Generative Archimedean Copulas: Supplementary Material

Yuting Ng[1] **Ali Hasan**[2] **Khalil Elkhalil**[1] **Vahid Tarokh**[1]

[1]Department of Electrical and Computer Engineering, Duke University
[2]Department of Biomedical Engineering, Duke University

1 PROBABILISTIC CONSTRUCTION OF ARCHIMEDEAN AND HIERARCHICAL ARCHIMEDEAN COPULAS

Copulas can be derived from cumulative distribution functions (CDFs) via Sklar's theorem, i.e. specify a joint CDF F, compute univariate CDFs $F_1, \cdots, F_d$ from the joint CDF, then obtain the copula as $C(\mathbf{u}) = F(F_1^{-1}(u_1), \cdots, F_d^{-1}(u_d)), \mathbf{u} \in [0,1]^d$. Sklar's theorem also applies to survival functions, i.e. for joint survival function $\bar{F}(\mathbf{x}) = P(X_1 > x_1, \cdots, X_d > x_d), \mathbf{x} \in \mathbb{R}^d$, with univariate survival functions $\bar{F}_1, \cdots, \bar{F}_d$ where $\bar{F}_j = P(X_j > x_j)$, the copula which couples $\bar{F}$ to $\bar{F}_1, \cdots, \bar{F}_d$ is called the *survival copula* and is given as the copula C for which $\bar{F}(x) = C(\bar{F}_1(x_1), \cdots, \bar{F}_d(x_d))$.

1.1 ARCHIMEDEAN COPULAS

We restate the probabilistic construction found in [Joe, 2014] Chapter 3.2, following [Marshall and Olkin, 1988]:

Let $G_1, \cdots, G_d$ be univariate CDFs. Let $Q \sim F_Q$ be a positive random variable with Laplace transform φ_Q, let $X_1, \cdots, X_d$ be dependent random variables that are conditionally independent given $Q = q$ such that $[X_j | Q = q] \sim G_j^q,\ q > 0$.

The joint CDF is:

$$F(x_1, \cdots, x_d) = \int_0^\infty G_1^q(x_1) \cdots G_d^q(x_d) dF_Q(q) = \varphi_Q(-\log G_1(x_1) - \cdots - \log G_d(x_d)), \tag{1}$$

with univariate CDFs obtained from the joint CDF as:

$$F_j(x_j) = \int_0^\infty G_j^q(x_j) dF_Q(q) = \varphi_Q(-\log G_j(x_j)),\ j \in \{1, \cdots, d\}, \tag{2}$$

and inverse:

$$F_j^{-1}(u_j) = G_j^{-1}(\exp\{-\varphi_Q^{-1}(u_j)\}),\ u_j \in (0,1),\ j \in \{1, \cdots, d\}, \tag{3}$$

such that the copula via Sklar's theorem is:

$$C(\mathbf{u}) = F(F_1^{-1}(u_1), \cdots, F_d^{-1}(u_d)) = \varphi_Q(\varphi_Q^{-1}(u_1) + \cdots + \varphi_Q^{-1}(u_d)),\ \mathbf{u} \in [0,1]^d. \tag{4}$$

The multivariate extension of bivariate Archimedean copulas was introduced in [Kimberling, 1974] with the condition that the above expression is a valid copula for any d whenever φ, known as the *generator* of the Archimedean copula, is *completely monotone*, i.e. the Laplace transform of a positive random variable [Bernstein, 1929, Widder, 1941]. The mixture representation with Laplace transform generators and an efficient algorithm for sampling from the mixture representation was subsequently given in [Marshall and Olkin, 1988].

We restate the sampling algorithm found in [McNeil, 2008], following [Marshall and Olkin, 1988]:

Consider $(U_1, \cdots, U_d) = (\varphi(E_1/M), \cdots, \varphi(E_d/M))$, where $(E_1, \cdots, E_d) \sim i.i.d.$ Exp(1) are independent and identically distributed unit exponentials and $M \sim F_M$ is a positive random variable with Laplace transform φ_M.

$$\begin{aligned} P(U_1 \leq u_1, \cdots, U_d \leq u_d) &= \int_0^\infty P(U_1 \leq u_1, \cdots, U_d \leq u_d | M = s) dF_M(s) \\ &= \int_0^\infty e^{-s(\varphi^{-1}(u_1) + \cdots + \varphi^{-1}(u_d)} dF_M(s) \\ &= \varphi_M(\varphi_M^{-1}(u_1) + \cdots + \varphi_M^{-1}(u_d)). \end{aligned} \tag{5}$$

Thus an algorithm for sampling $\mathbf{U} \sim C$ is to sample $M \sim F_M$ with Laplace transform φ_M, sample $(E_1, \cdots, E_d) \sim i.i.d.$ Exp(1), then compute $(U_1, \cdots, U_d) = (\varphi_M(E_1/M), \cdots, \varphi_M(E_d/M))$.

1.2 HIERARCHICAL ARCHIMEDEAN COPULAS

A simple nested mixture representation involving Laplace transform generators was introduced in [Joe, 1997]. Conditions for the nested copula to be a valid copula, called *sufficient nesting conditions*, was derived based on the composition of an *outer generator* φ_0 and an *inner generator* φ_1 to get a completely monotone Laplace transform *nested generator* $e^{-\nu_0 \varphi_0^{-1} \circ \varphi_1}$, where ν_0 is a positive random variable with Laplace transform φ_0, such that φ_0, φ_1 and $(\varphi_0^{-1} \circ \varphi_1)'$ are completely monotone.

We illustrate with a simple three-dimensional example found in [McNeil, 2008], following [Joe, 1997]. Consider the hierarchical Archimedean copula:

$$C(u_1, u_2, u_3) = \varphi_0(\varphi_0^{-1}(u_1) + \varphi_0^{-1} \circ \varphi_1(\varphi_1^{-1}(u_2) + \varphi_1^{-1}(u_3))), \tag{6}$$

where φ_0, φ_1 are Laplace transform generators of Archimedean copulas. We would like to express the above as a mixture of conditionally independent CDFs. Let G_0 be a distribution with Laplace transform φ_0:

$$\begin{aligned} C(u_1, u_2, u_3) &= \varphi_0(\varphi_0^{-1}(u_1) + \varphi_0^{-1} \circ \varphi_1(\varphi_1^{-1}(u_2) + \varphi_1^{-1}(u_3))) \\ &= \int_0^\infty e^{-\nu_0 \varphi_0^{-1}(u_1)} e^{-\nu_0 \varphi_0^{-1} \circ \varphi_1(\varphi_1^{-1}(u_2) + \varphi_1^{-1}(u_3))} dG_0(\nu_0) \\ &= \int_0^\infty F_0^{\nu_0}(u_1) C_{01}(F_0^{\nu_0}(u_2), F_0^{\nu_0}(u_3); \nu_0) dG_0(\nu_0), \end{aligned} \tag{7}$$

where $F_0(\cdot) := e^{-\varphi_0^{-1}(\cdot)}$ and $F_0^{\nu_0}$ is a valid CDF for any $\nu_0 > 0$. In addition, $C_{01}(\cdot; \nu)$ is an Archimedean copula with Laplace transform generator $\varphi_{01}(\cdot; \nu_0) = e^{-\nu_0 \varphi_0^{-1} \circ \varphi_1(\cdot)}$ and generator inverse $\varphi_{01}^{-1}(\cdot; \nu_0) = \varphi_1^{-1} \circ \varphi_0(-\log(\cdot)/\nu_0)$, such that $C_{01}(\cdot; \nu_0)$ taking marginals $F_0^{\nu_0}(u_2)$ and $F_0^{\nu_0}(u_3)$ as inputs gives:

$$\begin{aligned} C_{01}(F_0^{\nu_0}(u_2), F_0^{\nu_0}(u_3); \nu_0) &= \varphi_{01}(\varphi_{01}^{-1}(F_0^{\nu_0}(u_2); \nu_0) + \varphi_{01}^{-1}(F_0^{\nu_0}(u_3); \nu_0); \nu_0) \\ &= e^{-\nu_0 \varphi_0^{-1} \circ \varphi_1(\varphi_1^{-1} \circ \varphi_0(-\log(e^{-\nu_0 \varphi_0^{-1}(u_2)})/\nu_0) + \varphi_1^{-1} \circ \varphi_0(-\log(e^{-\nu_0 \varphi_0^{-1}(u_3)})/\nu_0))} \\ &= e^{-\nu_0 \varphi_0^{-1} \circ \varphi_1(\varphi_1^{-1}(u_2) + \varphi_1^{-1}(u_3))}. \end{aligned} \tag{8}$$

The completely monotone property of the Laplace transform generator $\varphi_{01}(\cdot; \nu) = e^{-\nu \varphi_0^{-1} \circ \varphi_1(\cdot)}$ then implies $(\varphi_0^{-1} \circ \varphi_1)'$ is completely monotone. In addition, letting $G_{01}(\cdot; \nu_0)$ be a distribution with Laplace transform $\varphi_{01}(\cdot; \nu_0)$, we express the hierarchical Archimedean copula as a nested mixture of conditionally independent CDFs:

$$\begin{aligned} C(u_1, u_2, u_3) &= \int_0^\infty F_0^{\nu_0}(u_1) C_{01}(F_0^{\nu_0}(u_2), F_0^{\nu_0}(u_3); \nu_0) dG_0(\nu_0) \\ &= \int_0^\infty F_0^{\nu_0}(u_1) \int_0^\infty e^{-\nu_{01} \varphi_{01}^{-1}(F_0^{\nu_0}(u_2))} e^{-\nu_{01} \varphi_{01}^{-1}(F_0^{\nu_0}(u_3))} dG_{01}(\nu_{01}; \nu_0) dG_0(\nu_0) \\ &= \int_0^\infty F_0^{\nu_0}(u_1) \int_0^\infty e^{-\nu_{01} \varphi_1^{-1}(u_2)} e^{-\nu_{01} \varphi_1^{-1}(u_3)} dG_{01}(\nu_{01}; \nu_0) dG_0(\nu_0) \\ &= \int_0^\infty \int_0^\infty F_0^{\nu_0}(u_1) F_1^{\nu_{01}}(u_2) F_1^{\nu_{01}}(u_3) dG_{01}(\nu_{01}; \nu_0) dG_0(\nu_0), \end{aligned} \tag{9}$$

where $F_1(\cdot) := e^{-\varphi_1^{-1}(\cdot)}$ and $F_1^{\nu_{01}}$ is a valid CDF for any $\nu_{01} > 0$.

This construction and condition were restated for nesting to arbitrary depth in [McNeil, 2008].

Based on the mixture representation, McNeil [2008] also provided algorithms for sampling nested Clayton and nested Gumbel copulas. It was also showed that Clayton and Gumbel copulas are unfortunately not compatible for nesting. The challenge was to find combinations of known distributions with φ_0, φ_1 and $e^{-\nu_0 \varphi_0^{-1} \circ \varphi_1}$ as their Laplace transforms. Sampling using McNeil [2008]'s algorithm for nested Ali-Mikhail-Haq, nested Frank, nested Joe, more parametric families and numerical inversion of Laplace transform was by [Hofert, 2008]. It was subsequently recognized in [Hering et al., 2010] that the sufficient nesting condition for $(\varphi_0^{-1} \circ \varphi_1)'$ to be completely monotone can be satisfied by letting $\varphi_1 = \varphi_0 \circ \psi_1$, where ψ_1 is the Laplace exponent, with completely monotone derivative, in the Laplace transform of Lévy subordinators.

We restate the probabilistic construction with Lévy subordinators from [Hering et al., 2010]:

For each 'time' $t \geq 0$, the Laplace transform of a Lévy subordinator Λ_t, i.e. a non-decreasing Lévy processes such as the compound Poisson process, is given as:

$$\mathbf{E}[e^{-x\Lambda_t}] = e^{-t\psi(x)}, \tag{10}$$

where $\psi(x)$ is the Laplace exponent.

Consider $(\frac{E_{1,1}}{\Lambda_1(M)}, \cdots, \frac{E_{1,d_1}}{\Lambda_1(M)}, \cdots, \cdots, \cdots, \frac{E_{J,1}}{\Lambda_J(M)}, \cdots, \frac{E_{J,d_J}}{\Lambda_J(M)})$, where $E_{j,i} \sim i.i.d.$ $\text{Exp}(1)$, Λ_j are Lévy subordinators with Laplace exponents ψ_j, and Λ_j are evaluated at a common 'time' $t = M$, where M is a positive random variable with Laplace transform φ_0. The hierarchical Archimedean copula is then constructed using the survival analog of Sklar's theorem.

The joint survival function is:

$$\begin{aligned}
P(\frac{E_{j,i}}{\Lambda_j(M)} > x_{j,i} \text{ for all} j, i) &= \mathbf{E}[\prod_{j=1}^{J}\prod_{i=1}^{d_J} e^{-\Lambda_j(M)x_{j,i}}] \\
&= \mathbf{E}[\prod_{j=1}^{J} e^{-\Lambda_j(M)\sum_{i=1}^{d_J} x_{j,i}}] \\
&= \mathbf{E}[\prod_{j=1}^{J} e^{-M\psi_j(\sum_{i=1}^{d_j} x_{j,i})}] \\
&= \mathbf{E}[e^{-M\sum_{j=1}^{J}\psi_j(\sum_{i=1}^{d_j} x_{j,i})}] \\
&= \varphi_0(\sum_{j=1}^{J} \varphi_0^{-1} \circ (\varphi_0 \circ \psi_j)(\sum_{i=1}^{d_j} x_{j,i})),
\end{aligned} \tag{11}$$

and each component $\frac{E_{j,i}}{\Lambda_j(M)}$ has survival function:

$$P(\frac{E_{j,i}}{\Lambda_j(M)} > x) = \mathbf{E}[e^{-x\Lambda_j(M)}] = \mathbf{E}[e^{-M\psi_j(x)}] = (\varphi_0 \circ \psi_j)(x). \tag{12}$$

Using the survival analog of Sklar's theorem, given the above univariate survival functions, the hierarchical Archimedean copula C with outer generator φ_0 and inner generators $\varphi_j = \varphi_0 \circ \psi_j$, we recover the above joint survival function.

2 EXPERIMENT DETAILS

Following the experiment setup in [Ling et al., 2020], the commonly-used copulas were the Clayton, Frank and Joe copulas, each governed by a single parameter and chosen to be 5, 15, and 3 respectively. Each dataset had 2000 train and 1000 test points. The real-world data were the Boston housing, Intel-Microsoft (INTC-MSFT) stocks and Google-Facebook (GOOG-FB) stocks. Each dataset was divided into train and test points in a 3:1 ratio, then rank-normalized to get approximately uniform margins.

Similar to the experiment parameters in [Ling et al., 2020], the tolerance for Newton's root-finding method was 1e-10. The generative neural network was a multilayer perceptron of comparable size, 2 hidden layers, each of width 10. We used

U(0, 1) as the input source of randomness, default weight initialization, LeakyReLU intermediate activations and $\exp(.)$ output activation. For training with maximum likelihood, we used the same optimization parameters: stochastic gradient descent (SGD) with learning rate 1e-5 and momentum 0.9 on sum of log-likelihoods. For training with goodness-of-fit, we used SGD with learning rate 1e-3 and momentum 0.9. For adversarial training, we used Adam [Kingma and Ba, 2015] with learning rate 1e-4, momentum 0.9 and betas (0.5, 0.999). The discriminative neural network had a single hidden layer of width 20, default weight initialization, LeakyReLU intermediate activations and sigmoid(.) output activation. All training methods used the same batch size of 200 and converged within 10k epochs. We reported the results at 10k epoch. Experiments were conducted using PyTorch, on a 2.7 GHz Intel Core i7 with 16 GB of RAM.

To reduce computation complexity during training, we used a smaller number $L = 100$ samples from the generative neural network to approximate the Laplace transforms. To increase inference accuracy for evaluation, we used a larger number $L = 1000$ samples from the generative neural network to approximate the Laplace transforms.

2.1 ENFORCING STRUCTURAL PROPERTIES

Compared to vanilla GAN [Goodfellow et al., 2014], our generating network must satisfy an Archimedean copula. We show this via the training progression for learning a Clayton copula in Figure 1.

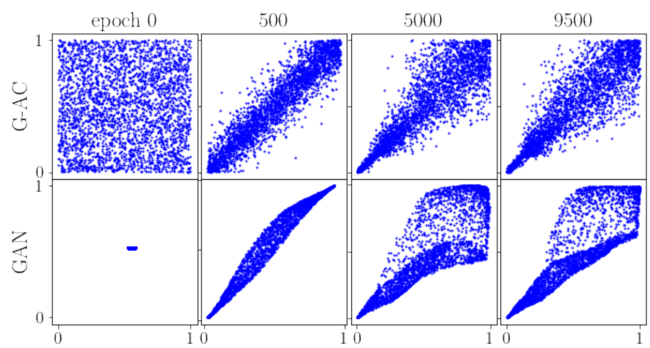

Figure 1: Training progression at epochs 0, 500, 5000 and 10000, for learning a Clayton copula. Samples from our copula, shown on top, must satisfy an Archimedean copula, while that from a vanilla GAN, shown below, may not.

References

Serge Bernstein. Sur les fonctions absolument monotones. *Acta Mathematica*, 52:1 – 66, 1929. doi: 10.1007/BF02592679. URL https://doi.org/10.1007/BF02592679.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

Christian Hering, Marius Hofert, Jan-Frederik Mai, and Matthias Scherer. Constructing hierarchical archimedean copulas with lévy subordinators. *Journal of Multivariate Analysis*, 101(6):1428–1433, 2010. ISSN 0047-259X. doi: https://doi.org/10.1016/j.jmva.2009.10.005. URL https://www.sciencedirect.com/science/article/pii/S0047259X09001961.

Marius Hofert. Sampling archimedean copulas. *Computational Statistics & Data Analysis*, 52(12):5163–5174, 2008. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2008.05.019. URL https://www.sciencedirect.com/science/article/pii/S0167947308002910.

Harry Joe. *Multivariate models and dependence concepts*. Chapman and Hall/CRC, May 1997. ISBN 9780367803896. doi: 10.1201/9780367803896. URL https://www.taylorfrancis.com/books/9781466581432.

Harry Joe. *Dependence modeling with copulas*. Chapman and Hall/CRC, June 2014. ISBN 9781466583238. doi: 10.1201/b17116. URL https://www.taylorfrancis.com/books/9781466583238.

Clark H. Kimberling. A probabilistic interpretation of complete monotonicity. *aequationes mathematicae*, 10(2):152–164, 1974. doi: 10.1007/BF01832852. URL https://doi.org/10.1007/BF01832852.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Chun Kai Ling, Fei Fang, and J. Zico Kolter. Deep archimedean copulas. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1535–1545. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/10eb6500bd1e4a3704818012a1593cc3-Paper.pdf.

Albert W. Marshall and Ingram Olkin. Families of multivariate distributions. *Journal of the American Statistical Association*, 83(403):834–841, 1988. doi: 10.1080/01621459.1988.10478671. URL https://www.tandfonline.com/doi/abs/10.1080/01621459.1988.10478671.

Alexander J. McNeil. Sampling nested archimedean copulas. *Journal of Statistical Computation and Simulation*, 78(6): 567–581, 2008. doi: 10.1080/00949650701255834. URL https://doi.org/10.1080/00949650701255834.

David Vernon Widder. *Laplace Transform (PMS-6)*. Princeton University Press, 1941. doi: doi:10.1515/9781400876457. URL https://doi.org/10.1515/9781400876457.