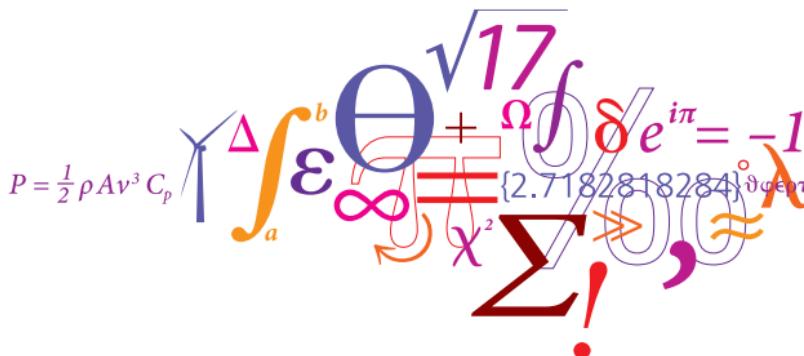


Polynomial Chaos Expansions for Wind Energy: Tutorial

Juan P. Murcia: PhD. student, jumu@dtu.dk

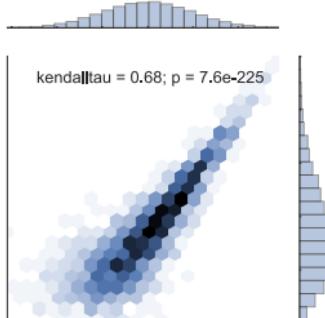
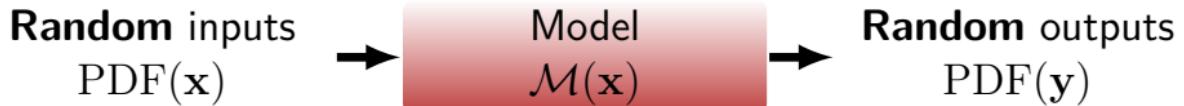
Pierre-E. Réthoré: Supervisor, Senior Scientist



Outline

- ① (A very short) Introduction to Polynomial Chaos Expansions
- ② A1. 2D Not correlated Normal
- ③ A2. 2D MvNormal
- ④ A3. 2D Conditionally Correlated
- ⑤ A4. 2D Copula Correlated
- ⑥ Conclusions

Uncertainty propagation problem



?

(A very short) Introduction to PCE

Monte-Carlo simulation

- Obtain a response sample by evaluating the model in each input realization:
 $y_i = \mathcal{M}(x_i)$
- Input sample, x_i , can be generated using advanced sampling methods such as: Latin Hypercube sampling (LHS), Halton or Hammersley sequences.
- **Pros:** Very robust and easy to implement and parallelize
- **Cons:** Convergence is slow ($\propto N^{-1/2}$)

Polynomial Chaos expansion

- Build a polynomial surrogate of the model: $y(\mathbf{x}) \approx \sum c_l \Psi_l(\mathbf{x})$
- A polynomial basis, $\Psi_l(\mathbf{x})$, is built with respect to PDF(\mathbf{x})
- The model is evaluated, $y_i = \mathcal{M}(x_i)$, and *projected/fitted* to the polynomial basis.
- The mean $\mathbb{E}(y)$, variance $\mathbb{V}(y)$ and Sobol's sensitivity index for each input S_i are obtain from the polynomial coefficients c_l .
- **Pros:** Convergence is fast ($\propto N^{-m}$, $m > 1$, m is problem dependent)
- **Cons:** How to define the order of the polynomials in each variable? How to avoid over fitting the model (Gibbs oscillations)?

(A very short) Introduction to PCE

Single uncertain variable x

$$y(x) \approx \sum_{l=0}^P c_l \Psi_l(x)$$

Define an inner product using PDF(x):

$$\langle f, g \rangle = \int f(x) g(x) \text{PDF}(x) dx$$

The polynomial basis is constructed such that $\Psi_0 = 1$ and:

$$\langle \Psi_l, \Psi_k \rangle = \begin{cases} 1 & \text{if } l = k \\ 0 & \text{if } l \neq k \end{cases}$$

$$\langle 1, \Psi_l \rangle = 0 \quad \forall l > 0 \quad \iff \quad \int \Psi_l(x) \text{PDF}(x) dx = 0 \quad \forall l > 0$$

Methods to find the coefficients c_l

Semi-Spectral projection (quadrature integration)

- Use a quadrature rule to approximate the integrals (nodes, x_i and weights ω_i). Gaussian quadrature is widely used.

$$c_l = \langle y, \Psi_l \rangle = \int y(x) \Psi_l(x) \text{PDF}(x) dx \approx \sum_{i=0}^N \omega_i y(x_i) \Psi_l(x_i)$$

- **Pros:** Very good for low number of dimensions
- **Cons:** Unstable for heavy tailed PDFs. Quadrature rules fail with most correlated variables

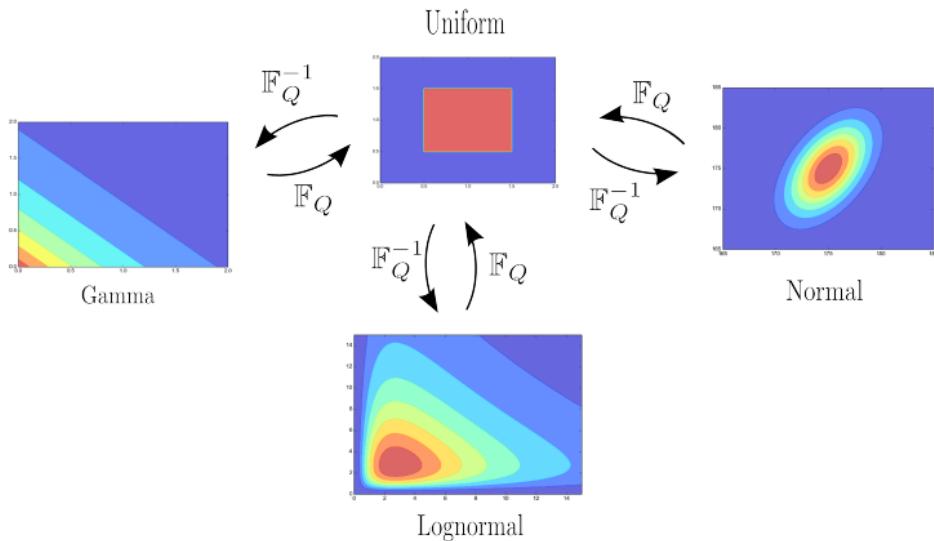
Point collocation (polynomial fit)

- Generate a small sample and fit the polynomial basis using Least squares or some other optimization method (e.g. LAR, LASSO).
- **Pros:** Very robust. Optimization algorithms are designed to handle large number of dimensions (sparsity) and correlated inputs.
- **Cons:** Not as efficient as semi-spectral collocation.

How to deal with correlated inputs?

Rosenblatt Transformation [Rosenblatt 1952]

- Transforms the correlated input variables (x) into a multi-dimensional uncorrelated uniform space (w). Solve the propagation problem in the uncorrelated space:
 $y(x) = y(\mathbb{F}_Q^{-1}(w))$. Use Legendre polynomials for uniform variables.
- It consists in using the inverse of the CDF of each variable in sequence. Chaospy includes this transformation [Feinberg 2015]. Graph reproduced from Chaospy tutorials.

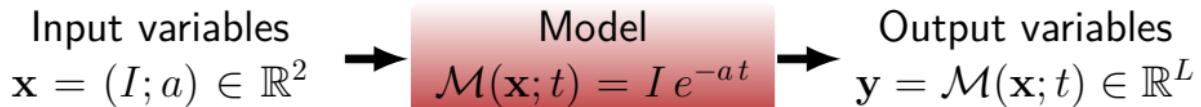


Outline

- ① (A very short) Introduction to Polynomial Chaos Expansions
- ② A1. 2D Not correlated Normal
- ③ A2. 2D MvNormal
- ④ A3. 2D Conditionally Correlated
- ⑤ A4. 2D Copula Correlated
- ⑥ Conclusions

Case A: A simple model

Model Description



Variables without uncertainty

- t : Location of the evaluation.

Uncertain variables

- I : Initial condition.
 - a : Rate of dissipation.

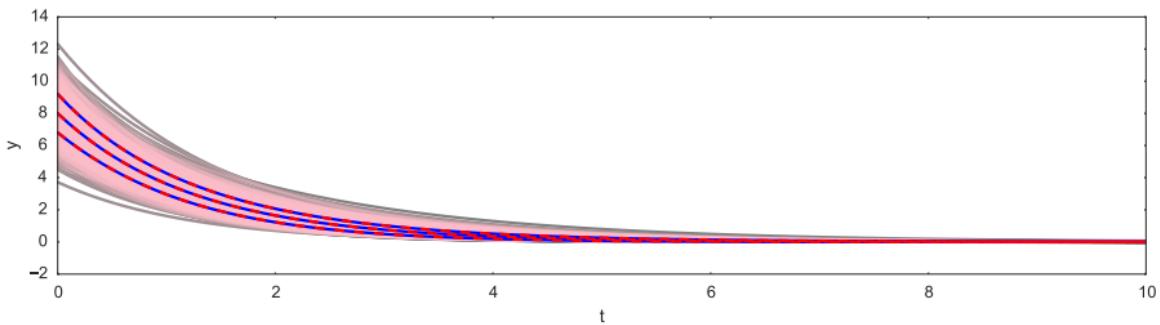
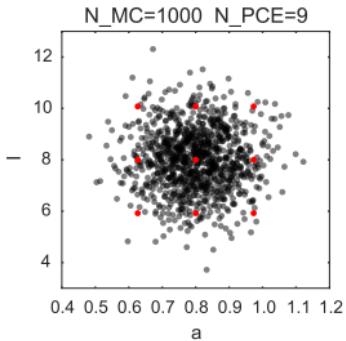
Case A1: A simple model

Inputs: 2D Not correlated Normal

$$y = I e^{-a t}$$

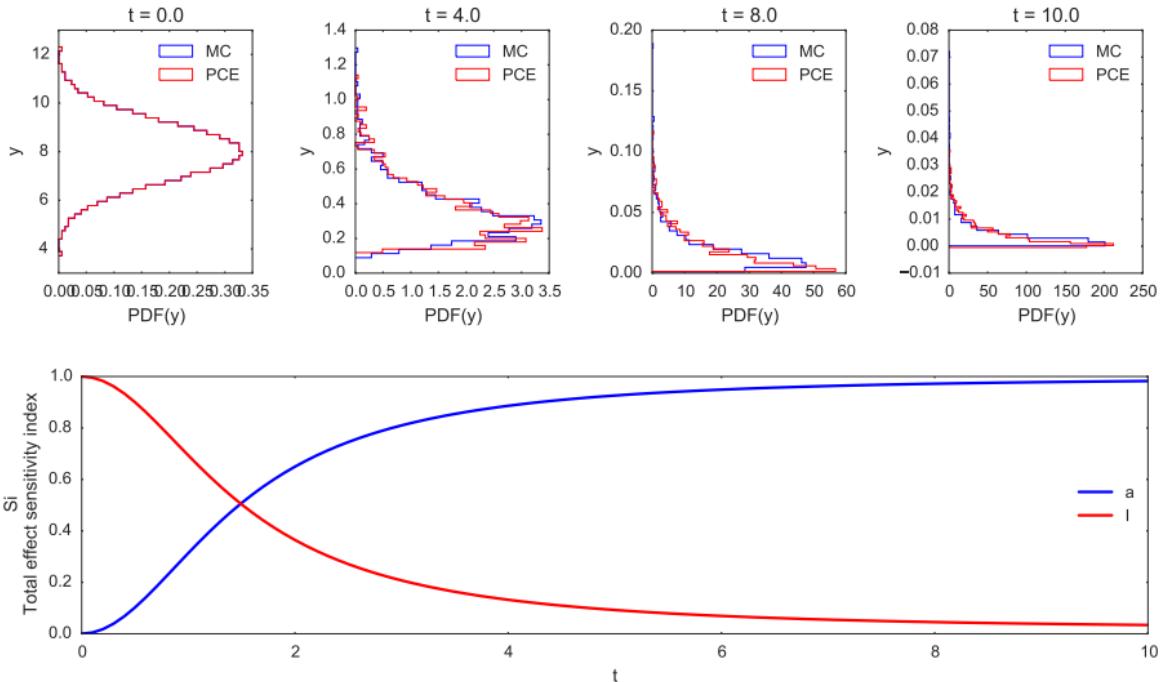
$$I \sim \text{Normal}(\mu_I = 8., \sigma_I = 2^{1/2})$$

$$a \sim \text{Normal}(\mu_a = 0.8, \sigma_a = 0.01^{1/2})$$



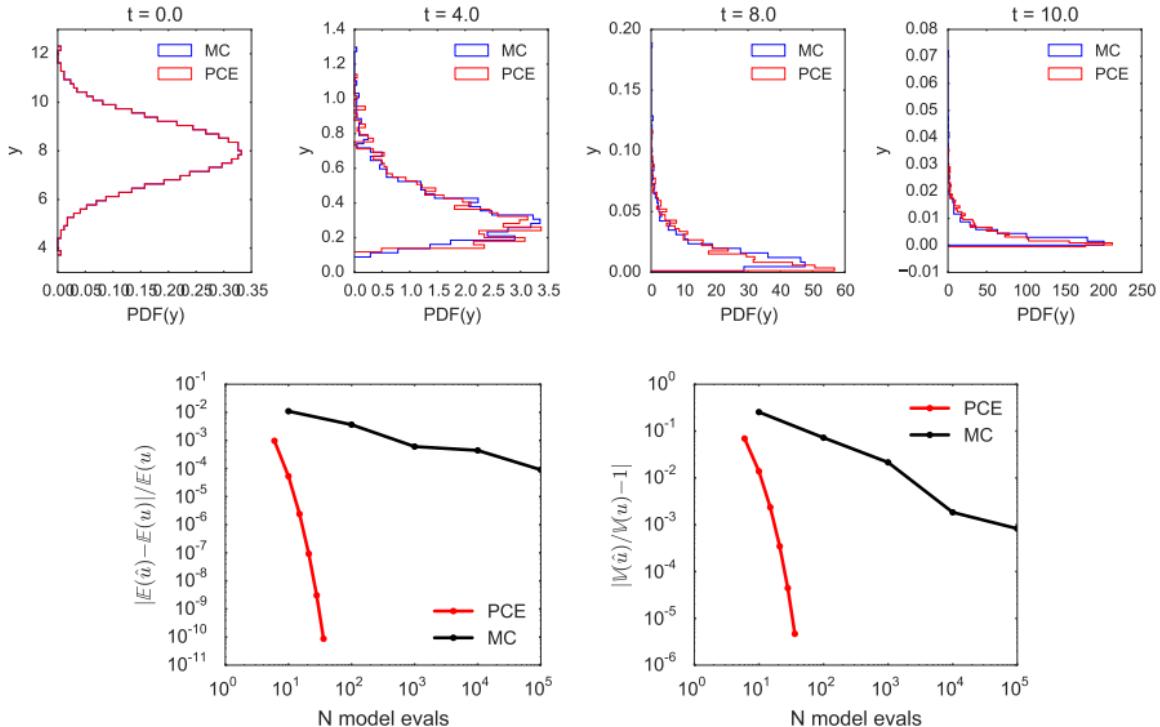
Case A1: A simple model

Inputs: 2D Not correlated Normal



Case A1: A simple model

Inputs: 2D Not correlated Normal



Outline

- ① (A very short) Introduction to Polynomial Chaos Expansions
- ② A1. 2D Not correlated Normal
- ③ A2. 2D MvNormal
- ④ A3. 2D Conditionally Correlated
- ⑤ A4. 2D Copula Correlated
- ⑥ Conclusions

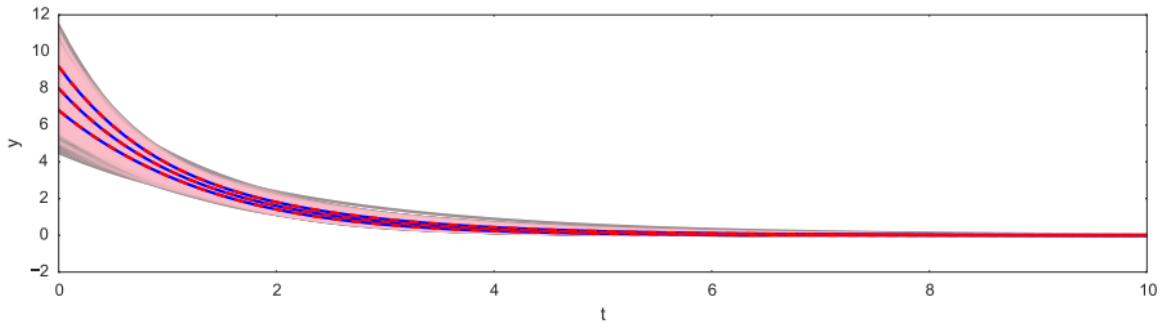
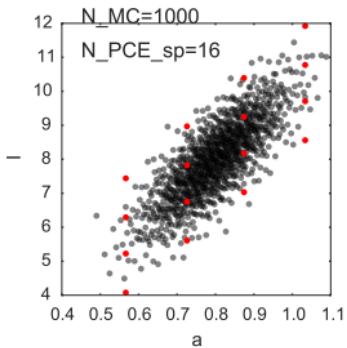
Case A2: A simple model

Inputs: 2D Correlated Normal

$$y = I e^{-a t}$$

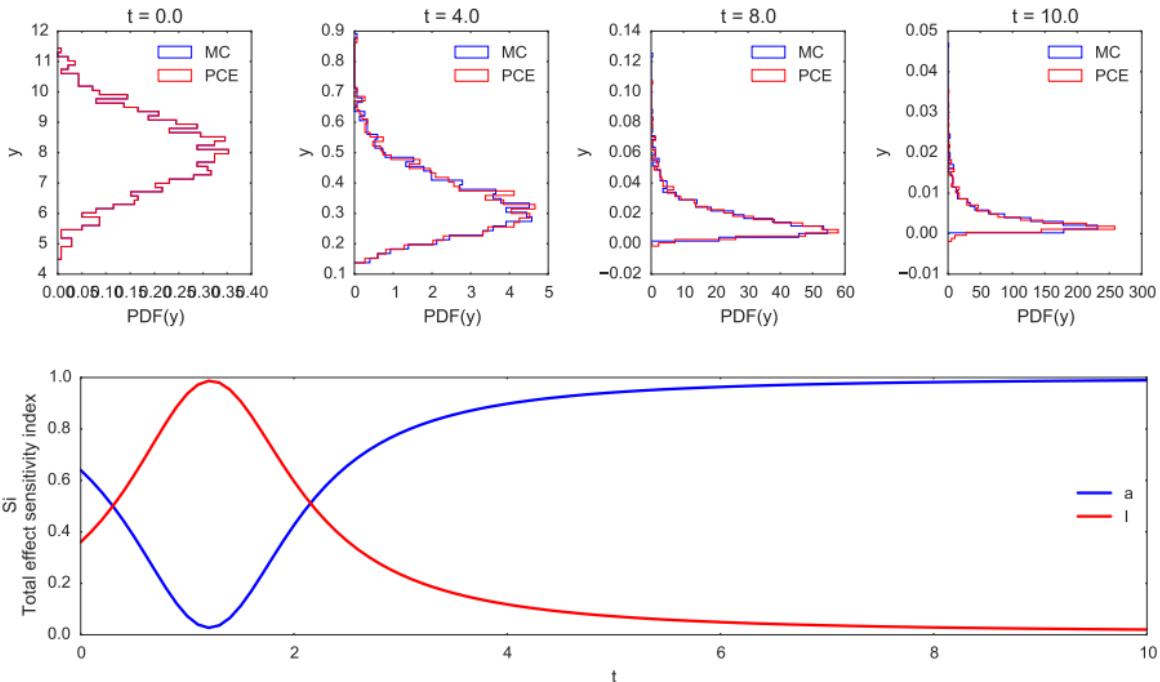
$$\begin{bmatrix} a \\ I \end{bmatrix} \sim \text{Normal}(\mu, \mathbb{C})$$

$$\mu = \begin{bmatrix} 0.8 \\ 8. \end{bmatrix} \quad \mathbb{C} = \begin{bmatrix} 0.01 & 0.1 \\ 0.1 & 2. \end{bmatrix}$$



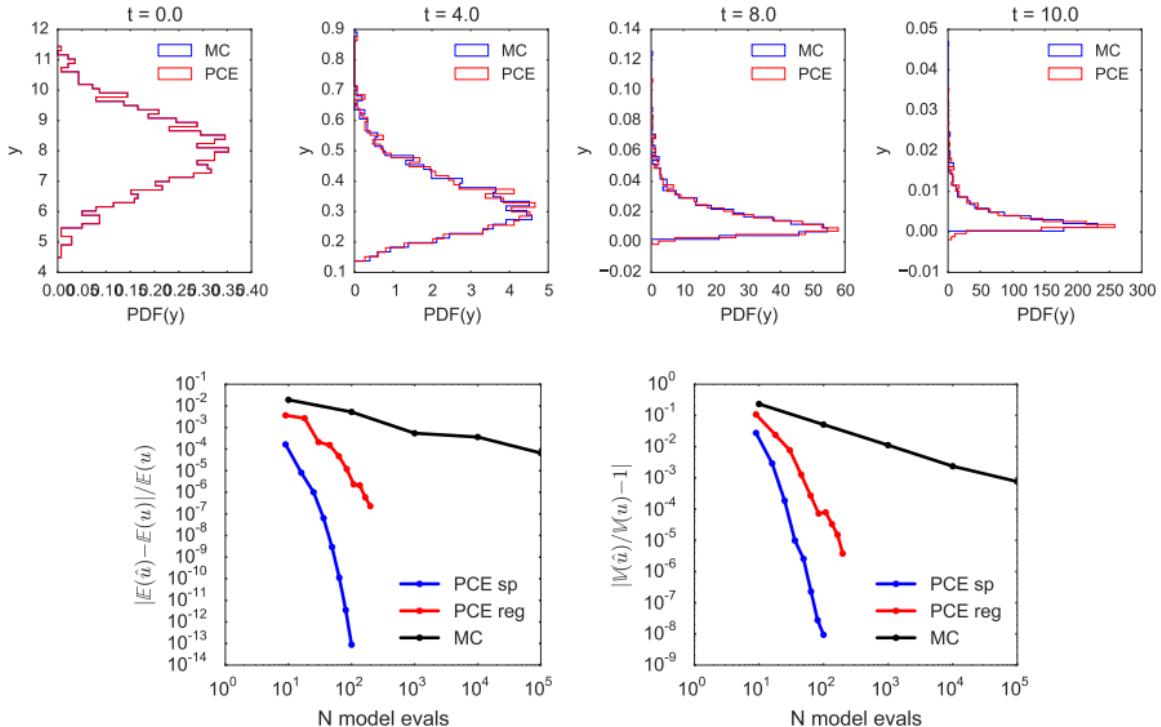
Case A2: A simple model

Inputs: 2D Correlated Normal



Case A2: A simple model

Inputs: 2D Correlated Normal



Outline

- ① (A very short) Introduction to Polynomial Chaos Expansions
- ② A1. 2D Not correlated Normal
- ③ A2. 2D MvNormal
- ④ A3. 2D Conditionally Correlated
- ⑤ A4. 2D Copula Correlated
- ⑥ Conclusions

Case A3: A simple model

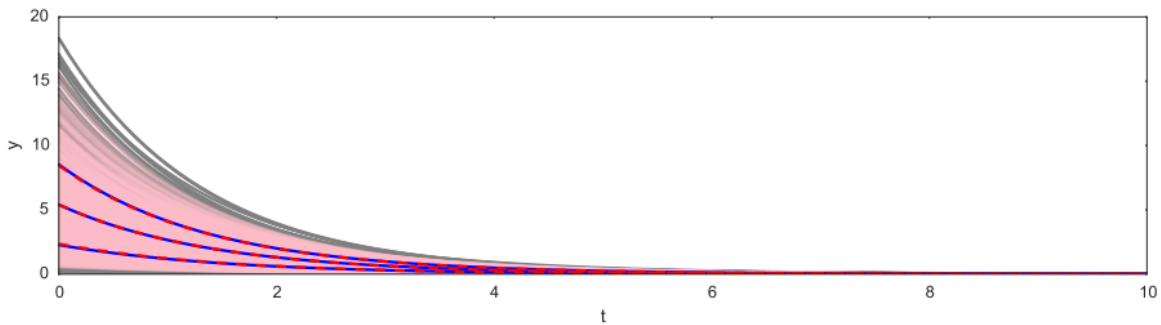
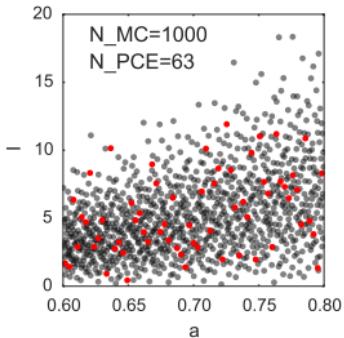
Inputs: 2D Conditionally Correlated

$$y = I e^{-a t}$$

$$a \sim Uniform(0.6, 0.8)$$

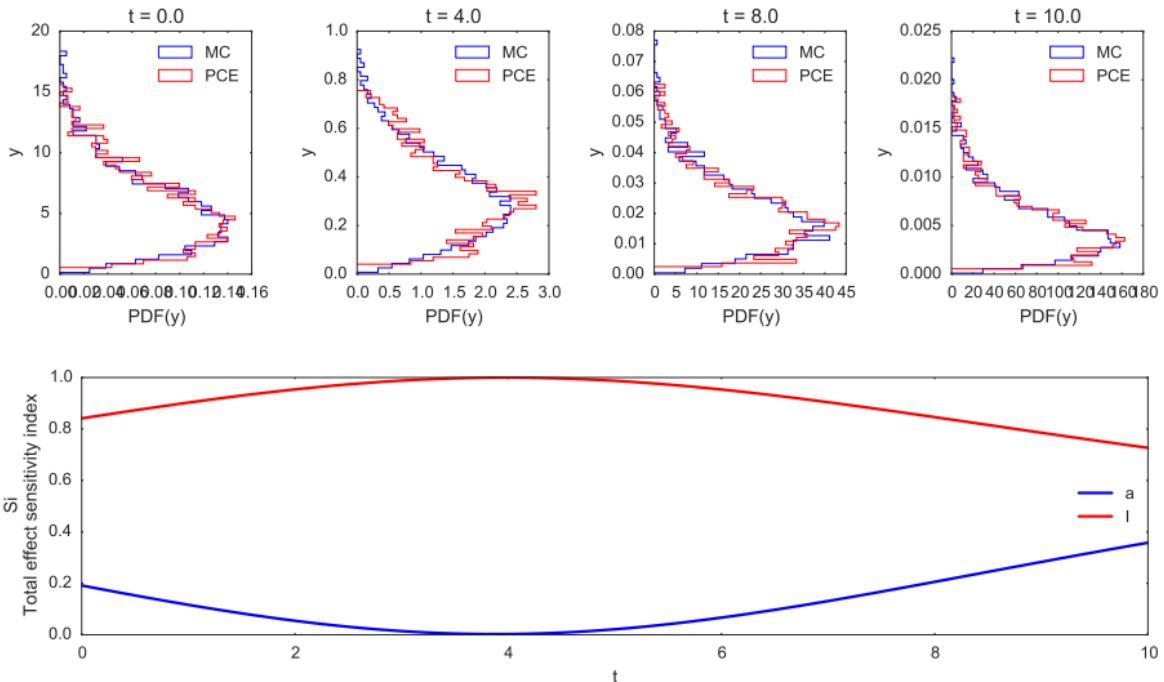
$$I \sim Weibull(k = 2.,$$

$$A = 6(0.3 + a)^4)$$



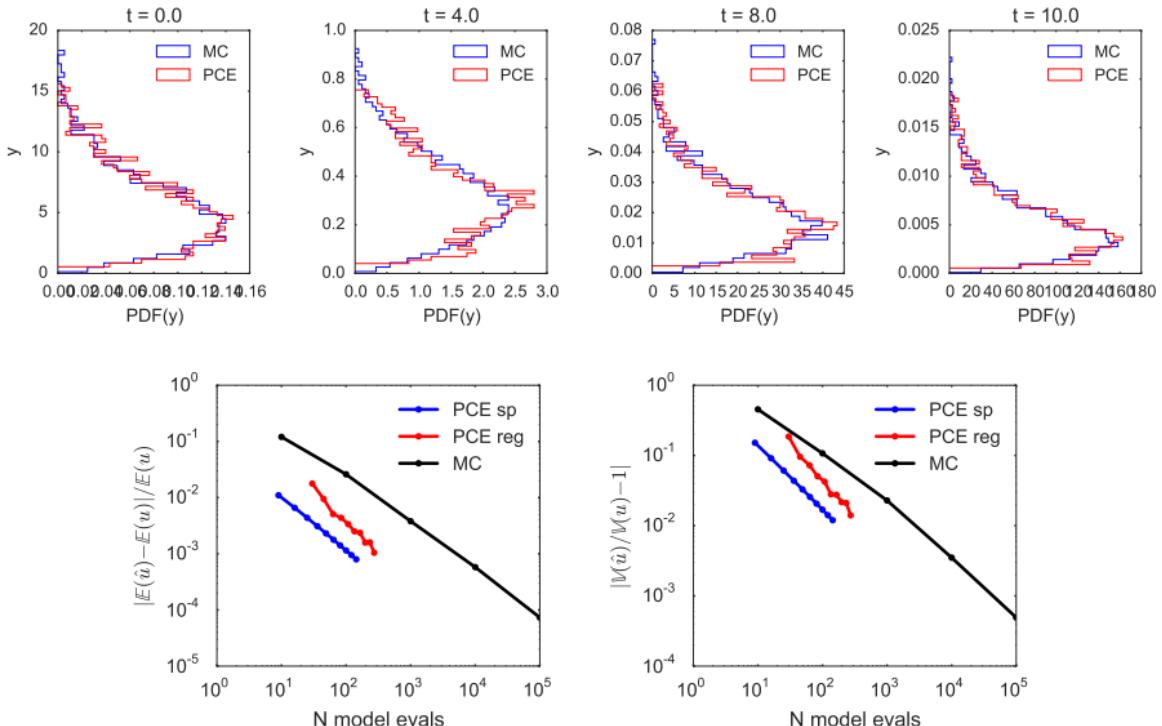
Case A3: A simple model

Inputs: 2D Conditionally Correlated



Case A3: A simple model

Inputs: 2D Conditionally Correlated



Options: A better surrogate and MC



Case A3: A simple model

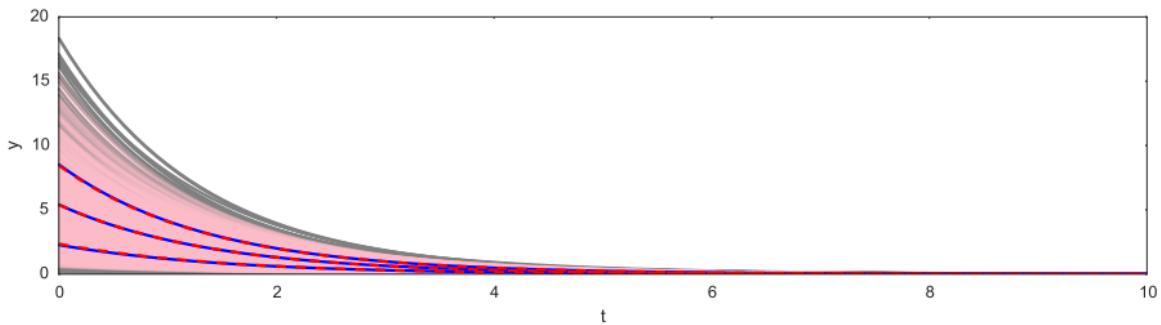
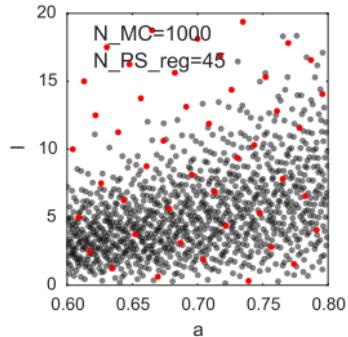
Inputs: 2D Conditionally Correlated - A better surrogate and MC

$$y = I e^{-a t}$$

$$a \sim Uniform(0.6, 0.8)$$

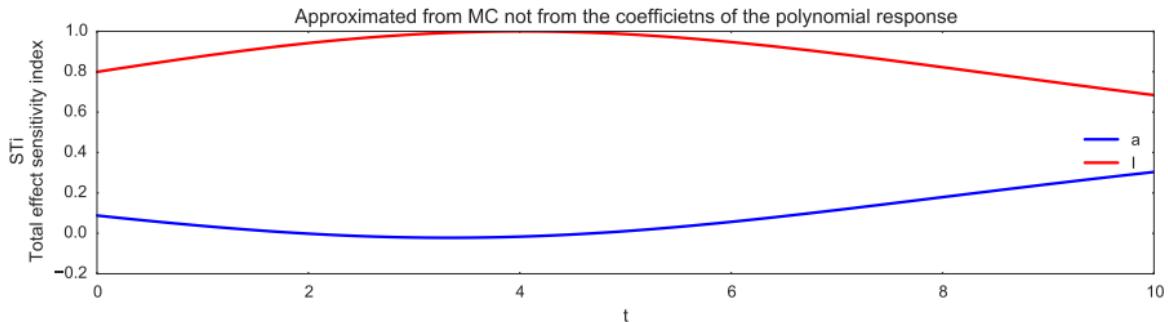
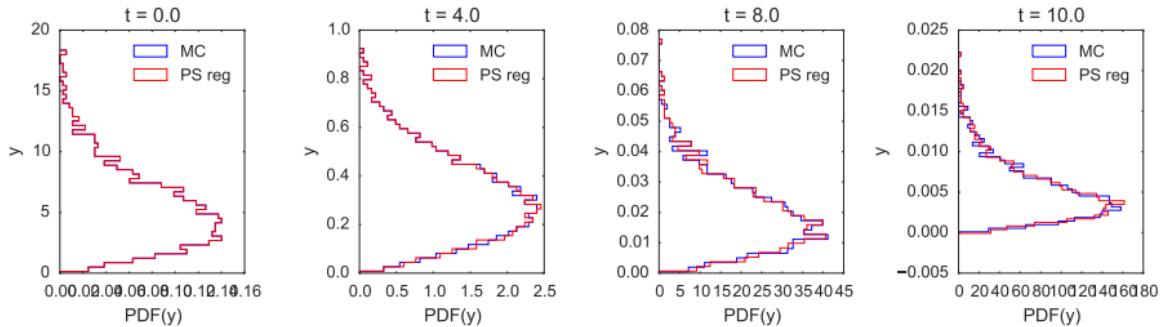
$$I \sim Weibull(k = 2.,$$

$$A = 6(0.3 + a)^4)$$



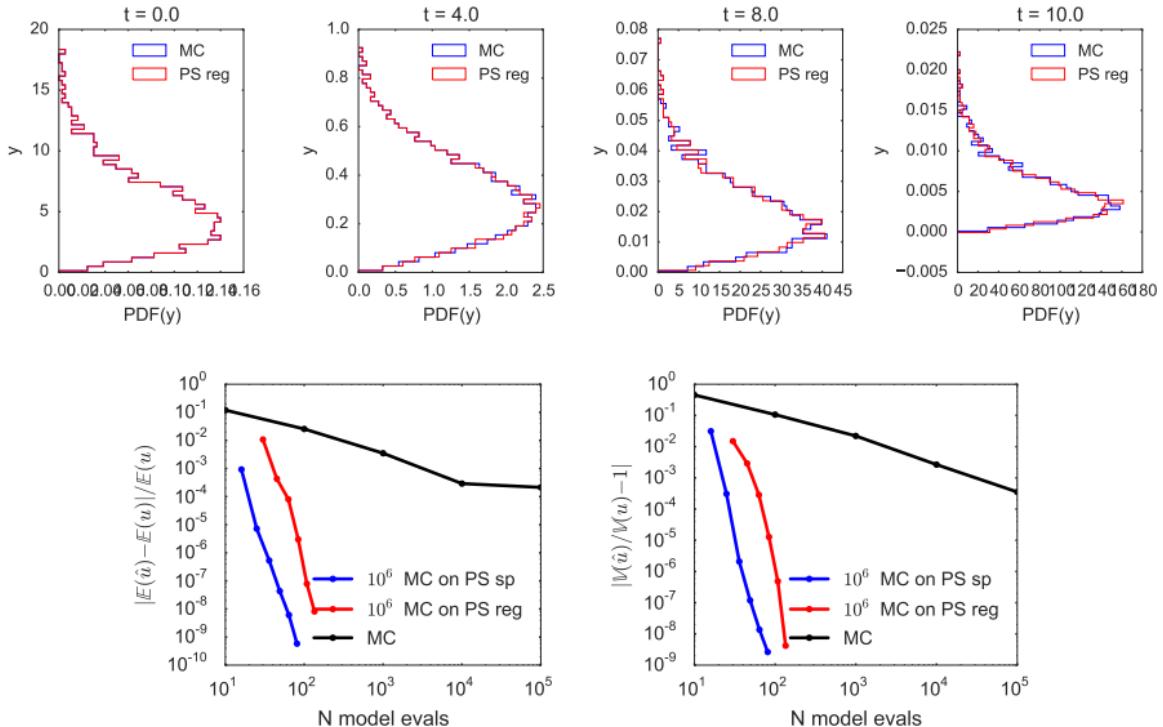
Case A3: A simple model

Inputs: 2D Conditionally Correlated - A better surrogate and MC



Case A3: A simple model

Inputs: 2D Conditionally Correlated - A better surrogate and MC



Outline

- ① (A very short) Introduction to Polynomial Chaos Expansions
- ② A1. 2D Not correlated Normal
- ③ A2. 2D MvNormal
- ④ A3. 2D Conditionally Correlated
- ⑤ A4. 2D Copula Correlated
- ⑥ Conclusions

Case A4: A simple model

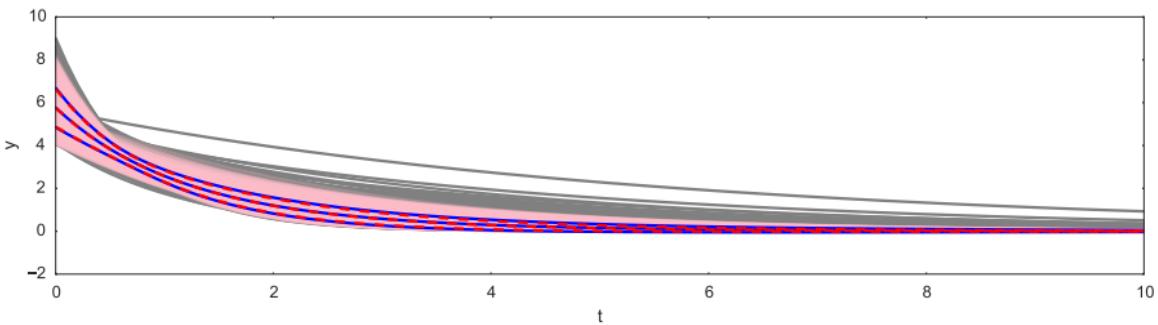
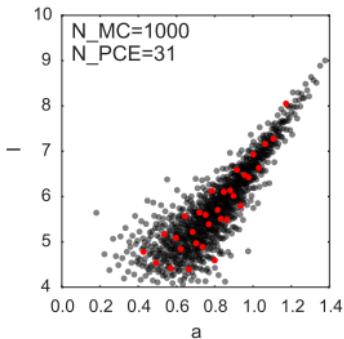
Inputs: 2D Copula Correlated

$$y = I e^{-a t}$$

$$a \sim Normal(\mu = .8, \sigma = .2)$$

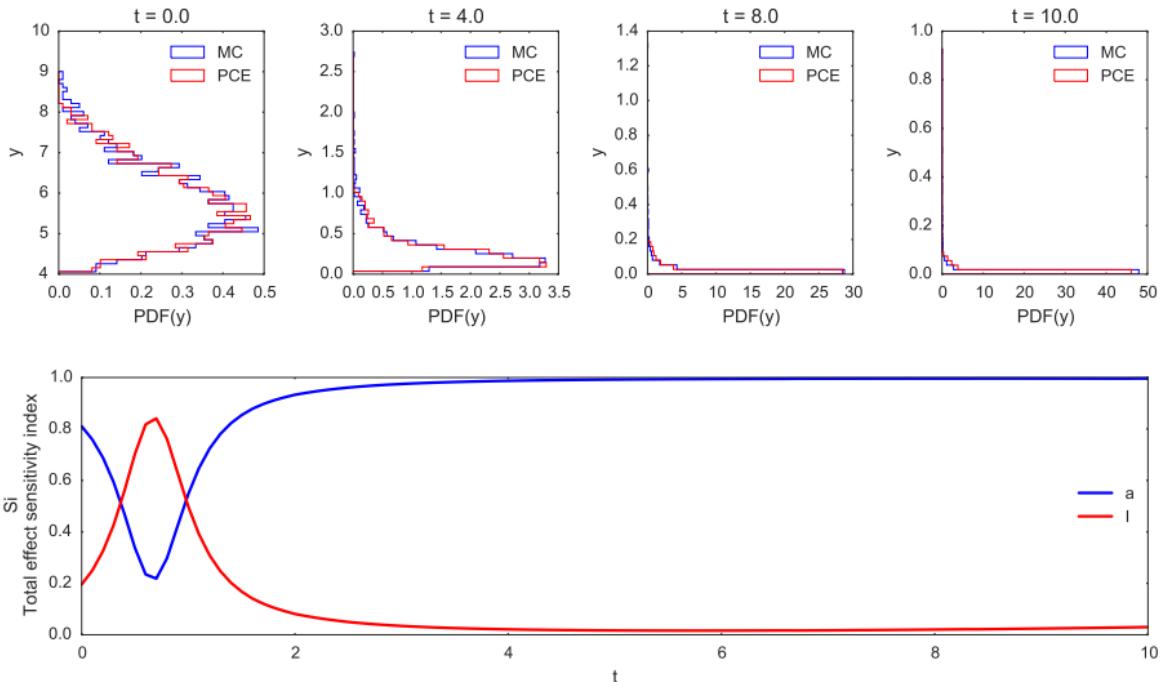
$$I \sim Weibull(k = 2., A = 2)$$

$$Joe(CDF(a), CDF(I), theta = 5.)$$



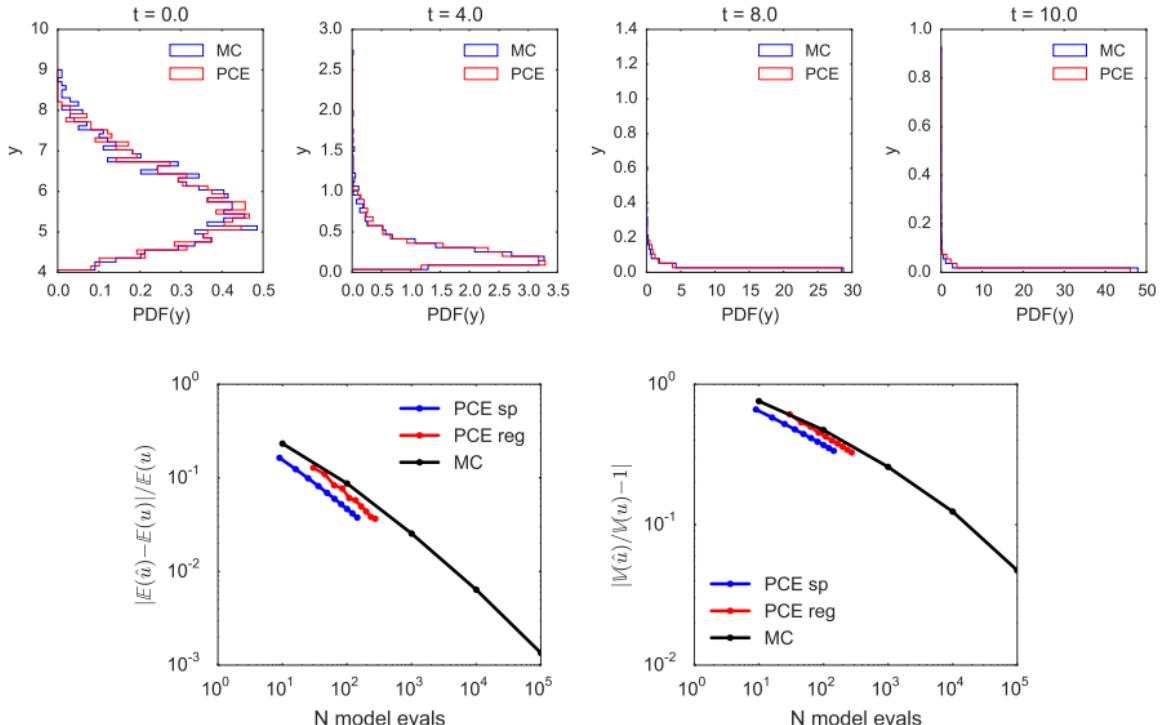
Case A4: A simple model

Inputs: 2D Copula Correlated



Case A4: A simple model

Inputs: 2D Copula Correlated



Options: A better surrogate and MC



Case A4: A simple model

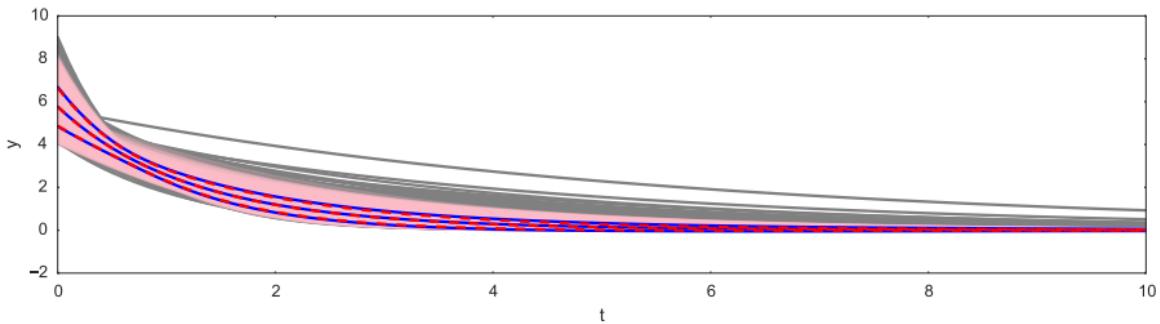
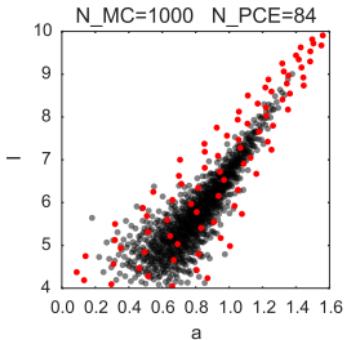
Inputs: 2D Copula Correlated - A better surrogate and MC

$$y = I e^{-a t}$$

$$a \sim Normal(\mu = .8, \sigma = .2)$$

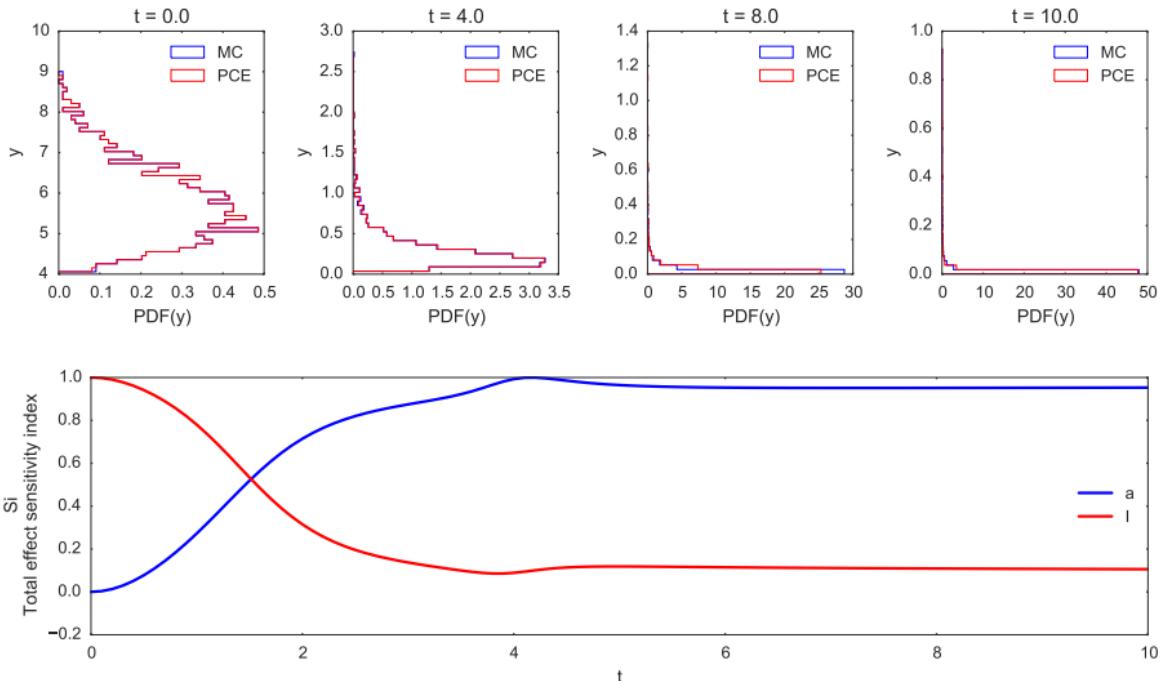
$$I \sim Weibull(k = 2., A = 2)$$

$$Joe(CDF(a), CDF(I), theta = 5.)$$



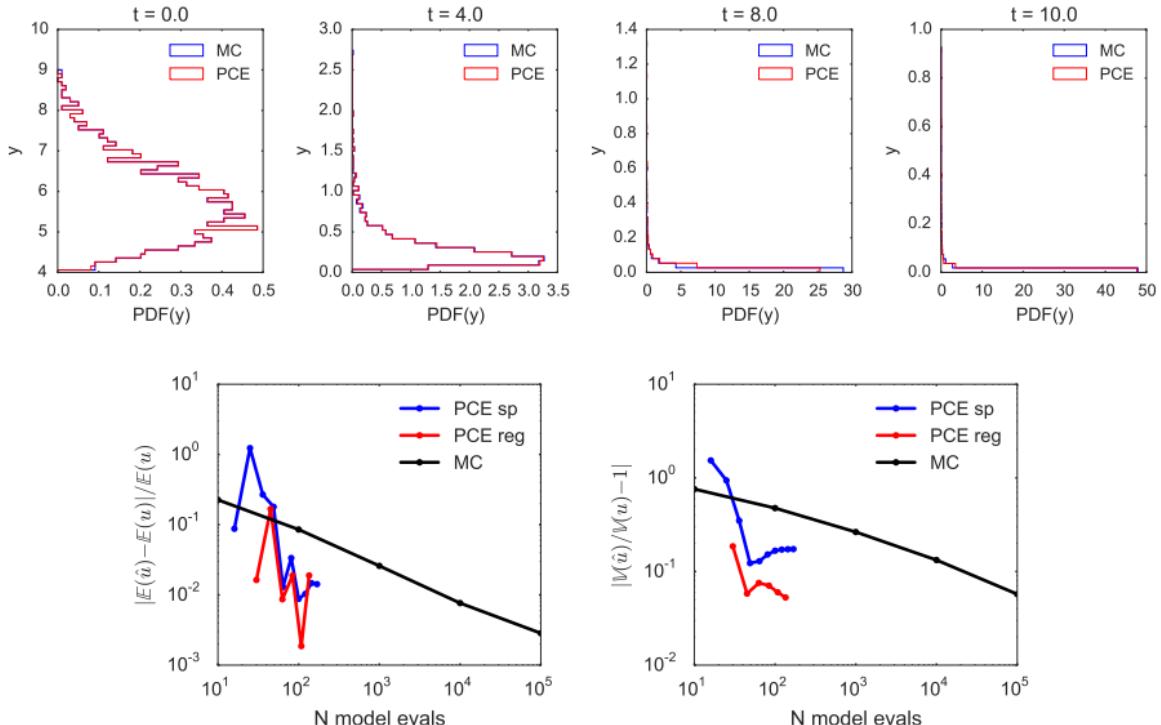
Case A4: A simple model

Inputs: 2D Copula Correlated - A better surrogate and MC



Case A4: A simple model

Inputs: 2D Copula Correlated - A better surrogate and MC



Outline

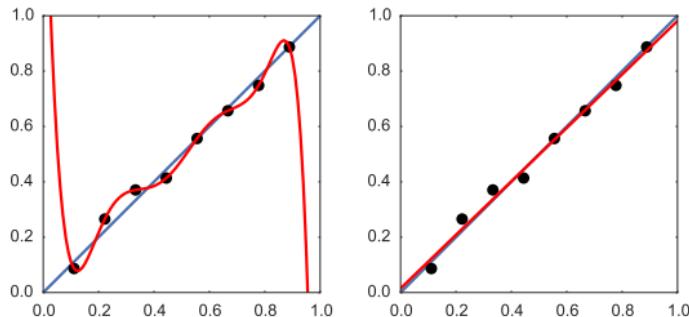
- ① (A very short) Introduction to Polynomial Chaos Expansions
- ② A1. 2D Not correlated Normal
- ③ A2. 2D MvNormal
- ④ A3. 2D Conditionally Correlated
- ⑤ A4. 2D Copula Correlated
- ⑥ Conclusions

How to avoid over fitting and achieve sparsity?

Sparse linear model regression

- Least Absolute Shrinkage and Selection Operator problem (LASSO) is useful to avoid over-fitting and achieve sparsity in the PCE.
- LASSO is a least squares minimization problem with a l_1 penalization on the coefficients (\mathbf{c}):

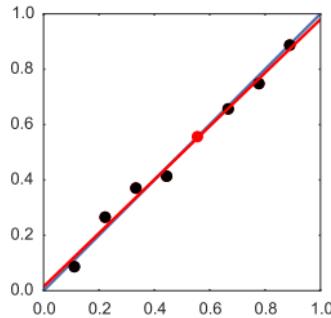
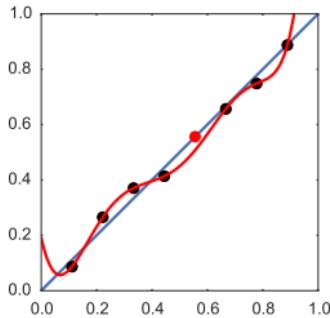
$$\min_{\mathbf{c}} \|\mathbf{c}\Psi - \mathbf{y}_k\|_2^2 + \alpha_k \|\mathbf{c}\|_1 = \min_{\mathbf{c}} \sum_{i=0}^{N-1} \left[\sum_{l=0}^{N_c-1} c_l \Psi_l(\mathbf{w}_i) - y_k(\mathbf{x}_i) \right]^2 + \alpha_k \sum_{l=0}^{N_c-1} |c_l|$$



How to select the right sparsity?

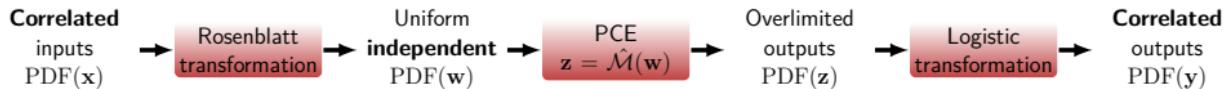
k-fold cross validation

- It divides the dataset in k groups and uses $k-1$ groups ("folds") for training and the remaining for validation. Repeat this process until all the groups have been the validation set.
- k-fold cross validation is repeated for multiple values of the sparsity parameter.
- As a result it gives the optimal sparsity parameter (α_k).



PCE of complex cases

Variable transformations steps



Rosenblatt transformation

- Used to decorrelate the variables
- Variables can be transformed to independent Uniform or Normal
- Inverse transformation used for MC sample. Use efficient sampling techniques in the unitary uniform uncorrelated space.

PCE model surrogate

- Polynomial chaos expansion working on the uncorrelated space.
- Trained using k-Fold validation to avoid over-fitting and prefer lower order polynomials (Least absolute shrinkage and selection operator - LASSO problem).

Logistic transformation

- Used to force fixed constraints in the outputs: i.e. to avoid overshoots.
- Can be used to smooth discontinuities and to impose only positive values.

References I

[Rosenblatt 1952] Rosenblatt, M 1952 Remarks on a multivariate transformation. Annals of Mathematical Statistics Vol. 23, pp 470-472.]

[Feinberg 2015] Feinberg, J., & Langtangen, H. P. (2015). Chaospy: An open source tool for designing methods of uncertainty quantification. Journal of Computational Science, 11, 46-57.

Questions?

J. P. Murcia
+45 2339 7790
jumu@dtu.dk
PhD Student
DTU Wind Energy

Technical University of Denmark (DTU)
Building 101
Risø Campus
Frederiksborgvej 399
4000 Roskilde, Denmark