Uncertainty Quantification Mini Project: Surrogate Model Construction for Black-Scholes Model using PCE

May 2025

Contents

1 Introduction

A polynomial chaos expansion (PCE) is a method to surrogate a computationally expensive model, such as Finite Element Analysis. The surrogate model can be expressed as the sum of orthonormal multivariate polynomials with respect to the probability density function (PDF) of the input space. For example, given a model $\mathcal{Y} = \mathcal{M}(\mathbf{X})$, the PCE is expressed as $\mathcal{Y} \approx \mathcal{M}_{PCE}(\mathbf{X}) = \sum_{\alpha \in \mathcal{A}} c_\alpha \Psi_\alpha(\mathbf{X})$ where $\mathcal{A}$ is a truncated set of orthonormal polynomials. The surrogate model can then be used to conduct Monte Carlo simulations as well as sensitivity analysis.

2 Selection of Computational Model

We choose to build a surrogate model for the Black-Scholes Model for option pricing, which is mathematically defined as

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

Here, $V(S,t)$ is the option price, S is the price of the underlying asset, t is time, σ is the volatility (stddev of asset's returns), r is the risk-free interest rate. We consider t time to be in *years*, price in \$, but one can assume it at any scale.

Given the assumptions of:

1. Option pricing is zero for a zero valued asset.
2. If the asset price is high enough, call option price would just be price of asset minus the cost to buy the stock
3. Just before maturity, call option price would be $max\{S - K, 0\}$

The Black-Scholes Partial Differential Equation analytically becomes:

$$\begin{aligned} P_{call}(S,t) &= \mathcal{N}(d_+)S - \mathcal{N}(d_-)Ke^{-r(T-t)} \\ d_+ &= \frac{1}{\sigma\sqrt{T-t}}\left[ln(\frac{S}{K}) + (r + \frac{\sigma^2}{2})(T-t)\right] \\ d_- &= d_+ - \sigma\sqrt{T-t} \end{aligned} \tag{1}$$

Here $\mathcal{N}(v)$ denotes the probability density function of a unit normal at v.

2.1 The probabilistic input model

For a concrete example, we take the instance of AAPL stock and its options. We observe on the date 05.05.2025, AAPL stock base price had a mean of 196 and an intra-day standard deviation of 0.6, therefore $S \sim \mathcal{N}(196, 0.6)$. We consider the prediction date to be the same date, therefore $t = 0$ as a fixed value. Looking at options in two years range, and considering possibly multiple maturity dates around that time, we take $T \sim \mathcal{U}(2, 2.25)$, signifying maturity dates being 730 to 820 days later. Furthur, the strike prices observed on various options lay between $K \sim \mathcal{U}(180, 215)$. Looking at year-long volatility of AAPL, that lay between $\sigma \sim \mathcal{U}(0.2, 0.8)$. Finally the risk-free interest rate was assumed from year long variance of 1.7% around 4.25%, giving $r \sim \mathcal{N}(0.0427, 0.03)$.

2.2 Sampling the experimental design

To construct the surrogate model, the coefficients of the PCE must be known. The projection of the analytical model to the orthonormal basic functions can be used to obtain the estimates of the coefficients. However, this method would require a numerical quadrature with $(p+1)^M$sampling points for a total degree basis of p and dimension M. A more efficient method is the least squares regression method that gives coefficients which minimize the square residual. This method is more computationally efficient, since sample sizes as small as $2\binom{M+p}{p}$ will give accurate estimates.

Here, since we have chosen the least squares regression model, the sampling points can be selected independently of the marginal distributions. The Latin Hypercube Sampling method was utilized.

2.3 Calculation of the polynomial coefficients with Ordinary Least Squares (OLS)

The outline of the algorithm is shown in Algorithm 1.

Algorithm 1 Obtaining Polynomial Chaos Expansion (PCE) using Ordinary Least Squares (OLS)

Require:
- $\mathcal{M}(\mathbf{X})$: computational model
- M: input dimension
- p: maximum total polynomial degree
- n_{exp}: number of model evaluations (experimental design size)
- $x_i \sim \mathcal{U}([a_i, b_i])$ or $x_i \sim \mathcal{N}(\mu_i, \sigma_i)$: marginal distribution of each input variable

1: **Generate** Latin Hypercube samples $\mathbf{u}_{\text{LHS}} \sim \mathcal{U}([0,1]^M)$ of size n_{exp}
2: **Transform to physical space:** $\mathbf{X}_{\text{exp}} = F_X^{-1}(\mathbf{u}_{\text{LHS}})$ using the inverse CDF of each marginal distribution
3: **Isoprobabilistic transform of inputs:** $\mathbf{X}_{\text{trans}} = \tau(\mathbf{u}_{\text{LHS}})$ where:
- τ maps each dimension in the hypercube $\mathcal{U}([0,1]^M)$ to $\mathcal{U}([-1,1])$ for uniform distributions and $\mathcal{N}(0,1)$ for Gaussian distributions. In both cases, the inverse CDF of the respective distribution is applied to the hypercube sampling space.

4: **Evaluate the model:** $\mathbf{Y}_{\text{exp}} = \mathcal{M}(\mathbf{X}_{\text{exp}})$
5: **Construct** univariate orthonormal polynomial basis $\{\pi_k(x)\}$ where:
- for uniform distributions, $\pi_k(x)$ are Legendre polynomials $P_n(x)$ and for Gaussian distributions, $\pi_k(x)$ are Hermite polynomials $H_{en}(x)$.

6: **Build multivariate basis:** $\Psi_{\boldsymbol{\alpha}}(\mathbf{x}) = \prod_{i=1}^{M} \pi_{\alpha_i}(x_i)$
7: **Apply truncation:** retain only terms where $|\boldsymbol{\alpha}| = \sum_{i=1}^{M} \alpha_i \leq p$
8: **Assemble design matrix:** $\boldsymbol{\Psi} \in \mathbf{R}^{P \times n_{\text{exp}}}$ with entries $\boldsymbol{\Psi}_{ij} = \Psi_i(\mathbf{X}_{\text{trans}}^{(j)})$
- P: total number of basis terms, given by $P = \binom{M+p}{p}$

9: **Compute coefficients:** $\mathbf{c} = (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\mathbf{Y}_{\text{exp}}$ (OLS solution)
10: **Construct surrogate:** $\hat{\mathcal{M}}_{\text{PCE}}(\mathbf{X}) = \sum_{i=1}^{P} c_i \Psi_i(\mathbf{X}_{\text{trans}})$

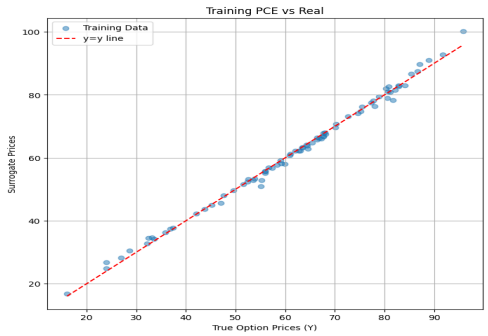
3 Numerical Experiments

3.1 Y-Y plots

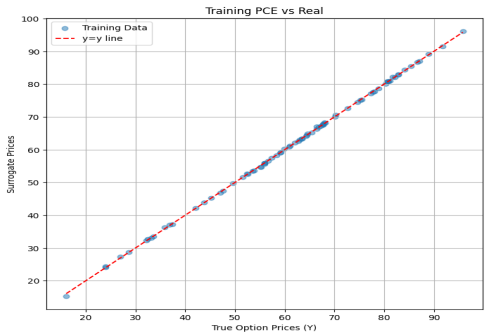
The Y-Y plot compares the output of the PCE model and the original model evaluated at the experimental design points. The experimental design of size $n_{exp} = 80$ was used to compare the Y-Y plots for PCE degree $p = 1$ and $p = 2$ (Figure 1).

The figures show that even a PCE degree p=1 will result in a model that accurately surrogates the true outputs. A PCE with degree p=2 results in a slightly more accurate surrogate model. This can be qualitatively evaluated using the polynomial coefficients and the Sobol indices.

Table 1 shows all the coefficients equal to or less than PCE degree 2. The largest coefficients are observed for polynomials with degree 1 in a certain variable and 0 for other variables. The first-order Sobol indices S_i and the total Sobol indices S_i^T are shown in Table 16 which also show that there are no significant interactions between the variables. Hence, the original Black Scholes model can be accurately surrogated using polynomials of degree 1, which is reflected in Figure 1.



(a) Y-Y plot for PCE degree 1



(b) Y-Y plot for PCE degree 2

Figure 1

S	T	K	σ	r	Coefficient
0	0	0	0	0	61.5788
0	0	0	0	1	4.6596
0	0	0	1	0	17.4078
0	0	1	0	0	-3.7668
0	1	0	0	0	1.0281
1	0	0	0	0	0.3989
0	0	0	0	2	0.2361
0	0	0	1	1	-0.6363
0	0	0	2	0	-0.2908
0	0	1	0	1	-0.1452
0	0	1	1	0	0.5933
0	0	2	0	0	0.1352
0	1	0	0	1	0.2294
0	1	0	1	0	0.2019
0	1	1	0	0	-0.0077
0	2	0	0	0	-0.0419
1	0	0	0	1	-0.0333
1	0	0	1	0	0.0940
1	0	1	0	0	-0.0490
1	1	0	0	0	-0.0406
2	0	0	0	0	0.0731

Table 1: Polynomial coefficients for each variable $[S, T, K, \sigma, r]$.

Case	**S**	**T**	**K**	$\boldsymbol{\sigma}$	**r**
S_i	$4.82 \cdot 10^{-4}$	$3.10 \cdot 10^{-3}$	0.0416	0.888	0.0638
S_i^T	$5.23 \cdot 10^{-4}$	$3.38 \cdot 10^{-3}$	0.0427	0.891	0.0652

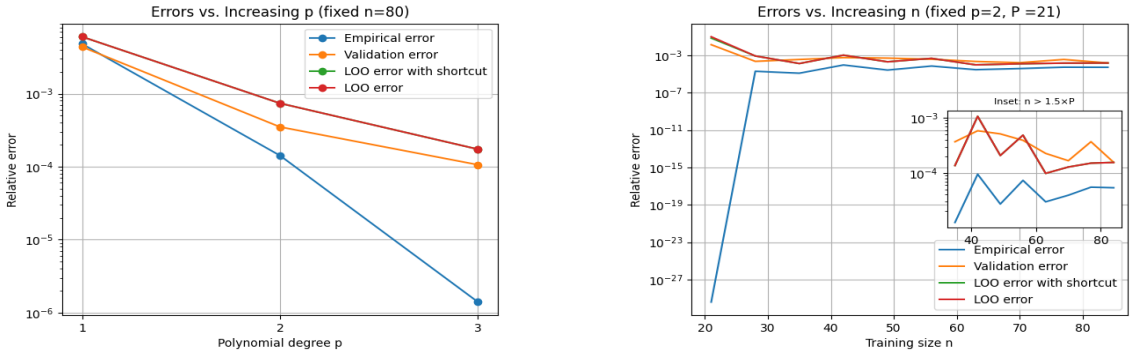
Table 2: Sobol Indices for the PCE model

3.2 Global Error

The global error of the PCE was calculated using the relative empirical error, the relative leave-one-out error, and the validation error. The empirical error uses the training set while the validation error samples a large validation set $\mathbf{X_{val}}, \mathcal{M}(\mathbf{X_{val}})$ and evaluates the PCE at the new data points. For the leave-one-out

error, one sample is left out of the experimental data set $\mathbf{X_{exp}}$ and a new PCE is calculated based on the $n_{exp} - 1$ data points. The error is then calculated at the data point that was excluded. Since this method is repeated for every single data point in $\mathbf{X_{exp}}$, a less computationally intensive alternative is implemented below.

The relative empirical error e_{emp}, the relative leave-one-out error e_{LOO}, and the relative validation error e_{val} are defined in Equation 2 to 9. $E_{\text{LOO,shortcut}}$ refers to an alternative to compute the leave-one-out error that does not reconstruct the PCE every time a data point is excluded. Otherwise, a PCE $\mathcal{M}_{\text{PCE}\setminus i}(x^i_{\text{exp}}))$must be constructed for each scenario in which the ith point is excluded. It should be noted that for the validation sample $\mathbf{X_{val}}$ with n_{val} sampling points , the Latin Hypercube Sampling is used to sample the data set and the isoprobabilistic transform shown in step 3 of Algorithm 1 is performed.



(a) Global Error analysis for fixed n

(b) Global Error analysis for fixed p

Figure 2

$$E_{\text{emp}} = \frac{1}{n_{\text{exp}}} \sum_{i=1}^{n_{\text{exp}}} \left(\mathcal{M}(x^i_{\text{exp}}) - \mathcal{M}_{\text{PCE}}(x^i_{\text{exp}})\right)^2 \tag{2}$$

$$e_{\text{emp}} = \frac{E_{\text{emp}}}{\text{Var}(\mathcal{Y})} \quad \text{where } \mathcal{Y} = \left\{\mathcal{M}(x^i_{\text{exp}}),\, i = 1, \ldots, n_{\text{exp}}\right\} \tag{3}$$

$$E_{\text{LOO}} = \frac{1}{n_{\text{exp}}} \sum_{i=1}^{n_{\text{exp}}} \left(\mathcal{M}(x^i_{\text{exp}}) - \mathcal{M}_{\text{PCE}\setminus i}(x^i_{\text{exp}})\right)^2 \tag{4}$$

$$E_{\text{LOO,shortcut}} = \frac{1}{n_{\text{exp}}} \sum_{i=1}^{n_{\text{exp}}} \left(\frac{\mathcal{M}(x^i_{\text{exp}}) - \mathcal{M}_{\text{PCE}}(x^i_{\text{exp}})}{1 - h_i}\right)^2 \tag{5}$$

$$h_i = \left(\mathbf{\Psi}^T(\mathbf{\Psi}\mathbf{\Psi}^T)^{-1}\mathbf{\Psi}\right)_{i,i} \tag{6}$$

$$e_{\text{LOO}} = \frac{E_{\text{LOO}} \text{ or } E_{\text{LOO,shortcut}}}{\text{Var}(\mathcal{Y})} \tag{7}$$

$$E_{\text{val}} = \frac{1}{n_{\text{val}}} \sum_{i=1}^{n_{\text{val}}} \left(\mathcal{M}(x^i_{\text{val}}) - \mathcal{M}_{\text{PCE}}(x^i_{\text{val}})\right)^2 \tag{8}$$

$$e_{\text{val}} = \frac{E_{\text{val}}}{\text{Var}(\mathcal{Y}_{val})} \quad \text{where } \mathcal{Y}_{val} = \left\{\mathcal{M}(x^i_{\text{val}}),\, i = 1, \ldots, n_{\text{val}}\right\} \tag{9}$$

Two numerical experiments were conducted to analyze the global error. First, the size of the experimental data n_{exp} was fixed at 80 and the degree of total polynomial truncation was increased from $p = 1$ until $P = \binom{M+p}{p} > n$ from where the OLS will begin to fit the experimental data. n_{exp} was chosen to be slightly

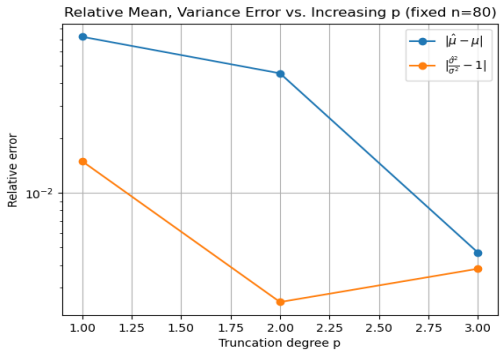
greater than $10M$ to allow 3 PCE degrees to be analyzed. All symbols are defined in Algorithm 1. Secondly, the degree of polynomial truncation p was kept at 2 and the size of the experimental design was increased from $n = P = \binom{M+p}{p}$ to $n = 4P = 4\binom{M+p}{p}$ with 10 increments. The validation error was calculated using 1000 samples for both experiments.

Figure 2(a) shows the result for the fixed n, increasing p experiment and Figure 2(b) shows the fixed p ,increasing n experiment. Capital $P = \binom{M+p}{p}$ in (b) refers to the constant number of multivariate polynomial functions used. For the former, all error values show a constant rate of error reduction when p increases, with the empirical error having a higher rate. In addition, the validation and LOO error are greater than the empirical error. This is because the empirical error is calculated at the points that have been used to construct the PCE.

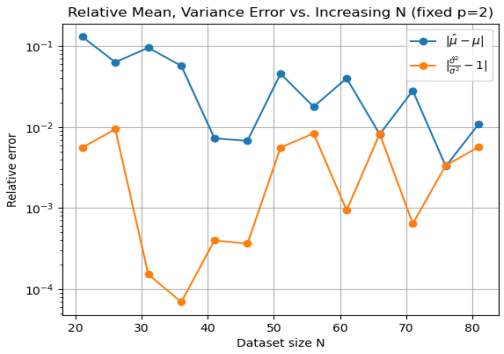
When increasing the size of the experimental data set, all error values will stabilize after n_{exp} is larger than 30 points. For the empirical error and the LOO error, the errors fluctuate since they are sensitive to the values of the experimental design data points. The validation error decreases slightly with increasing n_{exp}. For the smallest data set with $n_{exp} = P$, the empirical error will be close to 0 since the OLS solution will fit the given data. Overall, one can conclude that the PCE polynomial degree has a more effective effect in reducing the error since more basis functions can be used to capture the nonlinearity of the model and interaction between variables.

3.3 Relative error in mean and variance

An estimate of true mean and variance of the model output is found using Monte-Carlo sampling, with 10000 data-samples, and by law of large numbers, we treat these as the true values. The output of model predictions have their own mean and variance, which are compared to the true values using $|\hat{\mu} - \mu|$ and $|\frac{\hat{\sigma}^2}{\sigma^2} - 1|$.



(a) For fixed $N = 1000$, $P \leq N$



(b) For $p = 3$, $P \leq N \leq 4P$

Figure 3

We calculate model mean as c_0 and model variance as $\sum_{i=1}^{\infty} c_i$.

1. Increasing p until $P \leq N = 80$ shows both mean and variance relative errors decrease with the increase of model complexity. The increase in variance at $p = 3$ can be attributed to the drastic change in the constant order term, which would be traded off by the rest of the terms that deviates the model variance from true variance.

2. Increasing N until $4P$, $p = 2$ shows plots of variance and mean relative errors. The output is quite oscillating and random with a general decreasing trend in error as datapoints increase. The oscillating nature can be attributed to randomness of data, giving different mean and variance with every iteration, and also the fact that multiple dimensions of inputs interact to give a single output value.

4 Comparison to UQ[py]Lab

4.1 Model Validation

Outputs of the PCE model were compared to UQ[py]Lab using a validation set and with model coefficients. The UQ[py]Lab model has the same parameters as the custom implementation, however the constant variable t is modeled as a Gaussian distribution with no variance.

The validation set consists of 10000 samples selected with the built-in LHS method from UQ[py]Lab. A modified version of the model which accepts samples generated by UQ[py]Lab was used to compare PCE outputs with a Y-Y plot:

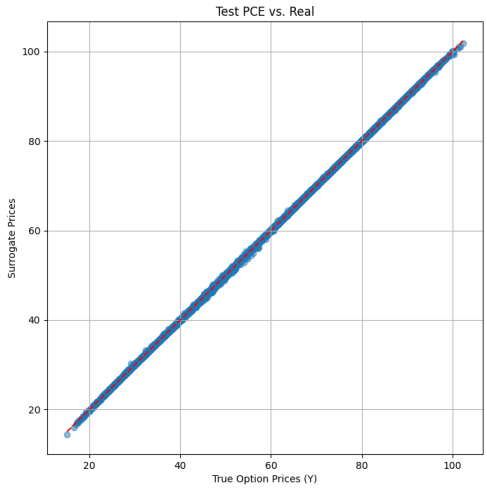


Figure 4: Custom implementation Y-Y plot compared to original model

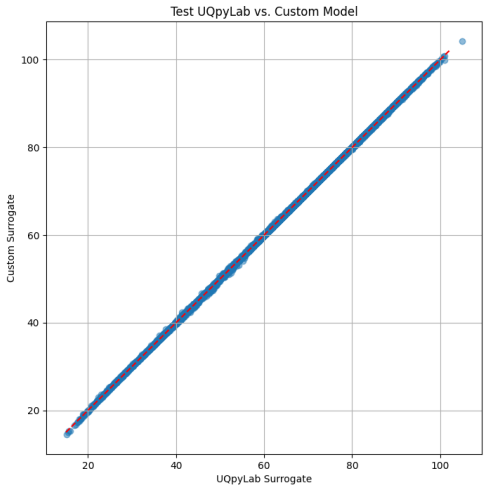


Figure 5: Comparison of custom implementation and UQ[py]Lab PCE outputs with 10000 samples

Metric	**Value**
E_{LOO}	$2.1008 \cdot 10^{-}5$
Normalized Empirical Error	$2.0766 \cdot 10^{-}6$

Table 3: Model accuracy metrics

Based on the outputs, the following metrics were obtained:

Metric	**Value**
RMSE	26.2163
Pearson R	0.9999
R^2	0.9999

Table 4: Agreement metrics for custom implementation and UQ[py]Lab PCE outputs with 10000 samples

Since UQ[py]Lab and the custom implementation follow a different order for calculating the indices and coefficients, these were reordered. Absolute and relative L_2 error based on the coefficient matrices were

calculated using:

$$\begin{aligned}
\Delta\mathbf{a} &= \mathbf{a}^{(\text{PCEModel})} - \mathbf{a}^{(\text{UQpyLab})}, \\
\|\Delta\mathbf{a}\|_2 &= \left\|\mathbf{a}^{(\text{PCEModel})} - \mathbf{a}^{(\text{UQpyLab})}\right\|_2, \\
\varepsilon_{L_2} &= \frac{\|\Delta\mathbf{a}\|_2}{\left\|\mathbf{a}^{(\text{UQpyLab})}\right\|_2}, \\
\varepsilon_{L_\infty} &= \|\Delta\mathbf{a}\|_\infty
\end{aligned}$$

The following results were obtained based on 56 coefficients:

Metric	**Value**
Coefficients L_2	0.1275
Relative L_2	0.0020
L_∞ error	0.0377
Pearson R	0.9999

Table 5: Comparison of coefficients for custom implementation and UQ[py]Lab PCE models

Coefficient output graph can be found in the appendix.

4.2 Sparse Solver Results

UQ[py]Lab offers alternative solvers for fitting. While these are more helpful for models with larger degrees, they could offer faster results with this. These were compared to the existing models, with the following results:

Metric	**Custom**	**OLS**	**OMP**	**LARS**
$E_{LOO}[\cdot 10^-5]$	2.0038	2.1008	1.1783	1.5375
Norm.Emp.Err.$[\cdot 10^-6]$	2.2420	2.0766	3.8197	3.04877
RMSE vs. Own Code	-	26.2163	26.2152	26.2195

Table 6: Performance metrics for different solvers

4.3 Sensitivity Analysis

Using various methods available in UQ[py]Lab, sensitivity analysis with the following methods were performed:

Input/Output Correlation

Correlation analysis was performed using the builtin correlation function with 10000 samples. It can be observed that volatility, (σ) has the highest influence on the result, followed by the strike price Ki which tends to reduce the option price. Since the difference between the two methods are not high, we can confirm monotonic behavior.

Method	S	T	K	σ	r
Correlation-Based Indices	0.017922	0.056947	-0.204786	0.942672	0.272935
Rank Correlation Based Indices	0.018454	0.054771	-0.194313	0.947558	0.252901
Difference Rank/Correlation	2.9684%	-3.9729%	5.3897%	0.5183%	-7.9217%

Table 7: Correlation coefficients for each variable $[S, T, K, \sigma, r]$.

Standard Regression Coefficients

Standard regression coefficients do not show a significant divergence from input/output correlation.

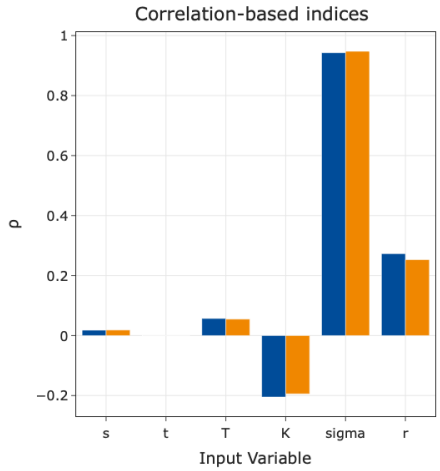


Figure 6: Comparison of correlation and rank-based correlation indices

Method	S	T	K	σ	r
Regression Sensitivity Indices	0.021581	0.056806	-0.202522	0.936164	0.252639
Correlation Based Indices	0.020976	0.054390	-0.191539	0.941566	0.231751
Difference Rank/Regression	-2.8842%	-4.4420%	5.7341%	0.5770%	-9.0131%

Table 8: Correlation coefficients for each variable $[S, T, K, \sigma, r]$.

Perturbation Method

With the perturbation method using centered calculation, significant differences for variables S, T, K and r can be observed which may indicate interaction effects or nonlinearity.

Method	S	T	K	σ	r
Perturbation-Based Indices	0.000483	0.003421	0.039265	0.895884	0.060946

Table 9: Correlation coefficients for each variable $[S, T, K, \sigma, r]$.

Sobol' Indices

Using the Sobol' indices, some interaction between the variables, specifically with σ can be observed, with close results to the perturbation method. Monte Carlo sampling amplifies the effect of σ and T.

Variables	S	T	K	σ	r
Total Sobol' Indices	0.000517	0.003537	0.043164	0.889546	0.066533
First Order Indices	0.000506	0.003334	0.041926	0.886396	0.064593
Variables	$\sigma \cdot r$	$\sigma \cdot K$	$\sigma \cdot T$	$T \cdot R$	$K \cdot R$
Second Order Indices	0.001800	0.001150	0.000149	0.000045	0.000043

Table 10: Sobol' indices with LHS sampling $[S, T, K, \sigma, r]$.

Variables	S	T	K	σ	r
Total Sobol' Indices	0.000510	0.003562	0.042583	0.890882	0.065638
First Order Indices	0.000500	0.003343	0.041385	0.887848	0.063783
Variables	$\sigma \cdot r$	$\sigma \cdot K$	$\sigma \cdot T$	$T \cdot R$	$K \cdot R$
Second Order Indices	0.001721	0.001119	0.000158	0.000052	0.000046

Table 11: Sobol' indices with MC sampling $[S, T, K, \sigma, r]$.

Cotter Measure and Morris Method

With the Cotter and Morris methods, local sensitivity and nonlinearities can be accommodated in the analysis. Using these methods with UQ[py]Lab, the following results were obtained with again a high degree of impact and nonlinearity for σ.

Variables	S	t	T	K	σ	r
Cotter Indices	0.560572	0.019199	2.879097	7.495058	31.430818	6.578880

Table 12: Cotter indices for each variable $[S, T, K, \sigma, r]$.

Variables	S	T	K	σ	r
μ	0.829457	3.679463	-13.219384	60.155595	9.455822
μ^*	0.829457	3.679463	13.219384	60.155595	9.455822
σ	0.096306	0.977161	2.780883	3.489644	1.951322

Table 13: Morris mean, first-order influence and standard deviation for each variable $[S, T, K, \sigma, r]$.

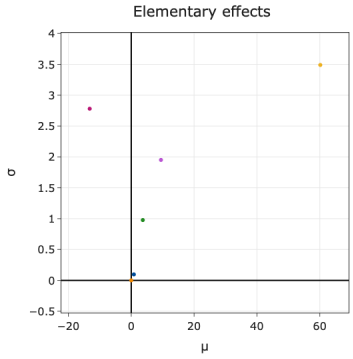


Figure 7: Morris sensitivity results (mean) for variables: S, T, t, K, σ, r

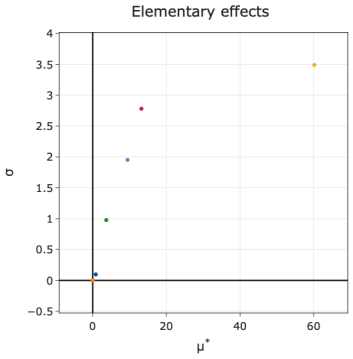


Figure 8: Morris sensitivity results (first order) for variables: S, T, t, K, σ, r

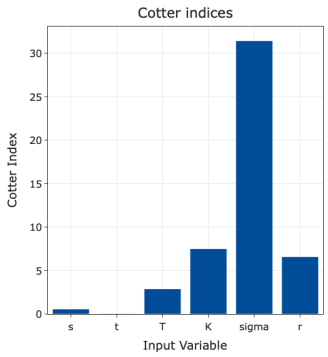


Figure 9: Cotter Indices

Borgonovo Indices

The density-based Borgonovo index can capture more non-monotonic behavior within the model. For the UQ[py]Lab implementation, it was assumed that t was a Gaussian distribution with $t \sim \mathcal{N}(0, 10^{-5})$ since zero results were not accepted. This method captures higher interaction behavior for r and K compared to others.

Variables	S	t	T	K	σ	r
Cotter Indices	0.054709	0.051809	0.066082	0.081381	0.612053	0.101044

Table 14: Correlation coefficients for each variable $[S, T, K, \sigma, r]$.

PCE-Based Sobol' Indices

PCE metamodels with the custom implementation and through UQ[py]Lab can be used to extract Sobol' indices. The models were compared to standalone Sobol' calculation and show more similarity to the MC-sampled results. Between the two models, high agreement can be observed.

Variables	S	T	K	σ	r
Total Sobol' Indices	0.000503	0.003507	0.042866	0.890033	0.066236
First Order Indices	0.000491	0.003303	0.041654	0.887016	0.064433
Variables	$\sigma \cdot r$	$\sigma \cdot K$	$\sigma \cdot T$	$T \cdot R$	$K \cdot R$
Second Order Indices	0.001678	0.001132	0.000153	0.000038	0.000034

Table 15: Sobol' indices for the UQ[py]Lab PCE model $[S, T, K, \sigma, r]$.

Variables	S	T	K	σ	r
Total Sobol' Indices	0.000523	0.00338	0.0427	0.891	0.0652
First Order Indices	0.000482	0.00310	0.0416	0.888	0.0638

Table 16: Sobol' indices for the custom PCE model

ANCOVA Indices

With analysis of covariance, sensitivity indices after adjusting for all other variables can be obtained. Results provided by UQ[py]Lab suggest low level of interaction between the variables, and similar total sensitivity to other methods.

Variables	S	T	K	σ	r
S: total	0	0.003	0.043	0.884	0.065
S^U: unique	0	0.003	0.041	0.880	0.063
S^I: interaction	0	0	0	0	0
S^C: correlated	0	0	0.002	0.004	0.002

Table 17: ANCOVA indices with MC sampling $[S, T, K, \sigma, r]$.

5 Conclusion

The Black Scholes model with five input variables (S: base price, T: maturity date, K: strike price, σ: volatility, r: risk-free interest rate) was surrogated using polynomial chaos expansion. Each variable was assumed to follow either a uniform or Gaussian distribution based on actual data from the AAPl stock. Sampling was conducted using the Lattin hypercube sampling method, and PCE coefficients were obtained using the ordinary least squares method. From the Y-Y plot, Sobol indices, analysis of global error and the relative error, we observed that the PCE accurately surrogates the Black Scholes model with a PCE polynomial degree of 1 and 30 sampling points. The Sobol indices and the ANCOVA indices calculated using the UQ[py]Lab analysis confirm this observation, showing that the Black-Scholes model can be explained by univariate terms with few influences of interaction terms.

6 Appendix

6.1 Contributions

1. Chin Zhe Tee: Implemented the PCE formulation in Python. Created the baseline of validation error calculations. Completed Y-Y plots experimentation.
2. Ryo Nakamori: Created the report structure, wrote down algorithm outline, conclusion, and filled in experimental results. Performed global error experiments.
3. Bartu Tuncay: Experimented and and cross-matched our implementation with the one from UQPyLab. Added their results to the report.
4. Anshuman Barnwal: Implemented the original model and input parameters. Added details to the report about model parameter selection and relative error calculation.

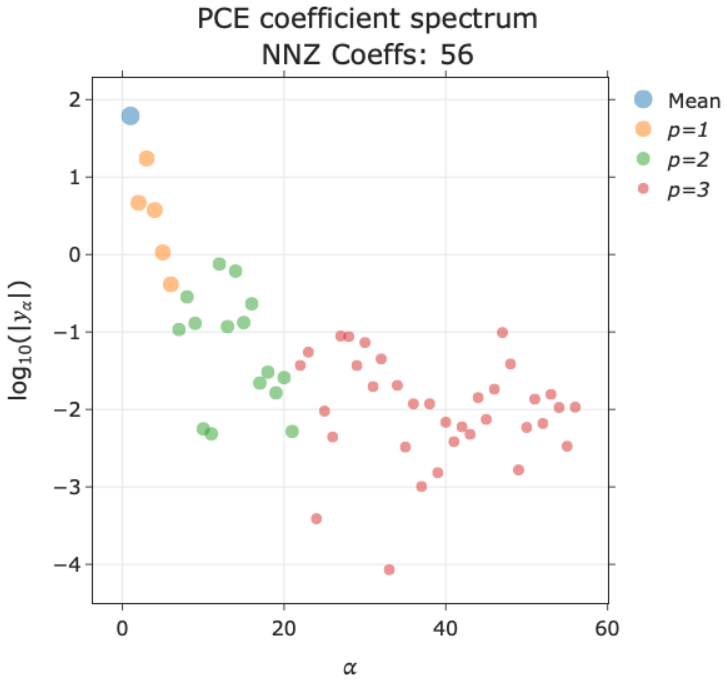


Figure 10: PCE coefficient spectrum (UQ[py]Lab model)

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every written paper or thesis authored during the course of studies. **In consultation with the supervisor**, one of the following two options must be selected:

☑ I hereby declare that I authored the work in question independently, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used no generative artificial intelligence technologies[1].

☑ I hereby declare that I authored the work in question independently. In doing so I only used the authorised aids, which included suggestions from the supervisor regarding language and content and generative artificial intelligence technologies. The use of the latter and the respective source declarations proceeded in consultation with the supervisor.

Title of paper or thesis:

Uncertainty Quantification Mini Project: Surrogating Black Scholes using PCE

Authored by:
If the work was compiled in a group, the names of all authors are required.

Last name(s):	**First name(s):**
Tee	Chin Zhe
Nakamori	Ryo
Tuncay	Bartu
Barnwal	Anshuman

With my signature I confirm the following:
- I have adhered to the rules set out in the Citation Guidelines.
- I have documented all methods, data and processes truthfully and fully.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for originality.

Place, date	**Signature(s)**
Zürich, 21-05-25	ChinZhe Tee
	Ryo Nakamori
	BT
	Anshuman

If the work was compiled in a group, the names of all authors are required. Through their signatures they vouch jointly for the entire content of the written work.

[1] For further information please consult the ETH Zurich websites, e.g. https://ethz.ch/en/the-eth-zurich/education/ai-in-education.html and https://library.ethz.ch/en/researching-and-publishing/scientific-writing-at-eth-zurich.html (subject to change).