Reliability-oriented Shapley effets estimation with Normalizing Flows

Lucas Monteiro[1,2,3]

Supervision : F. Bachoc[4], J. Morio[2], J. Demange-Chryst[2]

[1]Institut de Mathématiques de Toulouse, UMR5219 CNRS, 31062 Toulouse, France

[2]ONERA/DTIS, Université de Toulouse, F-31055 Toulouse, France

[3]ANITI, Toulouse, France

[4]Laboratoire Paul Painlevé, UMR8524 CNRS, Université de Lille

October 9th 2025

Uncertainty quantification

Numerical simulations : avoid the cost of experiments, **limit risks**

1. *Model definition* $Y = \phi(\mathbf{X})$
   ϕ numerical code, supposed to be deterministic, costly, black-box
   $\mathbf{X} = (X_1, \ldots, X_d)$ input vector, $\mathbb{R}^d$-valued ; Y the output, $\mathbb{R}$-valued

Uncertainty quantification

Numerical simulations : avoid the cost of experiments, **limit risks**

1. *Model definition* $Y = \phi(\mathbf{X})$
   ϕ numerical code, supposed to be deterministic, costly, black-box
   $\mathbf{X} = (X_1, \ldots, X_d)$ input vector, $\mathbb{R}^d$-valued ; Y the output, $\mathbb{R}$-valued

2. *Quantification of uncertainties' sources*
   $\mathbf{X}$: d-dimensional random vector, known law $\mathbb{P}_{\mathbf{X}}$, with density $f_{\mathbf{X}}$
   Assumptions : dependent inputs, $d \geq 10$, $\mathbb{P}_{\mathbf{X}}$ supposed Gaussian

Uncertainty quantification

Numerical simulations : avoid the cost of experiments, **limit risks**

1. *Model definition* $Y = \phi(\mathbf{X})$

   ϕ numerical code, supposed to be deterministic, costly, black-box

   $\mathbf{X} = (X_1, \ldots, X_d)$ input vector, $\mathbb{R}^d$-valued ; Y the output, $\mathbb{R}$-valued

2. *Quantification of uncertainties' sources*

   $\mathbf{X}$: d-dimensional random vector, known law $\mathbb{P}_{\mathbf{X}}$, with density $f_{\mathbf{X}}$

   Assumptions : dependent inputs, $d \geq 10$, $\mathbb{P}_{\mathbf{X}}$ supposed Gaussian

3. *Uncertainties propagation* : assess the output variability

   Given $\mathbb{P}_{\mathbf{X}}$, we study the r.v. $g(Y)$

Uncertainty quantification

Numerical simulations : avoid the cost of experiments, **limit risks**

1. *Model definition* $Y = \phi(\mathbf{X})$
   ϕ numerical code, supposed to be deterministic, costly, black-box
   $\mathbf{X} = (X_1, \ldots, X_d)$ input vector, $\mathbb{R}^d$-valued ; Y the output, $\mathbb{R}$-valued
2. *Quantification of uncertainties' sources*
   $\mathbf{X}$: d-dimensional random vector, known law $\mathbb{P}_{\mathbf{X}}$, with density $f_{\mathbf{X}}$
   Assumptions : dependent inputs, $d \geq 10$, $\mathbb{P}_{\mathbf{X}}$ supposed Gaussian
3. *Uncertainties propagation* : assess the output variability
   Given $\mathbb{P}_{\mathbf{X}}$, we study the r.v. $g(Y)$
4. *Sensitivity analysis (SA)* : prioritise sources of uncertainty
   How the uncertainty of $g(Y)$ can be attributed to input variable X_i

Variance-based global sensitivity analysis

What are the most influential inputs X_i on the variability of $g(Y) = Y$?

Variance-based global sensitivity analysis

What are the most influential inputs X_i on the variability of $g(Y) = Y$?

For independent inputs, the decomposition of variance (ANOVA) :

$$\mathbb{V}(Y) = \mathbb{V}(\phi(\mathbf{X})) = \sum_{u \subset \{1,..,d\} \setminus \{\emptyset\}} \mathbb{V}(\mathbb{E}[\phi(\mathbf{X})|\mathbf{X}_u] + Q_{v \subsetneq u})$$

leads to the closed Sobol' indices S_u^c for group $u \subset \{1, \ldots, d\}$:

$$\mathsf{S}_u^c = \frac{\mathbb{V}(\mathbb{E}[\phi(\mathbf{X})|\mathbf{X}_u])}{\mathbb{V}(\phi(\mathbf{X}))}$$

Variance-based global sensitivity analysis

What are the most influential inputs X_i on the variability of $g(Y) = Y$?

For independent inputs, the decomposition of variance (ANOVA) :

$$\mathbb{V}(Y) = \mathbb{V}(\phi(\mathbf{X})) = \sum_{u \subset \{1,..,d\} \setminus \{\emptyset\}} \mathbb{V}(\mathbb{E}[\phi(\mathbf{X})|\mathbf{X}_u] + Q_{v \subsetneq u})$$

leads to the closed Sobol' indices S_u^c for group $u \subset \{1, \ldots, d\}$:

$$\mathrm{S}_u^c = \frac{\mathbb{V}(\mathbb{E}[\phi(\mathbf{X})|\mathbf{X}_u])}{\mathbb{V}(\phi(\mathbf{X}))}$$

Convenient to interpret, but costly to estimate and require independence

Dependent inputs : ANOVA not unique, loosing its interpretation's power

How to deal with dependent inputs ?

Global Sensitivity analysis : Shapley effects

Among several available methods : **Shapley effects**

- From game theory (Shapley, 1953)
- Provide a fairly allocation of gains between players

Global Sensitivity analysis : Shapley effects

Among several available methods : **Shapley effects**

- From game theory (Shapley, 1953)
- Provide a fairly allocation of gains between players

The *Shapley effect* Sh_i associated with the *player/input* i is defined by

$$\mathrm{Sh}_i = \frac{1}{d} \sum_{u \subset \{1,\dots,d\} \setminus \{i\}} \binom{d-1}{|u|}^{-1} \left(c(u \cup \{i\}) - c(u) \right)$$

c a cost function measuring contribution

$c(u \cup \{i\}) - c(u)$: contribution of i to the group u

Global Sensitivity analysis : Shapley effects

Among several available methods : **Shapley effects**

- From game theory (Shapley, 1953)
- Provide a fairly allocation of gains between players

The *Shapley effect* Sh_i associated with the *player/input* i is defined by

$$\mathrm{Sh}_i = \frac{1}{d} \sum_{u \subset \{1,\dots,d\} \setminus \{i\}} \binom{d-1}{|u|}^{-1} \left(c(u \cup \{i\}) - c(u) \right)$$

c a cost function measuring contribution

$c(u \cup \{i\}) - c(u)$: contribution of i to the group u

Adaptation to sensitivity analysis (Owen, 2014) : $c(u) = \mathrm{S}_u^c$

Reliability analysis

For safety and certification purposes : need to understand failure scenarios

Failure : rare event, abnormal state, catastrophic event (important loss)

Aim : assess the risk $\rightarrow$ *estimate the failure probability*

Reliability analysis

For safety and certification purposes : need to understand failure scenarios

Failure : rare event, abnormal state, catastrophic event (important loss)

Aim : assess the risk $\rightarrow$ *estimate the failure probability*

Failure event is $\{Y > t\}$, and the *failure probability* is

$$p_t = \mathbb{P}(Y > t) = \mathbb{E}[\mathbf{1}_{]t,\infty[}(Y)] = \mathbb{E}[\mathbf{1}_{F_t}(\mathbf{X})] = \int_{F_t} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}$$

- $t \in \mathbb{R}$ the failure threshold
- $F_t = \{\mathbf{x} \in \mathbb{R}^d : \phi(\mathbf{x}) > t\}$ the failure domain

Failure probability and failing samples

To obtain a good estimation of p_t with a moderate number of calls to ϕ :

- Monte Carlo (not adapted to very small p_t)
- Importance Sampling
- Subset Sampling
- Control variates

Failure probability and failing samples

To obtain a good estimation of p_t with a moderate number of calls to ϕ :

- Monte Carlo (not adapted to very small p_t)
- Importance Sampling
- Subset Sampling
- Control variates

Recover N_f *failing samples* :

$$(\mathbf{X}^{(n)}) \text{ satisfying } \phi(\mathbf{X}^{(n)}) > t, \text{ denoted } (\widetilde{\mathbf{X}}^{(n)})$$

Failure probability and failing samples

To obtain a good estimation of p_t with a moderate number of calls to ϕ :

- Monte Carlo (not adapted to very small p_t)
- Importance Sampling
- Subset Sampling
- Control variates

Recover N_f *failing samples* :

$$(\mathbf{X}^{(n)}) \text{ satisfying } \phi(\mathbf{X}^{(n)}) > t, \text{ denoted } (\widetilde{\mathbf{X}}^{(n)})$$

Objective

Without additional call to ϕ : perform a **sensitivity analysis for $\mathbf{1}_{F_t}(\mathbf{X})$**

Reliability-oriented Sensitivity analysis (ROSA)

Sensitivity analysis is performed on $\mathbf{1}_{F_t}(\mathbf{X})$ instead of $\phi(\mathbf{X})$

Reliability-oriented Sensitivity analysis (ROSA)

Sensitivity analysis is performed on $\mathbf{1}_{F_t}(\mathbf{X})$ instead of $\phi(\mathbf{X})$

We can define T-S_u^c, the target closed Sobol index of u, by

$$\text{T-S}_u^c = \frac{\mathbb{V}(\mathbb{E}[\mathbf{1}_{F_t}(\mathbf{X})|\mathbf{X}_u])}{\mathbb{V}(\mathbf{1}_{F_t}(\mathbf{X}))}$$

Reliability-oriented Sensitivity analysis (ROSA)

Sensitivity analysis is performed on $\mathbf{1}_{F_t}(\mathbf{X})$ instead of $\phi(\mathbf{X})$

We can define T-S_u^c, the target closed Sobol index of u, by

$$\text{T-S}_u^c = \frac{\mathbb{V}(\mathbb{E}[\mathbf{1}_{F_t}(\mathbf{X})|\mathbf{X}_u])}{\mathbb{V}(\mathbf{1}_{F_t}(\mathbf{X}))}$$

Leading to T-Sh_i, the target Shapley effect of X_i (Il Idrissi et al., 2021)

$$\text{T-Sh}_i = \frac{1}{d} \sum_{u \subset \{1,\ldots,d\} \setminus \{i\}} \binom{d-1}{|u|}^{-1} (\text{T-S}_{u \cup \{i\}}^c - \text{T-S}_u^c)$$

Estimation of T-Sh$_i$

Existing literature for the estimation of T-Sh$_i$

Il Idrissi et al. (2021)

- Estimate T-S$_u^c$ by crude double Monte-Carlo
- Require too many calls to ϕ when $p_t \ll 1$
  (because $\mathbf{1}_{F_t}(\mathbf{X}^{(n)}) = 0$ for many samples $(\mathbf{X}^{(n)})$)

Demange-Chryst et al. (2023)

- Estimate T-S$_u^c$ by importance sampling (better precision)
- Allow estimation of T-Sh$_i$ with the same samples used to estimate p_t
- Limited to dimension $d < 10$

Contribution

Our contribution :

Extend estimation scheme of **target Shapley effects for larger dimensions**

Contribution

Our contribution :

Extend estimation scheme of **target Shapley effects for larger dimensions**

Overall methodolody to estimate T-Sh$_i$

1. Estimate p_t and obtain failing samples $(\widetilde{\mathbf{X}}^{(n)})$
2. Rewrite T-S$_u^c$ with conditional density $f_{\mathbf{X}_u|F_t}$ of $\mathbf{X}_u$
3. Estimate $f_{\mathbf{X}_u|F_t}$ with Normalizing Flows (suited for large dimensions)
4. Estimate T-S$_u^c$ by Monte-Carlo
5. Estimate T-Sh$_i$ with a another writing using permutations

Rewrite T-S_u^c

Alternative writing of target closed Sobol index (Perrin et Defaux, 2019)

$$\text{T-S}_u^c = \frac{p_t}{1-p_t}\mathbb{V}_{\mathbf{X}_u}\left[\frac{f_{\mathbf{X}_u|F_t}(\mathbf{X}_u)}{f_{\mathbf{X}_u}(\mathbf{X}_u)}\right]$$

$$= \frac{p_t}{1-p_t}\left(\mathbb{E}_{\mathbf{X}_u|F_t}\left[\frac{f_{\mathbf{X}_u|F_t}(\mathbf{X}_u)}{f_{\mathbf{X}_u}(\mathbf{X}_u)}\right]-1\right)$$

$f_{\mathbf{X}_u}$ marginal density of $\mathbf{X}_u$

$f_{\mathbf{X}_u|F_t}$ marginal density of $\mathbf{X}_u$ conditionally to the failure $\phi(\mathbf{X}) > t$

Estimation of T-S_u^c

p_t already estimated by $\widehat{p_t}$ in reliability analysis

$f_{\mathbf{X}}$ is Gaussian $\implies$ we can obtain $f_{\mathbf{X}_u}$

$f_{\mathbf{X}_u|F_t}$ must be estimated, **but !**

$\rightarrow$ may be large dimensional $(1 \leq \#(u) \leq d)$

$\rightarrow$ classical methods (KDE) suffer from the curse of dimensionality

$\rightarrow$ classical parametric methods lack flexibility

$\rightarrow$ resulting estimate must be tractable to compute $\mathbb{E}_{\mathbf{X}_u|F_t}[\cdot]$

Estimation of T-S_u^c

p_t already estimated by $\widehat{p_t}$ in reliability analysis

$f_{\mathbf{X}}$ is Gaussian $\implies$ we can obtain $f_{\mathbf{X}_u}$

$f_{\mathbf{X}_u|F_t}$ must be estimated, **but !**

$\rightarrow$ may be large dimensional $(1 \leq \#(u) \leq d)$

$\rightarrow$ classical methods (KDE) suffer from the curse of dimensionality

$\rightarrow$ classical parametric methods lack flexibility

$\rightarrow$ resulting estimate must be tractable to compute $\mathbb{E}_{\mathbf{X}_u|F_t}[\cdot]$

Proposed solution : Normalizing Flows

Estimate $f_{\mathbf{X}_u|F_t}$ with Normalizing Flows

Normalizing Flows (NF) : Papamakarios et al. (2021)

- From the field of generative modelling
- Flexible, suited for complex high-dimensional density estimation
- Provide explicit and tractable density (unlike GAN, VAE, etc.)

Estimate $f_{\mathbf{X}_u|F_t}$ with Normalizing Flows

Normalizing Flows (NF) : Papamakarios et al. (2021)

- From the field of generative modelling
- Flexible, suited for complex high-dimensional density estimation
- Provide explicit and tractable density (unlike GAN, VAE, etc.)

Principle

1. From a density $f_{\mathbf{Z}}$ easy to evaluate (Gaussian), same dimension as $f_{\mathbf{X}_u|F_t}$

Estimate $f_{\mathbf{X}_u|F_t}$ with Normalizing Flows

Normalizing Flows (NF) : Papamakarios et al. (2021)

- From the field of generative modelling
- Flexible, suited for complex high-dimensional density estimation
- Provide explicit and tractable density (unlike GAN, VAE, etc.)

Principle

1. From a density $f_{\mathbf{Z}}$ easy to evaluate (Gaussian), same dimension as $f_{\mathbf{X}_u|F_t}$
2. Build a C^1-diffeomorphism T_θ, parametrized by θ, providing the density

$$f_\theta(x) = f_{T_\theta(\mathbf{Z})}(x) = f_{\mathbf{Z}}\left(T_\theta^{-1}(x)\right) \left|\det J_{T_\theta^{-1}}(x)\right|$$

Estimate $f_{\mathbf{X}_u|F_t}$ with Normalizing Flows

Normalizing Flows (NF) : Papamakarios et al. (2021)

- From the field of generative modelling
- Flexible, suited for complex high-dimensional density estimation
- Provide explicit and tractable density (unlike GAN, VAE, etc.)

Principle

1. From a density $f_{\mathbf{Z}}$ easy to evaluate (Gaussian), same dimension as $f_{\mathbf{X}_u|F_t}$
2. Build a C^1-diffeomorphism T_θ, parametrized by θ, providing the density

$$f_\theta(x) = f_{T_\theta(\mathbf{Z})}(x) = f_{\mathbf{Z}}\left(T_\theta^{-1}(x)\right) \left|\det J_{T_\theta^{-1}}(x)\right|$$

3. NF learns θ by minimizing the *KL divergence* $D_{KL}(f_{\mathbf{X}_u|F_t} \| f_\theta)$

Estimate $f_{\mathbf{X}_u|F_t}$ with Normalizing Flows

Normalizing Flows (NF) : Papamakarios et al. (2021)

- From the field of generative modelling
- Flexible, suited for complex high-dimensional density estimation
- Provide explicit and tractable density (unlike GAN, VAE, etc.)

Principle

1. From a density $f_{\mathbf{Z}}$ easy to evaluate (Gaussian), same dimension as $f_{\mathbf{X}_u|F_t}$
2. Build a C^1-diffeomorphism T_θ, parametrized by θ, providing the density

$$f_\theta(x) = f_{T_\theta(\mathbf{Z})}(x) = f_{\mathbf{Z}}\left(T_\theta^{-1}(x)\right) \left|\det J_{T_\theta^{-1}}(x)\right|$$

3. NF learns θ by minimizing the *KL divergence* $D_{KL}(f_{\mathbf{X}_u|F_t} \| f_\theta)$
4. Learned with $(\widetilde{\mathbf{X}}^{(n)})_{n=1}^{N_f}$, $\widehat{\theta}$ provides the estimated density $\widehat{f}_{\mathbf{X}_u|F_t} = f_{T_{\hat{\theta}}(\mathbf{Z})}$

Estimation of T-S_u^c with Monte Carlo

With $\widehat{p}_t$, $\widehat{f}_{\mathbf{X}_u|F_t}$ and $f_{\mathbf{X}_u}$ $\implies$ estimation of $\mathbb{E}_{\mathbf{X}_u|F_t}[\cdot]$ and T-S_u^c

Estimation of T-S_u^c with Monte Carlo

With $\widehat{p}_t$, $\widehat{f}_{\mathbf{X}_u|F_t}$ and $f_{\mathbf{X}_u}$ $\implies$ estimation of $\mathbb{E}_{\mathbf{X}_u|F_t}[\cdot]$ and T-S_u^c

With $\widehat{f}_{\mathbf{X}_u|F_t}$ and $f_{\mathbf{X}_u}$ we estimate $E_u = \mathbb{E}_{\mathbf{X}_u|F_t}\left[\frac{f_{\mathbf{X}_u|F_t}(\mathbf{X}_u)}{f_{\mathbf{X}_u}(\mathbf{X}_u)}\right]$ by Monte Carlo :

$$\widehat{E}_u = \frac{1}{N_f}\sum_{n=1}^{N_f}\frac{\widehat{f}_{\mathbf{X}_u|F_t}\left(\widetilde{\mathbf{X}}_u^{(n)}\right)}{f_{\mathbf{X}_u}\left(\widetilde{\mathbf{X}}_u^{(n)}\right)}$$

Estimation of T-S_u^c with Monte Carlo

With $\widehat{p}_t$, $\widehat{f}_{\mathbf{X}_u|F_t}$ and $f_{\mathbf{X}_u}$ $\implies$ estimation of $\mathbb{E}_{\mathbf{X}_u|F_t}[\cdot]$ and T-S_u^c

With $\widehat{f}_{\mathbf{X}_u|F_t}$ and $f_{\mathbf{X}_u}$ we estimate $E_u = \mathbb{E}_{\mathbf{X}_u|F_t}\left[\frac{f_{\mathbf{X}_u|F_t}(\mathbf{X}_u)}{f_{\mathbf{X}_u}(\mathbf{X}_u)}\right]$ by Monte Carlo :

$$\widehat{E}_u = \frac{1}{N_f}\sum_{n=1}^{N_f}\frac{\widehat{f}_{\mathbf{X}_u|F_t}\left(\widetilde{\mathbf{X}}_u^{(n)}\right)}{f_{\mathbf{X}_u}\left(\widetilde{\mathbf{X}}_u^{(n)}\right)}$$

Then, $\widehat{E}_u$ and $\widehat{p}_t$ provide the estimate $\widehat{\text{T-S}_u^c}$ of T-S_u^c :

$$\widehat{\text{T-S}_u^c} = \frac{\widehat{p}_t}{1-\widehat{p}_t}\left[\widehat{E}_u - 1\right]$$

Estimation of T-Sh$_i$ with permutations

Subset aggregation

- compute $\widehat{\text{T-S}}_u^c$ for every $u \subset \{1, \ldots, d\}$, nb of indices is $O(2^d)$

Estimation of T-Sh$_i$ with permutations

Subset aggregation

- compute $\widehat{\text{T-S}}_u^c$ for every $u \subset \{1, \ldots, d\}$, nb of indices is $O(2^d)$

Permutation aggregation (Castro et al. 2009) :

$$\text{T-Sh}_i = \frac{1}{d!} \sum_{\sigma \in \mathcal{S}_d} \left(\text{T-S}^c_{P_i(\sigma) \cup \{i\}} - \text{T-S}^c_{P_i(\sigma)} \right)$$

- $\mathcal{S}(d)$: set of permutations of $\{1, \ldots, d\}$
- $P_i(\sigma)$: set of indices before i in permutation σ

Estimation of T-Sh_i with permutations

Subset aggregation

- compute $\widehat{\text{T-S}}^c_u$ for every $u \subset \{1, \dots, d\}$, nb of indices is $O(2^d)$

Permutation aggregation (Castro et al. 2009) :

$$\text{T-Sh}_i = \frac{1}{d!} \sum_{\sigma \in \mathcal{S}_d} \left(\text{T-S}^c_{P_i(\sigma) \cup \{i\}} - \text{T-S}^c_{P_i(\sigma)}\right)$$

- $\mathcal{S}(d)$: set of permutations of $\{1, \dots, d\}$
- $P_i(\sigma)$: set of indices before i in permutation σ

Example : $d = 3$, compute T-Sh_1 for the variable X_1

$(2, 1, 3) \quad \rightarrow \quad \text{T-S}^c_{\{2\}} - \text{T-S}^c_{\emptyset} \quad ; \quad \text{T-S}^c_{\{2,1\}} - \text{T-S}^c_{\{2\}} \quad ; \quad \text{T-S}^c_{\{2,1,3\}} - \text{T-S}^c_{\{2,1\}}$

Estimation of T-Sh$_i$

1. Uniformly sample M permutations $(\sigma^{(m)})_{m=1}^{M}$ from $\mathcal{S}(d)$

2. Build the estimate

$$\widehat{\text{T-Sh}}_i = \frac{1}{M}\sum_{m=1}^{M}(\widehat{\text{T-S}}^{c}_{P_i(\sigma)\cup\{i\}} - \widehat{\text{T-S}}^{c}_{P_i(\sigma)})$$

Estimation of T-Sh$_i$

1. Uniformly sample M permutations $(\sigma^{(m)})_{m=1}^{M}$ from $\mathcal{S}(d)$

2. Build the estimate

$$\widehat{\text{T-Sh}}_i = \frac{1}{M}\sum_{m=1}^{M}(\widehat{\text{T-S}}^c_{P_i(\sigma)\cup\{i\}} - \widehat{\text{T-S}}^c_{P_i(\sigma)})$$

Advantages

- Number of indices T-S^c_u can be reduced to $M(d-1)$ (Song, 2016)
- Allow trade-off between precision and computational cost (Maleki, 2013)
- Allow to obtain confidence interval or exact bounds (Maleki, 2013)

But ! Induce additional variability, unlike the subset method

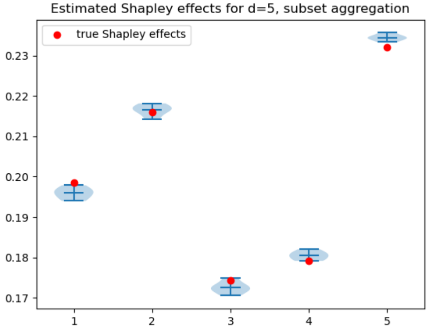
Numerical results

Gaussian Linear case

- $\mathbf{X} \sim \mathcal{N}(\mu, \Sigma)$, $\mu \in \mathbb{R}^d$, $\Sigma \in \mathcal{M}_d(\mathbb{R})$ and non-diagonal, $d = 7$
- For $\beta \in \mathbb{R}^d \setminus \{\mathbf{0}\}$, define $Y = \beta^\top \mathbf{X}$, $t = 5.5$, $\widehat{p_t}$ obtained with MC
- We have formula to obtain the true Shapley effects
- Repeat estimation scheme 5 times to obtain uncertainty

Gaussian Linear case

- $\mathbf{X} \sim \mathcal{N}(\mu, \Sigma)$, $\mu \in \mathbb{R}^d$, $\Sigma \in \mathcal{M}_d(\mathbb{R})$ and non-diagonal, $d = 7$
- For $\beta \in \mathbb{R}^d \backslash \{\mathbf{0}\}$, define $Y = \beta^\top \mathbf{X}$, $t = 5.5$, $\widehat{p}_t$ obtained with MC
- We have formula to obtain the true Shapley effects
- Repeat estimation scheme 5 times to obtain uncertainty



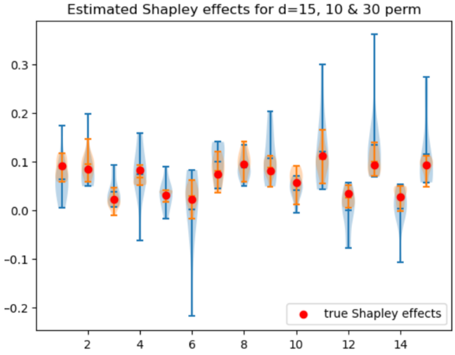
Gaussian Linear case

- $\mathbf{X} \sim \mathcal{N}(\mu, \Sigma)$, $\mu \in \mathbb{R}^d$, $\Sigma \in \mathcal{M}_d(\mathbb{R})$ and non-diagonal, $d = 15$
- For $\beta \in \mathbb{R}^d \setminus \{\mathbf{0}\}$, define $Y = \beta^\top \mathbf{X}$, $t = 8.5$, $\widehat{p}_t$ obtained with MC
- Repeat estimation scheme 10 times to obtain uncertainty
- 10 perm. (140 Sobol') and 30 perm. (420 Sobol') instead of 2^{15}

Numerical results

Gaussian Linear case

- $\mathbf{X} \sim \mathcal{N}(\mu, \Sigma)$, $\mu \in \mathbb{R}^d$, $\Sigma \in \mathcal{M}_d(\mathbb{R})$ and non-diagonal, $d = 15$
- For $\beta \in \mathbb{R}^d \backslash \{\mathbf{0}\}$, define $Y = \beta^\top \mathbf{X}$, $t = 8.5$, $\widehat{p}_t$ obtained with MC
- Repeat estimation scheme 10 times to obtain uncertainty
- 10 perm. (140 Sobol') and 30 perm. (420 Sobol') instead of 2^{15}



Conclusion

Contribution

Numerical code with a unique sample of correlated inputs, $d \geq 10$

Estimate Target Shapley effects with Normalizing Flows

Good results for $d = 15$

Conclusion

Contribution

Numerical code with a unique sample of correlated inputs, $d \geq 10$

Estimate Target Shapley effects with Normalizing Flows

Good results for $d = 15$

Improve estimation of target Shapley effects

Measure uncertainty from the NF **and** from estimations of T-S_u^c

Extend to applications with larger dimensions

→ Gaussian linear case with $d \geq 20$

→ Fire-spread model ($d = 10$), existing estimation is not good enough

Compare with existing target indices

Conclusion

Contribution

Numerical code with a unique sample of correlated inputs, $d \geq 10$

Estimate Target Shapley effects with Normalizing Flows

Good results for $d = 15$

Improve estimation of target Shapley effects

Measure uncertainty from the NF **and** from estimations of T-S_u^c

Extend to applications with larger dimensions

$\rightarrow$ Gaussian linear case with $d \geq 20$

$\rightarrow$ Fire-spread model ($d = 10$), existing estimation is not good enough

Compare with existing target indices

The Future : Optimal UQ, Copula learning with Normalizing Flows ?

References

L. Shapley, *A Value for n-Person Games*, 1953

A. Owen , *Sobol' Indices and Shapley Value*, 2014

M. Il Idrissi, V. Chabridon, B. Iooss, *Developments and applications of Shapley effects to reliability-oriented sensitivity analysis with correlated inputs*, 2021

J. Demange-Chryst, F. Bachoc, J. Morio, *Shapley effect estimation in reliability-oriented sensitivity analysis with correlated inputs by importance sampling*, 2023

G. Defaux, G. Perrin, *Efficient evaluation of reliability-oriented sensitivity indices*, 2019

G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, B. Lakshminarayanan, *Normalizing Flows for Probabilistic Modeling and Inference* , 2021

J. Castro, D. Gomez, J. Tejada, *Polynomial calculation of the Shapley value based on sampling*, 2009

E. Song, B. L. Nelson, J. Staum, *Shapley effects for global sensitivity analysis: Theory and computation*, 2016

S. Maleki, L. Tran-Thanh, G. Hines, T. Rahwan, A. Rogers, *Bounding the Estimation Error of Sampling-based Shapley Value Approximation*, 2013