An interpretable alternative for sensivity analysis : Mutual Information Copulas estimator

Adrien Cortes
Mines Saint-Etienne, ISFA Lyon
adrien.cortes@etu.emse.fr

September 1, 2023

Abstract

Sensitive analysis is an important tool for understanding the behavior of complex systems. In this article, we propose a novel framework for simultaneous learning of copula simulation and density estimation in a non-parametric way, enabling both global and local sensitive analysis with a single calibration of the model. Our approach is based on the link between copulas and mutual information. We compare our approach to state-of-the-art methods, such as Sobol indices, and show its effectiveness in a range of simulations. We also introduce a new sensitivity index, the local mutual information coefficient, which we use for normalization instead of traditional methods. Our results demonstrate the advantages of our framework for sensitivity analysis and highlight the potential of copula-based approaches for understanding complex systems.

1 Introduction

Sensitivity analysis is a useful technique in understanding the contribution of input uncertainties on output uncertainty [1]. Local sensitivity analysis involves examining the impact of input variations in the vicinity of a specific point, while global sensitivity analysis considers variations across the entire feasible space of inputs [2, 3]. Two broad categories of sensitivity indices have been created: variance-based, such as Sobol Indices [4], and those based on information theory [5], which are used in machine learning to identify the most significant features [6, 7]. Mutual information is an example of the latter approach [8], and a new sensitivity metric based on the relationship between copulas and mutual information has been proposed [8].

Recently, copulas have regained interest in machine learning due to their ability to model dependence in high-dimensional spaces [9]. Some works have proposed copula autoencoders for use as generative models [10, 11].

We were particularly drawn to two recent papers, [12] and [13]. [12] proposed the first universal, non-parametric model for estimating high-dimensional copula distributions with guaranteed uniformity of the marginal distributions. In addition, [13] proposed a method for learning copula density from observations and used Markov Chain Monte Carlo algorithm to simulate data. Although generative models such as autoencoders [14, 15] can learn how to simulate data, explicitly learning copula density is not straightforward.

Therefore, in this paper, we propose to combine the Implicit Generative Copula (IGC) model [12] and the copula density neural estimator (CODINE) [13] to create the Mutual Information Copulas Estimator (MICE). Our proposed model is capable of learning copula simulation and density estimation and defines a local mutual information index that connects global and local sensitivity analyses within a single framework. We will also compare our proposed model to the state-of-the-art Mutual Information Neural Estimator (MINE) [6] and demonstrate why our model provides a safer framework.

Our Contributions

1. We introduce a framework that allows for simultaneous learning of copula simulation and density estimation in a non-parametric manner, which is the first of its kind.

2. We use copulas to perform both global and local sensitivity analysis with only one model calibration, and we define a local mutual information index.
3. We compare our method to state-of-the-art standards for sensitivity analysis, such as Sobol indices.

2 Related Work

Copulas Copula come from the latin word *link*, coupling marginal distributions into a joint distribution. Sklars Theorem [16] insure that if the random vector $X = (X_1, \ldots, X_d)$ has joint distribution F and marginal distribution $F_1, \ldots, F_d$ if and only if there exist a unique copula C which is the joint distribution of $U = (U_1, \ldots, U_d) = (F_1(X_1), \ldots, F_d(X_d))$. We say that U is following the so called *copula distribution* P^C., distribution in the unit space with uniform marginales.

Mutual information The concept of entropy of a random variable X was first introduced by [17], and is defined as $H(X) = -\int_{\mathbf{X}} \log f(x) f(x) dx$. The mutual information between two random variables X and Y is defined as the decrease in entropy of Y when given information about X:

$$MI(X,Y) = \int_{\mathbf{X}} \int_{\mathbf{Y}} \log \frac{f(x,y)}{f_X(x) f_Y(y)} f(x,y) dy dx$$

Mutual information is positive, symmetric, and equal to zero if the two variables are independent. A remarkable relation is $MI(X,Y) = H(Y) - H(Y \mid X)$.

In addition, [6] proposed the Mutual Information Neural Estimator (MINE) based on the surprising relation $I(X,Z) = D_{KL}(\mathbb{P}_{XY} \| \mathbb{P}_X \otimes \mathbb{P}_Y)$ and $D_{KL}(\mathbb{P} \| \mathbb{Q}) = \sup_{T:\Omega \to \mathbb{R}} \mathbb{E}_{\mathbb{P}}[T] - \log\left(\mathbb{E}_{\mathbb{Q}}\left[e^T\right]\right)$. MINE approximates the function T that maximizes this quantity.

Recently, [8] proposed a new sensitivity index called the mutual information coefficient, which we will be using in our study. This coefficient is defined as

$$\rho_{MI}(X_1, X_2) = \sqrt{1 - \exp\left[-2MI(X_1, X_2)\right]}$$

instead of normalizing the mutual information by the entropy of Y as in [18]. ρ_{MI} is a *proper measure of dependence*. We have $\rho_{MI} = 0$ for independant variables, and $\rho_{MI} = 1$ in case of perfect dependance. And ρ_{MI} and the linear coefficient of correlation coincide in the case of joint Gaussian distribution.

Link between copulas and mutual information As surprising as it may be, there is a strong link between copulas and mutual information [19]. Some works, contrary to ours, use parametric copulas to estimate mutual information [20, 21].

The link between copulas and mutual information can be expressed as follows:

$$MI(X,Y) = E_{(U,V) \sim c(u,v)}[\log c(U,V)] \tag{1}$$

where $X = F_1^{-1}(U)$ and $Y = F_2^{-1}(V)$. We propose to learn how to simulate the tuple (U,V) and to learn the density $c(u,v)$ in order to directly learn the mutual information.

3 Our model

MICE Our proposal is to establish a connection between IGC and CODINE. In contrast to the approach taken in the original paper by Letizia et al. (2022) [13], which involves training directly on data observations, we aim to train our model using sample extracts from IGC. Our objective is to implicitly capture the underlying distribution using an infinite dataset, which can be particularly advantageous when working with limited data. Therefore, we learn the implicit density learned by IGC if we have a sufficiently complex neural network for CODINE. The interpretation is then straightforward, as we can analyze the behavior of IGC to explain the variation of the calculation of the coefficient. Furthermore, this approach is advantageous as IGC can simulate data, making the analysis even simpler. As demonstrated in equation 1, it is essential to learn how to simulate variables and their distribution in order to compute mutual information indices. Letizia et al. (2022) demonstrate that, if $\pi_U(\mathbf{u})$ represents a multivariate uniform distribution on the unit cube $[0,1]^d$, and

$$\mathcal{J}_f(T) = \mathbb{E}_{\mathbf{u} \sim c_U(\mathbf{u})}[T(\mathbf{u})] - \mathbb{E}_{\mathbf{u} \sim \pi_U(\mathbf{u})}\left[exp((T(\mathbf{u})) - 1))\right] \tag{2}$$

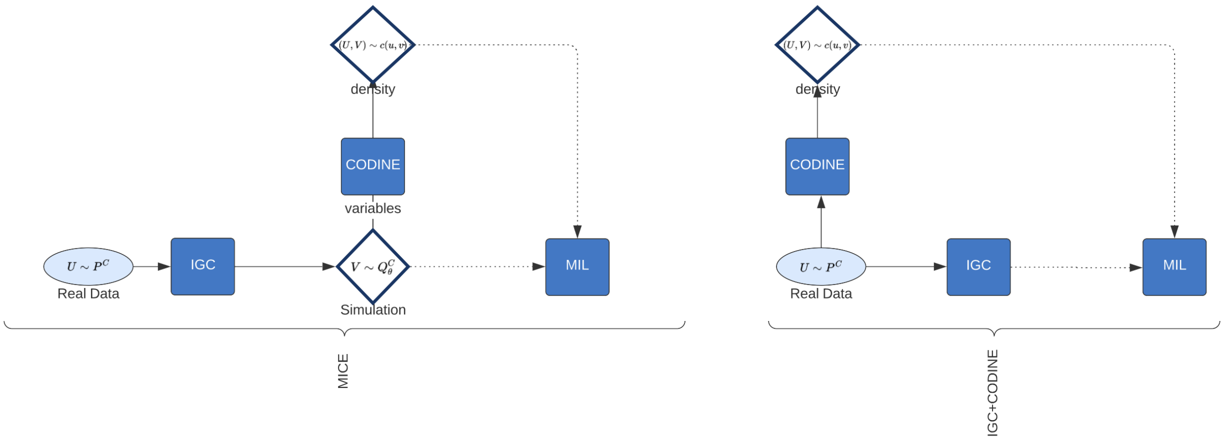

Figure 1: We conducted a comparison between two models - the first model, MICE, trains CODINE directly on the simulation data of IGC, while the second model trains IGC and CODINE separately. Our results in the next section show that MICE can reduce the variance of CODINE's learning, improve its performance slightly, and is much more suitable for interpretability of the framework as a whole.

then

$$c_U(\mathbf{u}) = exp((\hat{T}(\mathbf{u})) - 1)$$

where

$$\hat{T}(\mathbf{u}) = \arg\max_T \mathcal{J}_f(T)$$

To maximize the objective function in equation 2, we opt to approximate the function T using a neural network. We train on samples drawn from the distribution Q^C_θ, instead of directly training on samples drawn from the copula distribution $c_U(\mathbf{u})$. The IGC procedure involves two steps: first, a random noise vector $\mathbf{Z} \in \mathbb{R}^K$ is drawn from a latent distribution and mapped to a latent variable $\mathbf{Y} \in \mathbb{R}^D$ using a neural network $\mathbf{g}_\theta : \mathbb{R}^K \to \mathbb{R}^D$. Then, $\mathbf{Y}$ is drawn from the distribution Q_θ.

You considere the transformation for $d = 1, ...D$:

$$v_d = F_{Y_d}(y_d)$$

then $\mathbf{V} \sim Q^C_\theta$, and you can use it to minimise 2. To be able to train IGC model, you need to minimise an energy distance :

$$ED^2\left(P^C, Q^C_\theta\right) = 2\mathbb{E}\|\mathbf{U} - \mathbf{V}\| - \mathbb{E}\left\|\mathbf{U} - \mathbf{U}'\right\| - \mathbb{E}\left\|\mathbf{V} - \mathbf{V}'\right\|$$

where, $\mathbf{U}, \mathbf{U}'$ and $\mathbf{V}, \mathbf{V}'$ are independent copies of a random vector with distribution P^C and Q^C_θ respectively. It holds that $ED^2\left(P^C, Q^C_\theta\right) = 0$ if and only if $P^C = Q^C_\theta$. There are some difficulties in the training explaining directly in the original paper[1].

Validation and interpretation Our model, as depicted in Figure 1, is straightforward to validate. To determine whether the model has captured all the dependencies, we simply compare the simulated copulas with the empirical copulas. Moreover, the realizations and copula densities are useful for evaluating the level of convergence achieved by the model. Our model, which is based on the copula structure, has a significant advantage over the Sobol indices approach in terms of its ability to capture global dependencies. Sobol indices provide sensitivity measures, indicating the contribution of each input variable to the output variability. However, this approach does not provide information about the global dependence structure, which is essential for accurately capturing the overall behavior of the system. In contrast, the copula structure provides a more global view of the dependencies between the input variables, which is a strength of our model. By leveraging this information, our model can better capture the underlying relationships between the input variables and the output.

[1]It reside in a differentiable approximation of cumulative distribution functions

4 Local Mutual information index

After approximating the copula and its density from the data, it becomes possible to define a new sensitivity index called the local mutual information index. In the litterature you can find [22] a similar name for an other coefficient. But to the best of our knowledge, this is a new definition we propose here actually. This index measures the mutual information between two variables over a given range of values, defined by the parameters α and β. Specifically, the local mutual information index is defined as:

$$MIL_{[\alpha,\beta]}(X,Y) = \int_{\alpha}^{\beta} \int_{0}^{1} c(u,v) \log(c(u,v)) du dv \tag{3}$$

where α and β are quantiles of X.

It is worth noting that as α and β approach 0 and 1, respectively, the local mutual information index converges to the mutual information between the two variables :

$$\lim_{\alpha \to 0} \lim_{\beta \to 1} MIL_{[\alpha,\beta]}(X,Y) = IM(X,Y)$$

The advantage of this sensitivity estimator is that it can be easily computed for every α and β once the model is trained, without requiring additional computational resources. However, we will see in applications that it is necessary to have a minimum number of values to having a significative prediction. So we preconise $\alpha, \beta \in [0.2, 0.8], \quad \alpha < \beta$.

This feature provides a natural link between global and local sensitivity analysis using only one model, which is highly desirable. Furthermore, compared to the MINE approach, where the copula is not learned, our model is much more interpretable. By learning the copula, we can better understand the underlying relationships between the input variables and their impact on the output.

5 Applications

In this paragraph, we are considering $(\Omega, \mathcal{F}, \mathbb{P})$ as a probabilized space. We consider a function:

$$f : \mathbb{R}^n \to \mathbb{R} \tag{4}$$

Here, f is a mapping that takes n input variables in $\mathbb{R}^n$ and returns a real output. f can represent various objects such as a physical system, a black box, or any other imaginable function. We do not make any assumptions about f. In our sensitive analysis problem, we observe inputs $(X_1, \ldots, X_n)$ and the corresponding output $f(X_1, \ldots, X_n)$. In general, we cannot choose the inputs we observe, and we model them as random variables. There exist various real-world problems that involve sensitive analysis, such as analyzing water quality [23], nuclear reactors [24], or chemical models [25].

Classical methods typically involve calculating mutual information or Sobol indices (variance-based indices) of the different inputs. However, in this paragraph, we will explore an alternative approach by using MICE to interpret sensitivity calculations. We will demonstrate how MICE can be employed to challenge and enhance classical sensitivity analysis methods.

A classical case of application in sensitivity analysis: Learning Bivariate Copulas In this example case, we will conduct a sensitivity analysis of X_1 on $f(X_1, \ldots, X_n)$. For the sake of simplicity, let us assume that we observe the couple $(X_1, f(X_1, \ldots, X_n))$. Furthermore, suppose that the joint distribution of this couple is characterized by a known copula. Specifically, we assume that $(X_1, f(X_1, \ldots, X_n))$ is successively described by several copulas, and our objective is to learn these copulas [26] using MICE. By doing so, we can better understand the limitations of MICE in modeling complex copulas and use this insight to refine traditional sensitivity analysis methods. We consider :

1. **Clayton** $C_\theta(u,v) = (u^{-\theta} + v^{-\theta} - 1)^{-1/\theta}$ where $\theta \in [1, \infty)$
2. **Gumbel** $C(u,v) = e^{-\left((-\ln u)^{\theta} + (-\ln v)^{\theta}\right)^{\frac{1}{\theta}}}$ where $\theta \in [1, \infty)$
3. **Independent**, $C(u,v) = uv$

Our results are presented in 2 and 3.

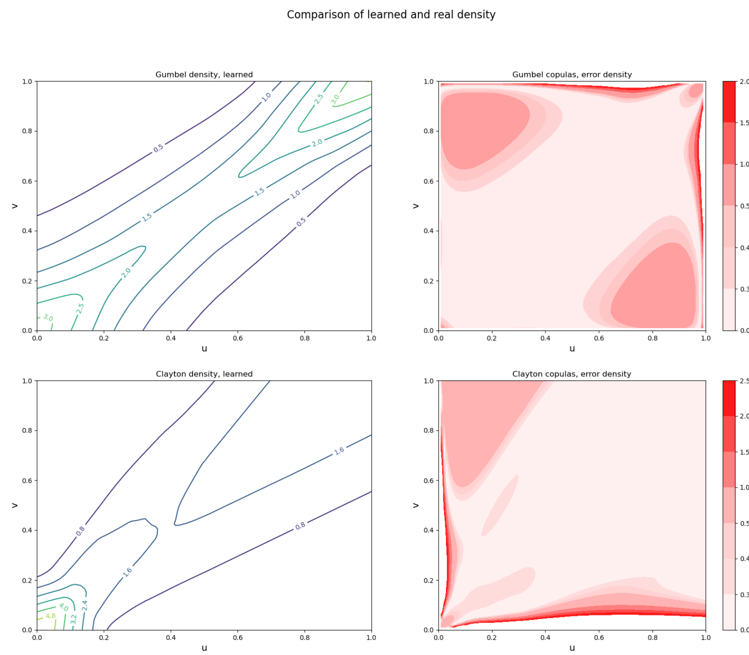

Figure 2: In this illustration, we have plotted the density learned by MICE on 1000 observations of the couple $(X_1, f(X_1, \ldots, X_n))$ for the Gumbel and Clayton copulas with $\theta = 2$. Additionally, we have included a representation of the error between the learned density and the theoretical density. We observe that MICE is able to accurately approximate the density in regions where it is not too high. However, due to the architecture of our neural network, we encounter difficulties in estimating high-density regions. This is an important consideration to keep in mind, as it can directly impact the mutual information and lead to underestimation. For copulas such as the independent or Gaussian copulas, which have fewer high-density regions, MICE provides a very good approximation.

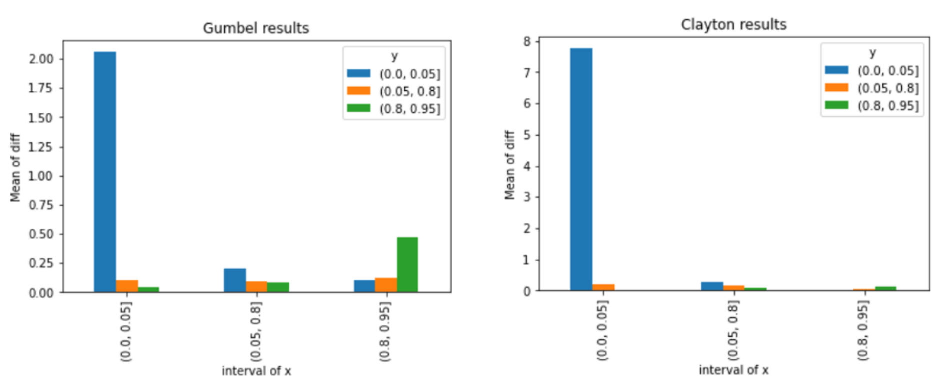

Figure 3: We have also plotted the mean approximation error for different quantile values of X_1 and $Y = f(X_1, \ldots, X_n)$. As expected, we observe that the error significantly increases when the density is high, such as for very small or large quantile values. This is due to the difficulties in accurately estimating high-density regions with MICE, as noted previously. Therefore, when performing sensitivity analysis with MICE, it is important to take into account the potential underestimation of mutual information in high-density regions.

Neural networks may struggle to approximate the area where the density of bivariate random variables is very high due to a phenomenon called "vanishing gradients." When the probability density function (PDF) of a bivariate random variable has very sharp peaks or high values, the gradients of the loss function with respect to the network parameters become very small in those regions. As a result, the network may have difficulty updating its parameters to fit those regions properly, leading to poor performance.

To address these challenges, various techniques have been developed to improve the performance of neural networks for approximating the density of bivariate random variables with high-density regions, such as adding more layers to the network, using different activation functions, and incorporating regularization techniques.

We also want to emphasize the advantages of training CODINE on samples extracted from IGC, rather than directly on the data. Firstly, when dealing with low data situations, learning the distribution implicitly learned by IGC reduces the problem to focusing only on IGC. Furthermore, as shown in the table below, training CODINE on IGC outputs improves its precision. We calculate the mean absolute error between the predicted density $\hat{c}$ by MINE and the real Gumbel density c of parameter $\theta = 2$ for two different cases. In case A, we calculate the error on all the predicted density values $E_{U,V\sim u(0,1)}[|c(U,V) - \hat{c}(U,V)|]$. In case B, we calculate the error $E_{U,V\sim u(0.2,0.8)}[|c(U,V) - \hat{c}(U,V)|]$:

Number of observations	MICE A	CODINE A	MICE B	CODINE B
50.0	**0.308342**	0.381140	**0.152919**	0.372463
100.0	0.340987	**0.295519**	0.243545	**0.196574**
150.0	**0.241253**	0.360648	**0.172606**	0.273210
200.0	**0.208672**	0.244702	**0.135392**	0.200315
400.0	**0.245526**	0.332591	**0.137816**	0.276372
2000.0	**0.204087**	0.236324	**0.124744**	0.164436
4000.0	**0.204138**	0.241717	**0.127119**	0.148564

We observed that for a fixed number of observations $n = 150$, the mean absolute error of estimation has a variance of **0.001220** for MICE, whereas it is **0.001956** for CODINE. This indicates a reduction in the variance of estimation when using MICE.

Based on the results presented in the previous sections, several conclusions can be drawn. First, MICE appears to be more accurate than CODINE in estimating the density of a copula in a low-data setting. This is supported by the mean absolute error of estimation being lower for MICE compared to CODINE. Additionally, training CODINE on sample extract from IGC, instead of directly training it on data, helps increase its precision. This is evidenced by the mean absolute error of estimation being lower when CODINE is trained on IGC outputs compared to when it is trained directly on data.

However, it is important to note that MICE has difficulties estimating densities in regions where they are very high. This is particularly relevant when there is a strong correlation between two random variables on the tail. As such, it is important to be mindful of this limitation when interpreting the results of the mutual information coefficient calculations.

Overall, MICE appears to be a valuable framework for interpreting the calculations of mutual information coefficient. Its ability to accurately estimate the density of a copula in a low-data setting, as well as its potential to reduce the variance of estimation, make it a promising tool for future research in the field of sensitivity analysis.

Sensitivity analysis on a multimodal distribution In addition to our previous experiments, we perform a sensitivity analysis of a simple function using our model and compare it with state-of-the-art Sobol Indices [3] and MINE [6]. It has been discussed in [27] and [28] that methods based on mutual information outperform variance-based methods like Sobol indices when dealing with *multimodal distributions*. We propose to study such an example, which is particularly relevant in the insurance context for sensitivity analysis of embedded parameters. Let $X \sim N(0,1)$, $Y \sim N(10,1)$ and $Z \sim B(0.5)$. We considere the function :

$$f : (x, y, z) \longrightarrow \left(2x^2 + 3y\right)(zx + (1 - z)y)$$

We are going to perform the sensitivity analysis of $f(X, Y, Z)$ for the variables X and Y and compare it to Sobol indices. Figure 4 shows the function and its densities. We have a multimodal distribution, which is a common occurrence in many real-world applications such as finance, climate modeling, and medical diagnosis. In finance, stock prices and asset returns can have multiple peaks due to market volatility and other factors. In climate modeling, multimodal distributions can arise due to complex interactions between atmospheric, oceanic, and land surface processes. In medical diagnosis, multimodal distributions can occur when there are multiple causes or risk factors for a particular disease or condition. Overall, multimodal distributions are prevalent in many fields and require specialized techniques to analyze and model accurately.

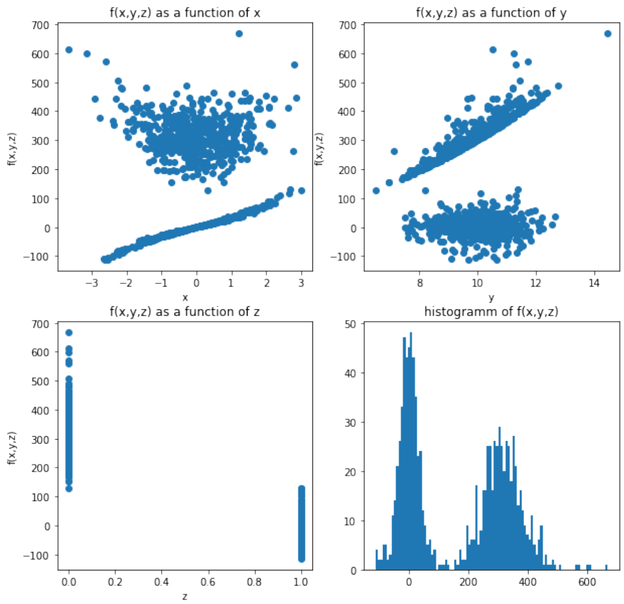

Figure 4: This figure depicts the histogram of our function along with the correlation between the input variables and the output.

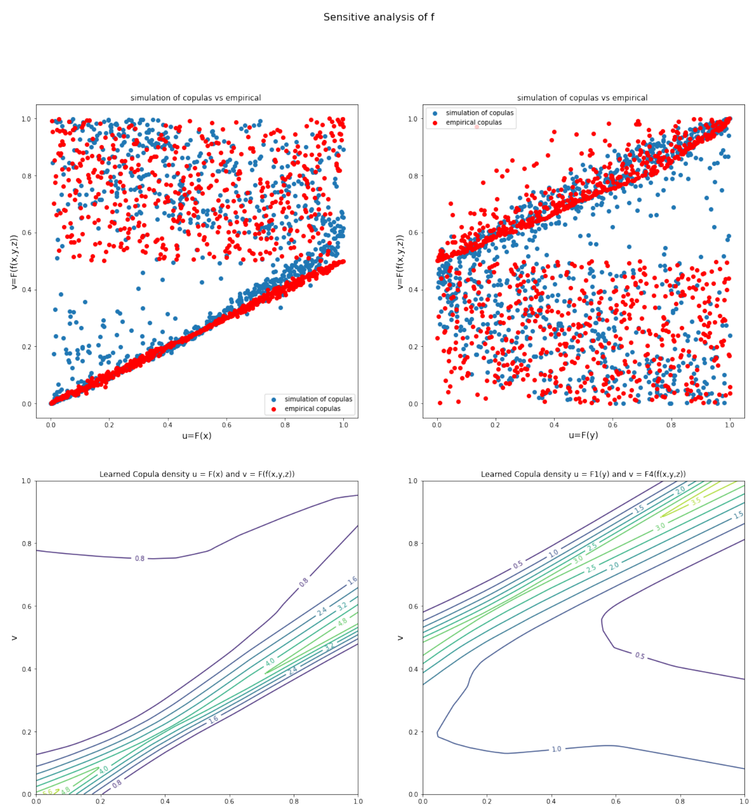

Figure 5: Our model has successfully learned the copulas and density levels. By observing the imperfections, we can have a good idea of the quality of our estimation and anticipate the model's performance.

The mutual information coefficient and Sobol indices were computed and compared. It was found that with 1000 points, Sobol indices did not converge, requiring an increase in the number of points up to 13,000 for an acceptable confidence interval. The resulting values are as follows:

		X	Y	Number of observations
Variance indices	Sobol	0.06	0.04	12000
ρ_{MI}	MICE	0.86	0.73	1000
	MINE	0.94	0.53	1000
	K-NN	0.97	0.87	1000

According to the analysis performed with MICE, there is a strong relationship between the values of X and f, as well as between the values of Y and f. However, Sobol indices fails to identify this relationship and declares X and Y as having no impact on f. We also observed that MICE underestimates the Mutual Information coefficient. This can be explained by the fact that the neural network struggles to separate different regions of space, as shown in Figure 5. To achieve better results, we can train the neural network for a longer period or look for a more complex structure.

An actuarial application We want to show that our model is able to correclty perform in a standard actuarial application and we want to prensent the use of sensitive analysis coefficients.

In this section we take the work of [29]. This study analyzes a savings portfolio without a guaranteed interest rate using an internal model that performs stochastic projections and calculates economic equity after one year. The model considers stock and interest rate risks, as well as lapses of insured parties due to insufficient interest rates. Economic scenarios from December 31, 2008, were used, assuming identical volatility parameters for each set of risk-neutral secondary simulations. In the following paragraph, we use the symbol E to represent the value of an economic variable at the end of a specified time period, and μ_E to denote its expected value. According to the Solvency 2 measure, the amount of capital required can be determined using the following formula:

$$C = \mu_E - q_{0.5\%}(E)$$

where $q_{0.5\%}(E)$ represents the 0.5% quantile of the variable E. It is important to note that the projections of E can change significantly depending on the market data, leading to a large variation in the value of C when changing market data. However, by satisfying certain conditions described in [29], we can calculate the overall capital requirement using the following equation:

$$\mathrm{C}_{SF} = \sqrt{C_S^2 + C_{ZC}^2 + C_{prod}^2 + 2.\rho_{S,ZC} \cdot C_S C_{ZC} + 2\rho_{S,prod} \cdot C_S C_{prod} + 2 \cdot \rho_{ZC,prod} C_{ZC} C_{prod}}$$

(5)

where we have the capital requirement for stocks, C_S; the capital requirement for zero coupons, C_{ZC}; and the capital requirement for cross-effects, C_{prod}. These individual components are defined and discussed in detail in the original article.

C_S	C_{ZC}	C_{prod}
555.9	723.6	227.6

	E_1^S	E_1^{ZC}	E_1^{prod}
E_1^S	1	21.5%	40.2%
E_1^{ZC}	21.5%	1	32.5%
E_1^{prod}	40.2%	32.5%	1

First experiment, Market data varying Our experiment involves conducting a sensitive analysis of correlation parameters in the context of a financial market. To achieve this, we randomly vary the correlation parameters $\rho_{ZC,prod}$, $\rho_{S,prod}$, and $\rho_{S,ZC}$ within the range of $[-0.6, 0.6]$ while ensuring that the correlation matrix remains semi-definite positive at each step. This is achieved by using the uniform distribution, where $\rho_{ZC,prod}, \rho_{S,prod}, \rho_{S,ZC} \sim U([-0.6, 0.6])$.

In addition, we introduce a level of realism by randomly varying the value of C_S, the initial capital in our model. While in a realistic world, the production capital C_{prod} should also vary, for this particular experiment, we assume C_{prod} to be constant. We consider two different values for C_S based on the sign of $\rho_{ZC,prod}$. If $\rho_{ZC,prod} > 0$, then we set $C_S = 900$, and if $\rho_{ZC,prod} < 0$, then we set $C_S = 559.9$.

$$C_S = \begin{cases} 900 & if \quad \rho_{ZC,prod} > 0 \\ 559.9 & if \quad \rho_{ZC,prod} < 0 \end{cases}$$

This create a strong dependence between the capital C_{SF} and $\rho_{ZC,prod}$. Results are described in 6.

Second experiment, Market data constant In the second experiment, we repeat the same process as the first experiment, but we keep C_S fixed at 555.9. In this scenario, where the market data is constant, the correlation between the coefficient $\rho_{ZC,prod}$ and the capital is less significant than in the first experiment.

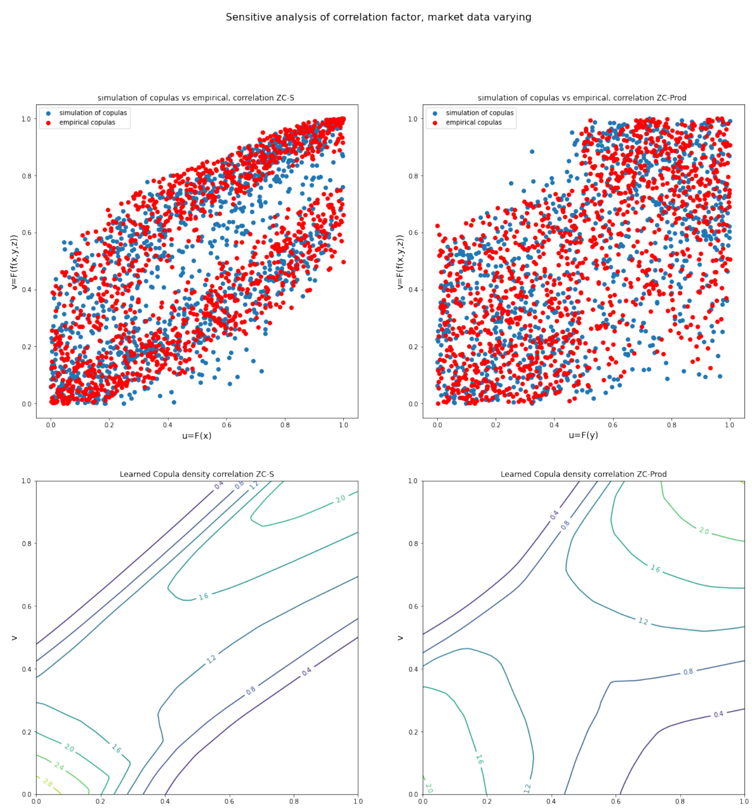

Figure 6: We have effectively trained our model to recognize copulas and density levels. By examining any flaws in our estimation, we can gauge the model's performance and anticipate its accuracy. We also discovered that MICE, when applied to the two dependency structures, produced a highly precise sensitivity index.

		$\rho_{S,ZC}$	$\rho_{S,prod}$	$\rho_{ZC,prod}$
First experiment	Sobol	0.58	0.04	0.02
	MICE	0.72	0	0.59
Second experiment	Sobol	0.77	0.07	0.13
	MICE	0.86	0	0.13

During our investigation, we noticed several things. Firstly, in the initial experiment, where market data was subject to change, we found that the Sobol indices failed to accurately capture the relationship between C_{SF} and $\rho_{ZC,prod}$. Furthermore, the Sobol sensitive index of $\rho_{S,ZC}$ was underestimated due to the multivariate distribution when the market data changed. However, these issues were resolved with MICE. In the second experiment, where the market data remained constant, we observed that both Sobol and MICE produced comparable results, which pleased us. However, in the context of market data changes, MICE outperformed Sobol and was more useful and explainable.

As a conclusion, we perform local sensitivity measure in 7.

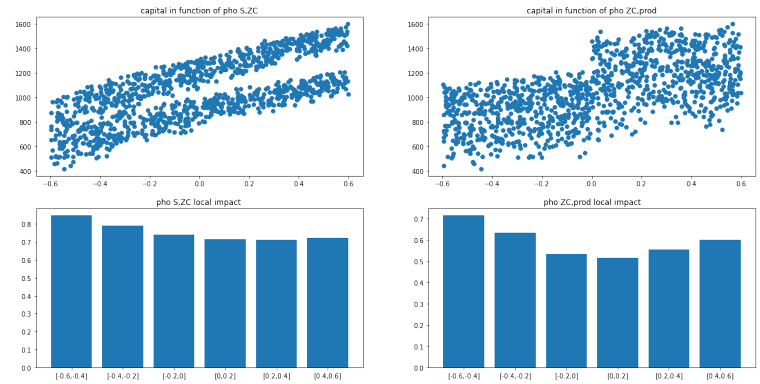

Figure 7: In the context of changing market data, our local sensitive index demonstrates that the influence of $\rho_{ZC,prod}$ is minimized when its correlation is within the range of $[-0.2, 0.2]$. Meanwhile, the impact of $\rho_{S,ZC}$ appears to remain constant, which is consistent with our expectations.

6 Limits and points of attention

Choosing an appropriate neural network architecture for a particular problem can significantly enhance the performance of our method. Our objective was to introduce a new approach for conducting sensitive analyses that prioritizes interpretability and is easy to understand. We opted to utilize the same parameterization for the neural networks across all of our problems to demonstrate a "standard performance". However, readers should explore different parameterizations to ensure that the copulas are correctly learned.

7 Conclusion

In conclusion, we have presented MICE, a new method for sensitivity analysis based on copulas and machine learning. Compared to traditional methods such as Sobol indices and MINE, MICE has several advantages. First, MICE allows for a much finer-grained analysis, providing local sensitivity indices that can reveal the impact of a variable on a function within specific intervals of its domain. Second, MICE is computationally efficient, allowing for quick training of the model and fast computation of the sensitivity indices. Finally, MICE provides an interpretable model that captures the underlying copula structure, making it easier to understand the relationships between variables and their impact on the function. Overall, MICE is a powerful new tool for sensitivity analysis that can provide insights into complex models and help researchers better understand the behavior of their systems.

References

[1] Andrea Saltelli, Marco Ratto, Terry Andres, Francesca Campolongo, Jessica Cariboni, Debora Gatelli, Michaela Saisana, and Stefano Tarantola. *Global sensitivity analysis: the primer*. John Wiley & Sons, 2008.

[2] Qiao Liu and Toshimitsu Homma. A new computational method of a moment-independent uncertainty importance measure. *Reliability Engineering & System Safety*, 94(7):1205–1211, 2009.

[3] Andrea Saltelli. Sensitivity analysis for importance assessment. *Risk analysis*, 22(3):579–590, 2002.

[4] Bertrand Iooss and Andrea Saltelli. Introduction to sensitivity analysis. In *Handbook of uncertainty quantification*, pages 1103–1122. Springer, 2017.

[5] Yu-Jay Huoh. *Sensitivity analysis of stochastic simulators with information theory*. University of California, Berkeley, 2013.

[6] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International conference on machine learning*, pages 531–540. PMLR, 2018.

[7] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.

[8] Umberto Alibrandi, Lars V Andersen, and Enrico Zio. Mutual information for global sensitivity analysis and adaptive-learning surrogate modelling. In *Proceedings of the REC2021, International Workshop on Reliable Engineering Computing, Taormina, Italy*, pages 17–20, 2021.

[9] Harry Joe. *Dependence modeling with copulas*. CRC press, 2014.

[10] Natasa Tagasovska, Damien Ackerer, and Thibault Vatter. Copulas as high-dimensional generative models: Vine copula autoencoders. *Advances in neural information processing systems*, 32, 2019.

[11] Suwon Suh and Seungjin Choi. Gaussian copula variational autoencoders for mixed data. *arXiv preprint arXiv:1604.04960*, 2016.

[12] Tim Janke, Mohamed Ghanmi, and Florian Steinke. Implicit generative copulas. *Advances in Neural Information Processing Systems*, 34:26028–26039, 2021.

[13] Nunzio A Letizia and Andrea M Tonello. Copula density neural estimation. *arXiv preprint arXiv:2211.15353*, 2022.

[14] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[15] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018.

[16] H Carley and MD Taylor. A new proof of sklar's theorem. *Distributions with given marginals and statistical modelling*, pages 29–34, 2002.

[17] Claude E Shannon and Warren Weaver. The mathematical theory of information. *Urbana: University of Illinois Press*, 97(6):128–164, 1949.

[18] Néstor Barraza, Sérgio Moro, Marcelo Ferreyra, and Adolfo de la Peña. Mutual information and sensitivity analysis for feature selection in customer targeting: A comparative study. *Journal of Information Science*, 45(1):53–67, 2019.

[19] Jian Ma and Zengqi Sun. Mutual information is copula entropy. *Tsinghua Science & Technology*, 16(1):51–54, 2011.

[20] Robin AA Ince, Bruno L Giordano, Christoph Kayser, Guillaume A Rousselet, Joachim Gross, and Philippe G Schyns. A statistical framework for neuroimaging data analysis based on mutual information estimated via a gaussian copula. *Human brain mapping*, 38(3):1541–1573, 2017.

[21] X Zeng and TS Durrani. Estimation of mutual information using copula density function. *Electronics letters*, 47(8):493–494, 2011.

[22] Lionel Gueguen, Santiago Velasco-Forero, and Pierre Soille. Local mutual information for dissimilarity-based image segmentation. *Journal of mathematical imaging and vision*, 48(3):625–644, 2014.

[23] XY Sun, Lachlan TH Newham, Barry FW Croke, and John P Norton. Three complementary methods for sensitivity analysis of a water quality model. *Environmental Modelling & Software*, 37:19–29, 2012.

[24] Melisa Reyes-Fuentes, Edmundo del Valle-Gallegos, Julian Duran-Gonzalez, Javier Ortíz-Villafuerte, Rogelio Castillo-Durán, Armando Gómez-Torres, and Cesar Queral. Aztusia: A new application software for uncertainty and sensitivity analysis for nuclear reactors. *Reliability Engineering & System Safety*, 209:107441, 2021.

[25] Andrea Saltelli, Marco Ratto, Stefano Tarantola, and Francesca Campolongo. Sensitivity analysis for chemical models. *Chemical reviews*, 105(7):2811–2828, 2005.

[26] Erik Kole, Kees Koedijk, and Marno Verbeek. Selecting copulas for risk management. *Journal of Banking & Finance*, 31(8):2405–2423, 2007.

[27] Esteve Borras Mora, James Spelling, and Adriaan H van der Weijde. Benchmarking the pawn distribution-based method against the variance-based method in global sensitivity analysis: Empirical results. *Environmental modelling & software*, 122:104556, 2019.

[28] Benjamin Auder and Bertrand Iooss. Global sensitivity analysis based on entropy. In *Safety, reliability and risk analysis-Proceedings of the ESREL 2008 Conference*, pages 2107–2115, 2008.

[29] Laurent Devineau and Stéphane Loisel. Risk aggregation in solvency ii: How to converge the approaches of the internal models and those of the standard formula? 2009.