value of  $\hat{w}_j$  “by chance”. Dividing by the standard error compensates for this uncertainty.

## 4. Magnitude and Uncertainty Pruning Criterion

Inspired by Wald test, we evaluate the importance of parameters by taking both their magnitudes and their uncertainties into account. By using this combined criterion, we can prune to the desired proportion of parameters in a model or a specific layer. Further, we also introduce a “pseudo bootstrap” approach to efficiently estimate the uncertainties of the weights by keeping track of their changes during the training process.

### 4.1. Definition

Based on our observation of the similarity between the Wald test statistic and the ABS pruning criterion in Section 3.3 and the interpretation of the Wald test statistic, we define our M&U pruning criterion as

$$\tau_j = \frac{|\hat{w}_j|}{\lambda + \tilde{\sigma}_{\hat{w}_j}}. \quad (9)$$

Here,  $\tilde{\sigma}_{\hat{w}_j}$  is an uncertainty estimate of  $\hat{w}_j$ , a quantity that should reflect information regarding how uncertain the estimated parameter  $\hat{w}_j$  is. We will cover possible choices for  $\tilde{\sigma}_{\hat{w}_j}$  in Sections 4.2 and 4.3.

The hyperparameter  $0 \leq \lambda$  is a tuning parameter that determines the balance between magnitude and uncertainty. When  $\lambda \rightarrow \infty$ , no uncertainty is taken into account and the M&U pruning criterion becomes equal to the ABS pruning criterion used by Han *et al.* [21]. As  $\lambda$  decreases, more and more uncertainty is taken into account. In case of  $\lambda = 0$ , the criterion  $\tau_j$  has the same functional form as the Wald test statistic. If one chooses  $\tilde{\sigma}_{\hat{w}_j}$  such that it is an analytically justified estimate of the standard error,  $\tau_j$  is simply the Wald

test statistic. This implies that M&U pruning criterion can be seen as a generalization of the Wald test statistic and the ABS pruning criterion used by Han *et al.* [21].

## 4.2. Choosing Uncertainty Estimate $\tilde{\sigma}_{\hat{w}_j}$

Ideally, one would use the standard error as an uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}$ , like in the original Wald test statistic formula. However, there is at the moment no asymptotic theories developed for non-convex neural networks with commonly used estimation methods such as Adam [29] and RMSprop [49].

In rare cases when a neural network is convex (or piecewise convex) with the negative log-likelihood, or an equivalent quantity such as cross entropy as loss function, one can take analytically derived asymptotic variance formulas such as Equation (3), Equation (4), or other ones [50] and estimate these uncertainty measures via their sample equivalent (e.g., by replacing the expectation with a sum over the samples). However, estimating those quantities has a number of numerical bottlenecks. For example, to obtain the sample equivalent of  $\mathcal{I}_w$ , one needs to calculate the Hessian matrix, which requires the second derivative of the log-likelihood function with respect to all the parameters in the model. Although modern deep learning softwares offer highly optimized automatic differentiation for the first derivative, many of the automatic differentiation implementations do not support the second derivative. In case they do support it, the corresponding implementations are not as highly optimized as those for the first derivative.

Another computational bottleneck is the computation of  $K_w$ . To estimate this matrix, one needs to compute, for every sample separately, the first derivative of the log-likelihood with respect to the model parameters. Automatic differentiation implementations of most deep learning software packages are optimized for mini-batch operations and do not offer the first derivative for each sample of the mini-batch separately. One can, in principle, obtain those derivatives by using a mini-batch size of 1 or by using the techniques suggested by Goodfellow [18]. However, both methods would add a significant amount of extra computations next to the regular forward and backward pass operations during training. Yet, the biggest computational obstacle is probably that one needs to take the inverse of  $\mathcal{I}_w$ . For a network with, say, a million parameters, inverting this matrix is infeasible in practice even using significant computational resources. Since deep neural networks can easily contain millions of parameters, this problem gets even worse.

Now, we introduce two possible alternatives to compute the uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}$ . The first alternative is to make use of bootstrapping [15]. There have been extensive studies done on both theoretical and numerical properties of this approach [4, 14, 10, 45] and it is a canonical method in statistics for estimating the standard error in case an ana-

**Algorithm 1** *Pseudo bootstrap*, our proposed novel algorithm for a fast estimation for uncertainty of each weight in neural network by using their changes during the training. The algorithm described below is for a single weight  $w_j$  and can be repeated for all the weights in the neural network.

---

**Require:**  $B \in \mathbb{N}$ : Weight collection size

**Require:**  $w = [w_1, \dots, w_p]$ : Parameter vector of the given neural network.

**Require:**  $n_{\text{iter}}$ : The total number of weight update iterations that is going to be made during the training phase. Note that this is the total number of iterations and not the total number of epochs.

```

1:  $w_{j,\{B\}}$ : An empty vector of length  $B$ .
2:  $i_{\text{iter}} \leftarrow 0$  (Initialize training iteration count)
3:  $i_B \leftarrow 0$  (Initialize weight collecting count)
4: while training the target model with mini-batch SGD do
5:    $i_{\text{iter}} \leftarrow i_{\text{iter}} + 1$ 
6:   if  $n_{\text{iter}} - B < i_{\text{iter}} \leq n_{\text{iter}}$  then
7:      $i_B \leftarrow i_B + 1$ 
8:      $w_{j,\{B\}}[i_B] \leftarrow w_j$  (Record current value of  $w_j$ .)
9:   end if
10: end while
11:  $\tilde{\sigma}_{\hat{w}_j} = \text{std}(w_{j,\{B\}})$  (Compute uncertainty estimate of  $\hat{w}_j$  by using its ‘fluctuations’ during the training.)
12: return  $\tilde{\sigma}_{\hat{w}_j}$ 

```

---

lytical estimate is not feasible. The disadvantage is that we need to train the same model multiple times, which is a big increase in the computational costs. It is a relief that this is parallelizable since each bootstrap sample can be trained independently. Because neural networks are usually trained with large number of observations, Jackknife resampling method [13], which requires repetition of training process equal to the number of total data points, is not suitable.

## 4.3. Pseudo Bootstrap

The second alternative is a novel estimation scheme, called *pseudo bootstrap*, which uses uncertainty information gathered during the training process. This idea stems from the fact that the training process with mini-batch SGD has similarities with the bootstrap resampling strategy: Each mini-batch we draw during training can be seen as a bootstrap sample with replacement and of smaller size. In addition, the iterative weight update process of mini-batch SGD can be seen as training with bootstrap samples, where each bootstrap training instance uses the parameter values from the previous instance as parameter initialization.

The pseudo bootstrap approach is sketched in Algorithm 1: To obtain uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}$ , the algorithm observes how  $\hat{w}_j$  is changing during the last  $B$  iterations of mini-batch SGD training and record it into ‘weight collection’:  $w_{j,\{B\}} = [w_{j,1}, \dots, w_{j,B}]$  (see Line 4 to 10). After the training, we simply obtain the uncertainty estimate with  $\tilde{\sigma}_{\hat{w}_j} = \text{std}(w_{\{B\}})$  (see Line 11). We also tried differ-

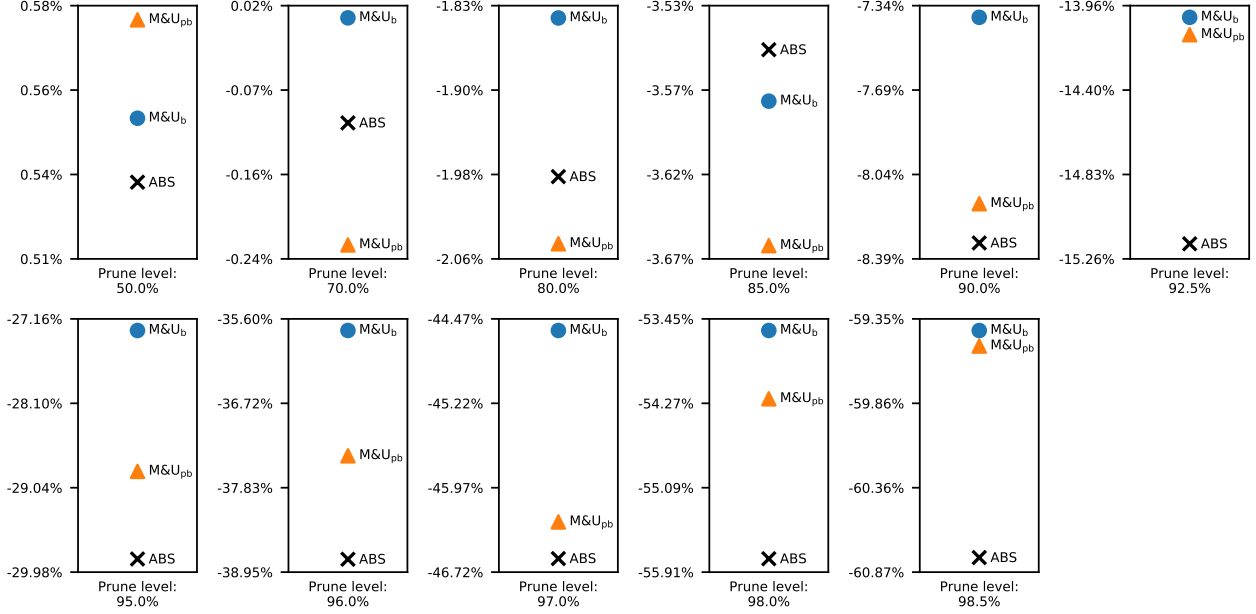


Figure 1. Result of Experiment 1. The  $y$ -axis of each plot shows the loss in test accuracy due to pruning (with retraining afterwards). It is computed as the test accuracy of the pruned model subtracted by the test accuracy of the unpruned original model. Blue circles with ‘M&U<sub>b</sub>’ indicate that our M&U pruning method with  $\tilde{\sigma}_{\tilde{w}_j}^{\text{boot}}$  is used. Orange triangles with ‘M&U<sub>pb</sub>’ mean that our M&U pruning method with  $\tilde{\sigma}_{\tilde{w}_j}^{\text{pboot}}$  is used. Black crosses with ‘ABS’ mean that the ABS pruning criterion from Han *et al.* [21] is used. The test accuracy of the unpruned model is 81.21%.

ent “collecting strategies” such as recording weight values through the entire training process with a certain interval between them. However, the best result was obtained by recording the weight values during the last iterations.

#### 4.4. Reparameterization of $\lambda$

The hyperparameter  $\lambda$  determines the tradeoff between magnitude and uncertainty. It is therefore crucial to consider reasonable values for  $\lambda$ . Note that the actual impact of  $\lambda$  on the balance between magnitude and uncertainty depends largely on which scale  $w_j$  is. For instance, if we decide to use batch normalization and the absolute value of  $w_j$  suddenly becomes much smaller, then a previously “optimal” value for  $\lambda$  (e.g.,  $\lambda = 1$ ) might not be optimal anymore.

To deal with this problem, we suggest to reparametrize  $\lambda$  via

$$\lambda = \lambda^* \cdot \text{std}(\mathbf{w}_{\{\text{layer}_k\}}) \quad (10)$$

when one prunes within layer  $k$ . Here,  $\lambda^*$  is a fixed global assignment for  $\lambda$  and  $\text{std}(\mathbf{w}_{\{\text{layer}_k\}})$  depicts the standard deviation of the weights with the  $k$ -th layer.<sup>1</sup>

<sup>1</sup>Naturally, instead of considering  $\mathbf{w}_{\{\text{layer}_k\}}$ , one can also consider the standard deviation based on all the weights via  $\mathbf{w}_{\{\text{model}\}}$  such that  $\lambda$  is scaled by the whole model (or by a specific part of a certain layer).

## 5. Experiments

We implement our M&U pruning criterion in PyTorch [42]. We evaluate with three experiments on Fashion MNIST, CIFAR-10 and MNIST dataset. Experiment 1 is repeated 100 times while Experiment 2 and 3 are repeated 20 times. The reported results are averaged results over those repetitions.

### 5.1. Experiment 1: Comparison of uncertainty estimates

The Fashion MNIST dataset consists of 28x28 grayscale images, each with a label from one of the 10 classes [56]. The training set contains 60,000 examples and the test set contains 10,000 examples. To make the bootstrap estimation computationally feasible, we have to keep the size of this experiment as small as possible. Thus, we use only a balanced subset of size 6,000 example from the training data. From this subset, 600 examples are set apart for validation during training. For testing, the entire test set is used. We employ a very small feedforward CNN with 2 convolution layers and 3 fully connected layers, combined with max pooling, ReLU activation function and dropout. The first fully connected layer gets a feature map of size 6 x 6 x 6 as input. The two hidden layers have 32 and 48 neurons

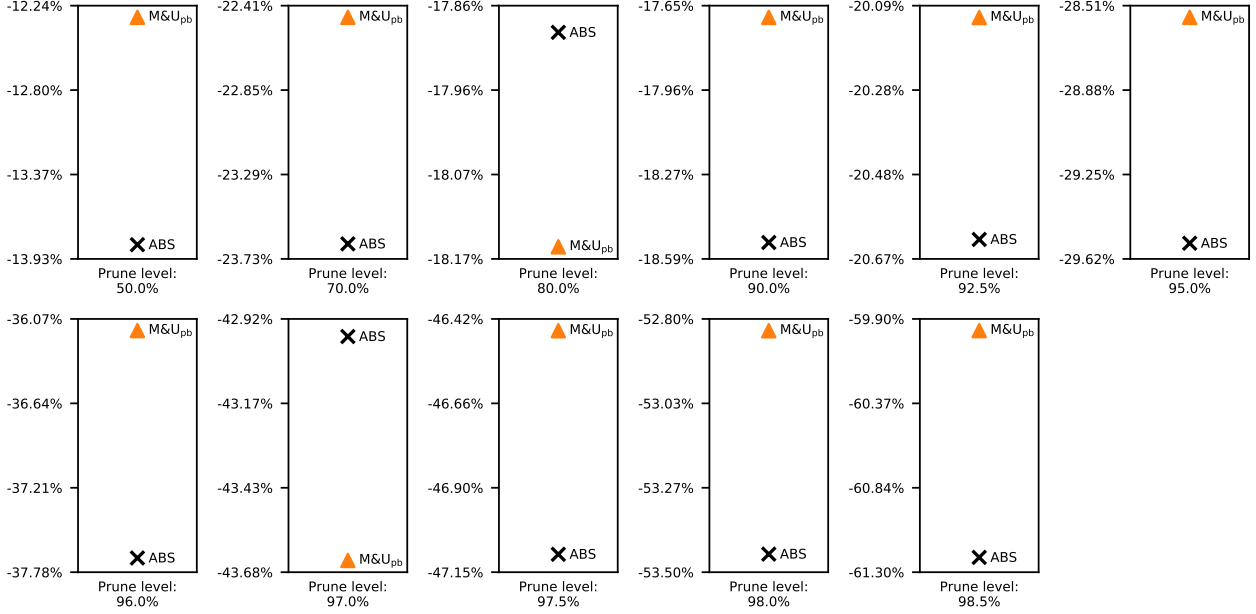


Figure 2. Result of Experiment 2. The  $y$ -axis of each plot shows the loss in test accuracy due to pruning (with retraining afterwards). It is computed as the test accuracy of the pruned model subtracted by the test accuracy of the unpruned original model. Orange triangles with ‘M&U<sub>pb</sub>’ mean that our M&U pruning method with  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$  is used. Black crosses with ‘ABS’ mean that the ABS pruning criterion from Han *et al.* [21] is used. The test accuracy of the unpruned model is 79.98%.

each. The model has a total amount of 9,660 parameters. We choose cross entropy as loss function. The model is trained for 500 epochs with RMSprop with  $\alpha = 0.99$ , initial learning rate of  $10^{-3}$  and a mini-batch size of 64. The learning rate decays by a factor of 0.75 every 20 epochs.

First, since the total number of parameters is small, we tried to estimate the covariance matrix of  $w$  with analytical expressions (3) and (4). However,  $\mathcal{I}_w$  and  $K_w$  were not invertible most of the time. As alternative, we obtained the pseudo inverse of those matrices. Yet, the pseudo inverse contained almost always negative diagonal elements. As a sanity check, we estimated the expectation of the score vector and many of its elements had values that were far away from 0. This suggests that the maximum likelihood theory does not hold in this case. This was expected, since many of the assumptions from LeCam [32] and Toulis and Airoldi [50], including convexity, are broken.

Next, we estimate  $\tilde{\sigma}_{\hat{w}_j}$  with bootstrap. We draw 100 bootstrap datasets and train 100 independent instances of the model with them. With the weights from those 100 trained models, we obtain the bootstrap based uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$ .

Lastly, we obtain the pseudo bootstrap based uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$ , with the method described in Section 4.3 and  $B = 200$ . We tried different values of  $\lambda^*$  and empirically choose  $\lambda^* = 10^{-1}$  for  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$  and  $\lambda^* = 10^{-4}$  for

$\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$ .

With the obtained  $\tilde{\sigma}_{\hat{w}_j}$  and  $\lambda$ , we prune  $a\%$  of parameters by using our M&U pruning criterion from each fully connected layer. The values of  $a$  we tried, can be found in Figure 1. For comparison, we also pruned the model by using the ABS pruning criterion from Han *et al.* [21]. After pruning, each model is retrained for 200 epochs.

Figure 1 shows the result of the experiment. One can observe that our M&U pruning criterion either with  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$  or  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$  gives a better (*i.e.*, smaller) test accuracy drop compared to the ABS pruning criterion. The difference is rather small when the pruning level is relatively low, but the gap between the two increases as the pruning level increases. Using  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$  gives the most optimal result (defeats ABS 10 out of 11 times), but using  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$  also gives better results than the ABS pruning criterion (defeats ABS 8 out of 11 times).

## 5.2. Experiment 2: CIFAR-10

We further examine the performance of our M&U pruning criterion on the CIFAR-10 dataset consisting of 32x32 RGB images, each with a label from one of the 10 classes [30]. The training set contains 50,000 examples and the test set contains 10,000 examples. Unlike Experiment 1, we utilize the whole training set. 5,000 examples of the

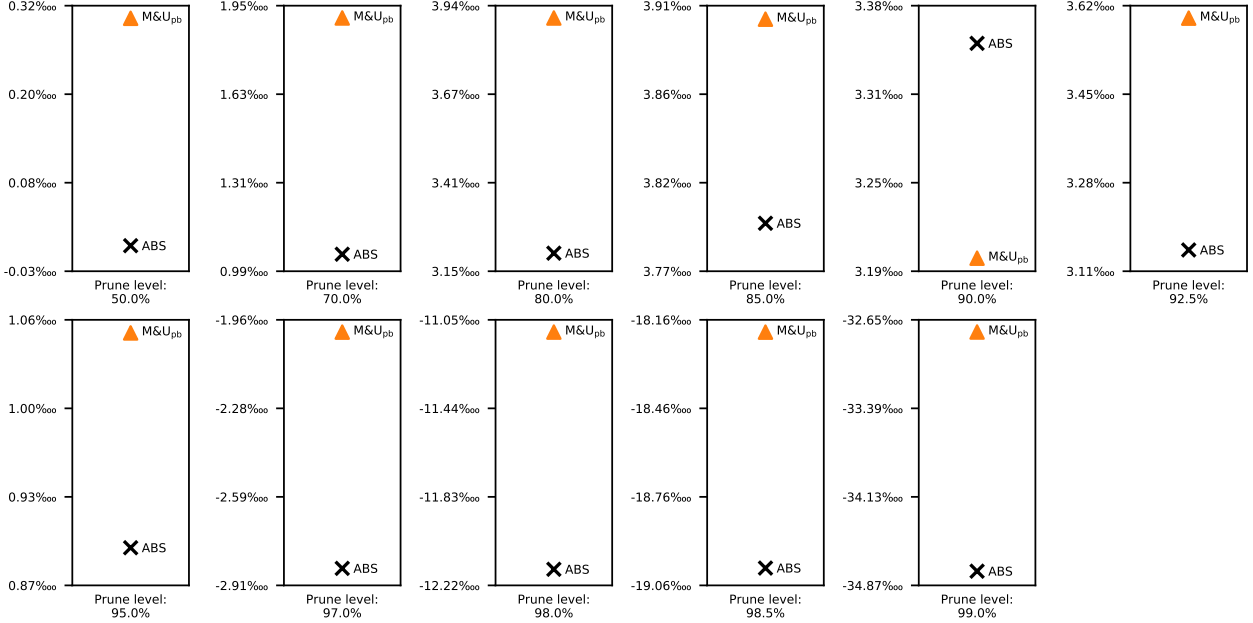


Figure 3. Result of Experiment 3. The  $y$ -axis of each plot shows the loss in test accuracy due to pruning (with retraining afterwards). It is computed as the test accuracy of the pruned model subtracted by the test accuracy of the unpruned original model. Note that the unit of the  $y$ -axis is basis point, and not percent. Orange triangles with ‘M&U<sub>pb</sub>’ mean that our M&U pruning method with  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$  is used. Black crosses with ‘ABS’ mean that the ABS pruning criterion from Han *et al.* [21] is used. The test accuracy of the unpruned model is 98.5160%.

training set are used as validation set during training. We employ a smaller version of AlexNet [31]. The model has 3 convolution layers with ReLU activation function and max pooling, which gives a feature map of size 128\*4\*4 as its output. The networks also has four hidden layers with 512, 512, 512, and 256 neurons each. The model has a total of 1,801,546 parameters. As the rapid drop in test accuracy suggests, this model has relatively low degree of overparameterization. We choose cross entropy as loss function. The model is trained for 1,000 epochs with RMSprop with the same configuration as in Experiment 1. The learning rate decays with a factor of 0.75 every 50 epochs.

Due to large amount of computation required for  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$ , we only use  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$  with  $B = 200$ . We empirically choose  $\lambda^* = 1$ .

We prune parameters of each fully connected layer by using our M&U pruning criterion or ABS pruning criterion with the pruning levels found in Figure 2. After pruning, the model is retrained for 200 epochs. Figure 2 shows the results of the experiment. Our M&U pruning criterion with  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$  gives better test accuracy than the criterion from Han *et al.* [21] in 9 out of 11 times.

### 5.3. Experiment 3: Overparameterization

To explore the case of high overparameterization, we use MNIST dataset [35] with a MLP model. The dataset consists of 28x28 grayscale images, each with a label from one of the 10 classes. The training set contains 60,000 examples and the test set contains 10,000 examples. We use 6,000 examples of the training set for validation during training. The MLP model has 3 hidden layers with 512, 1024, and 512 neurons each, which results in a total number of 1,457,162 parameters. The model is trained for 1,000 epochs with RMSprop with the same configuration as in Experiment 2, except the initial learning rate  $10^{-4}$ .

Due to the large amount of computation required for  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$ , we only use  $\tilde{\sigma}_{\hat{w}_j}^{\text{pboot}}$  with  $B = 200$ . We empirically choose  $\lambda^* = 1$ . For this experiment, we use the iterative pruning strategy described in Han *et al.* [21]. We prune the parameters of each fully connected layer by using our M&U pruning criterion or ABS pruning criterion with the pruning levels found in Figure 3. After pruning, the model is retrained for 200 epochs. Figure 3 shows the results of the experiment. Since the model is highly overparameterized, the performance drop is minimal, even when we pruned 99% of the weights. In fact, pruning up to 95% of weights increases the test accuracy of the model, possibly

because pruning decreased overfitting. Our M&U pruning criterion with  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$  gives better test accuracy than using ABS pruning criterion in 10 out of 11 times.

Note that since the model is highly overparameterized, one can prune a large number of the weights without sacrificing too much predictive power. For example, Figure 3 shows that we can prune 99% of the weights while losing only 0.33% of the test accuracy. (Note that the unit of the  $y$ -axis in Figure 3 is basis point, and not percent.) The performance difference between M&U pruning criterion and ABS pruning criterion might seem small due to this minimal test accuracy drop throughout all pruning levels. But when one takes this into account, one can see that our M&U pruning criterion still outperforms the ABS criterion consistently.

## 6. Conclusion and Discussion

Motivated by the Wald test statistic, we derived the M&U pruning criterion for neural networks. Our M&U pruning criterion takes both magnitude and uncertainty into account and can balance between them through the hyperparameter  $\lambda$ .

When  $\lambda = 0$  and analytically justified standard error of  $\hat{w}_j$  is used as uncertainty estimate  $\tilde{\sigma}_{\hat{w}_j}$ , the M&U pruning criterion is equal to the Wald test statistic. When  $\lambda \rightarrow \infty$ , the M&U pruning criterion is equal to the absolute value based criterion in Han *et al.* [21]. Thus, our M&U pruning criterion is a generalization of both Wald test statistic and the magnitude based criterion in Han *et al.* [21]. We also suggested a reparametrization of  $\lambda$  that makes finding reasonable values for  $\lambda$  easier.

Further, we introduced ‘pseudo bootstrap’, a very efficient method of measuring uncertainty of weights by tracking their updates during training. The experiments suggest that M&U pruning criterion outperforms the criterion suggested in Han *et al.* [21] most of the time, in terms of test accuracy of the pruned model, with the best result achieved by using  $\tilde{\sigma}_{\hat{w}_j}^{\text{boot}}$ . Thus, using our M&U pruning criterion can lead to models that require less memory and at smaller computational cost, while minimizing the loss of predictive power.

In the future, we wish to investigate numerical and theoretical properties of pseudo bootstrap and compare it to that of ‘usual’ bootstrap. Due to limited amount of GPU resource, we had to keep our models small. We would like to investigate the performance of our method on bigger and deeper neural network models. Further, it will be fruitful to combine our pruning criterion with other types of compression methods such as quantization, Huffman coding, and regularization.

## Acknowledgments

This work is supported by Google Cloud Platform through their research credits program.

## References

- [1] U. Anders and O. Korn. Model selection in neural networks. *Neural networks*, 12(2):309–323, 1999. 1
- [2] S. Anwar, K. Hwang, and W. Sung. Structured pruning of deep convolutional neural networks. *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 13(3):32, 2017. 2
- [3] A. K. Bera and Y. Biliyas. Rao’s score, neyman’s  $c(\alpha)$  and silvey’s  $lm$  tests: an essay on historical developments and some new results. *Journal of Statistical Planning and Inference*, 97(1):9–44, 2001. 3
- [4] P. J. Bickel, D. A. Freedman, et al. Some asymptotic theory for the bootstrap. *The annals of statistics*, 9(6):1196–1217, 1981. 4
- [5] G. Casella and R. L. Berger. *Statistical Inference*. Duxbury Pacific Grove, CA, 2002. 2
- [6] S. Changpinyo, M. Sandler, and A. Zhmoginov. The power of sparsity in convolutional neural networks. *arXiv preprint arXiv:1702.06257*, 2017. 2
- [7] W. Chen, J. Wilson, S. Tyree, K. Weinberger, and Y. Chen. Compressing neural networks with the hashing trick. In *International Conference on Machine Learning (PMLR)*, pages 2285–2294, 2015. 1
- [8] Y. Cheng, D. Wang, P. Zhou, and T. Zhang. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*, 2017. 1
- [9] M. Courbariaux, Y. Bengio, and J.-P. David. Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in neural information processing systems (NeurIPS)*, pages 3123–3131, 2015. 2
- [10] A. C. Davison and D. V. Hinkley. *Bootstrap methods and their application*, volume 1. Cambridge university press, 1997. 4
- [11] M. Denil, B. Shakibi, L. Dinh, N. De Freitas, et al. Predicting parameters in deep learning. In *Advances in neural information processing systems (NeurIPS)*, pages 2148–2156, 2013. 1
- [12] E. L. Denton, W. Zaremba, J. Bruna, Y. LeCun, and R. Fergus. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in neural information processing systems (NeurIPS)*, pages 1269–1277, 2014. 2
- [13] B. Efron. *The Jackknife, the Bootstrap, and Other Resampling Plans*, volume 38. Siam, 1982. 4
- [14] B. Efron. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*, pages 569–593. Springer, 1992. 4
- [15] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. CRC press, 1994. 4
- [16] R. F. Engle. Wald, likelihood ratio, and lagrange multiplier tests in econometrics. *Handbook of econometrics*, 2:775–826, 1984. 3

- [17] Y. Gong, L. Liu, M. Yang, and L. Bourdev. Compressing deep convolutional networks using vector quantization. *arXiv preprint arXiv:1412.6115*, 2014. 2
- [18] I. Goodfellow. Efficient per-example gradient computations. *arXiv preprint arXiv:1510.01799*, 2015. 4
- [19] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT press, 2016. 1
- [20] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *International Conference on Learning Representations (ICLR)*, 2016. 2, 3
- [21] S. Han, J. Pool, J. Tran, and W. Dally. Learning both weights and connections for efficient neural network. In *Advances in neural information processing systems (NeurIPS)*, pages 1135–1143, 2015. 1, 2, 3, 4, 5, 6, 7, 8
- [22] J. W. Hardin. The sandwich estimate of variance. In *Maximum likelihood estimation of misspecified models: Twenty years later*, pages 45–73. Emerald Group Publishing Limited, 2003. 2
- [23] B. Hassibi, D. G. Stork, and G. J. Wolff. Optimal brain surgeon and general network pruning. In *IEEE international conference on neural networks*, pages 293–299, 1993. 1
- [24] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio. Quantized neural networks: Training neural networks with low precision weights and activations. *The Journal of Machine Learning Research (JMLR)*, 18(1):6869–6898, 2017. 2
- [25] K. Hwang and W. Sung. Fixed-point feedforward deep neural network design using weights+ 1, 0, and- 1. In *2014 IEEE Workshop on Signal Processing Systems (SiPS)*, pages 1–6. IEEE, 2014. 2
- [26] M. Jaderberg, A. Vedaldi, and A. Zisserman. Speeding up convolutional neural networks with low rank expansions. *arXiv preprint arXiv:1405.3866*, 2014. 2
- [27] G. Kauermann and R. J. Carroll. The sandwich variance estimator: Efficiency properties and coverage probability of confidence intervals. 2000. 2
- [28] G. Kauermann and R. J. Carroll. A note on the efficiency of sandwich covariance matrix estimation. *Journal of the American Statistical Association*, 96(456):1387–1396, 2001. 2
- [29] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 2, 4
- [30] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 6
- [31] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems (NeurIPS)*, pages 1097–1105, 2012. 7
- [32] L. Le Cam. Maximum likelihood: an introduction. *International Statistical Review*, 58(2):153–171, 1990. 2, 6
- [33] V. Lebedev, Y. Ganin, M. Rakhuba, I. Oseledets, and V. Lempitsky. Speeding-up convolutional neural networks using fine-tuned cp-decomposition. *arXiv preprint arXiv:1412.6553*, 2014. 2
- [34] V. Lebedev and V. Lempitsky. Fast convnets using group-wise brain damage. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2554–2564, 2016. 2
- [35] Y. LeCun. The mnist database of handwritten digits. <http://yann.lecun.com/exdb/mnist/>, 1998. 7
- [36] Y. LeCun, J. S. Denker, and S. A. Solla. Optimal brain damage. In *Advances in neural information processing systems (NeurIPS)*, pages 598–605, 1990. 1
- [37] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf. Pruning filters for efficient convnets. In *International Conference on Learning Representations (ICLR)*, 2017. 2
- [38] Z. Lin, M. Courbariaux, R. Memisevic, and Y. Bengio. Neural networks with few multiplications. In *International Conference on Learning Representations (ICLR)*, 2016. 2
- [39] C. Louizos, M. Welling, and D. P. Kingma. Learning sparse neural networks through  $l_0$  regularization. In *International Conference on Learning Representations (ICLR)*, 2018. 1
- [40] S. Narang, E. Elsen, G. Diamos, and S. Sengupta. Exploring sparsity in recurrent neural networks. In *International Conference on Learning Representations (ICLR)*, 2017. 2, 3
- [41] J. Neyman and E. S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231(694-706):289–337, 1933. 3
- [42] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017. 5
- [43] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision (ECCV)*, pages 525–542. Springer, 2016. 2
- [44] A. See, M.-T. Luong, and C. D. Manning. Compression of neural machine translation models via pruning. In *The SIGNLL Conference on Computational Natural Language Learning (CoNLL)*, 2016. 2, 3
- [45] J. Shao and D. Tu. *The jackknife and bootstrap*. Springer Science & Business Media, 2012. 4
- [46] S. D. Silvey. The lagrangian multiplier test. *The Annals of Mathematical Statistics*, 30(2):389–407, 1959. 1, 3
- [47] S. Srinivas and R. V. Babu. Data-free parameter pruning for deep neural networks. In *British Machine Vision Conference (BMVC)*, 2015. 1
- [48] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research (JMLR)*, 15(1):1929–1958, 2014. 2
- [49] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012. 2, 4
- [50] P. Toulis, E. M. Airoldi, et al. Asymptotic and finite-sample properties of estimators based on stochastic gradients. *The Annals of Statistics*, 45(4):1694–1727, 2017. 2, 3, 4, 6
- [51] K. Ullrich, E. Meeds, and M. Welling. Soft weight-sharing for neural network compression. In *International Conference on Learning Representations (ICLR)*, 2017. 1

- [52] A. W. Van der Vaart. *Asymptotic Statistics*. Cambridge University Press, 2000. [2](#)
- [53] V. Vanhoucke, A. Senior, and M. Z. Mao. Improving the speed of neural networks on cpus. 2011. [2](#)
- [54] L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus. Regularization of neural networks using dropconnect. In *International Conference on Machine Learning (ICML)*, pages 1058–1066, 2013. [2](#)
- [55] H. White. Maximum likelihood estimation of misspecified models. *Econometrica: Journal of the Econometric Society*, pages 1–25, 1982. [2](#), [3](#)
- [56] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. [5](#)