OpenTURNS (parts of) release highlights

M.Baudin (EDF)

User Day #18, June 13th 2025, EDF Lab

Contents

Joint work with A. Dutfoy, J.Schueller.

QuantileConfidence

- Goal: estimate the confidence interval of a quantile[1].
- In the nuclear industry, this is called[2] "Wilks's method", but **this is not proper**, since Wilks's paper introduces a tolerance interval, not a confidence interval (this is different in the bilateral case).

DETERMINATION OF SAMPLE SIZES FOR SETTING TOLERANCE LIMITS

By S. S. Wilks

Princeton University, Princeton, N. J.

[1] See Meeker, W. Q., Hahn, G. J., and Escobar, L. A. (2017). *Statistical intervals: a guide for practitioners and researchers*, volume 541. John Wiley & Sons.

[2] See Wilks, S. S. (1941). *Determination of sample sizes for setting tolerance limits.* The Annals of Mathematical Statistics, 12(1), 91-96.

QuantileConfidence

Features[3]

- Compute a confidence interval of a quantile from a sample without any hypothesis on the distribution of the sample. Unilateral or bilateral.
- Compute the sample size so that the extreme observations of a sample create a confidence interval of a quantile.
- Based on **efficient algorithms**: e.g. no hard-coded upper bounds on the sample size, no iterative algorithm if a special function can be used, etc.
- Compute either an $\mathrm{ot.Interval}$ or the rank of the order statistics.
- Compute an asymptotic bilateral confidence interval.
- Many formulas are based on the quantile (or complementary quantile) of level β (the confidence level) of the binomial distribution with parameters α (the quantile level) and n (the sample size).
- The Wilks class is deprecated in 1.25.
- In OT 1.24: Improves the documentation[4]

[3] See PR #2882.
[4] See PR #2712.

QuantileConfidence

Example. Compute the upper tail confidence interval $\left]-\infty, X_{(k_{up})}\right]$ such that:

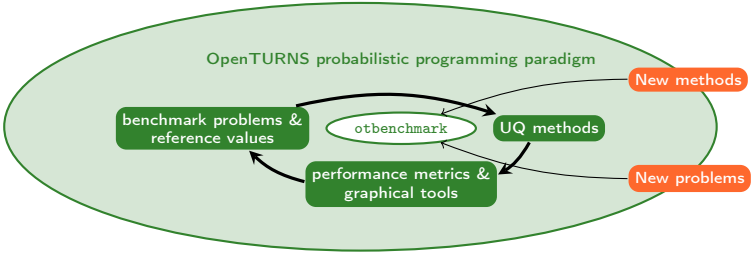
$$\mathbb{P}\left(x_\alpha \in \left]-\infty, X_{(k_{up})}\right]\right) \geq \beta.$$

```
import openturns as ot
import openturns.experimental as otexp
alpha = 0.05  # The quantile level
beta = 0.95  # The confidence level
algo = otexp.QuantileConfidence(alpha, beta)
# If the size is known
rank = algo.computeUnilateralRank(100)  # Returns rank = 9 in {0, ..., 99}
# On a ot.Sample
sample = ot.Gumbel().getSample(100)
ci = algo.computeUnilateralConfidenceInterval(sample)
```

Depending on the value of the parameters, an exception may be produced if the sample size is too small or the confidence level is too close to 1.

otbenchmark

Joint work with E. Fekhari, M. Baudin, V. Chabridon, Y. Jebroun, J. Schueller.
otbenchmark[5] is a benchmark package for Uncertainty Quantification.



Use cases:

- test a new UQ algorithm on a panel of problems
- compare several UQ algorithms available on a given benchmark problem

[5] See Fekhari, E., Baudin, M., Chabridon, V., & Jebroun, Y. (2021). *otbenchmark: An open source Python package for benchmarking and validating uncertainty quantification algorithms.* In 4th International Conference on Uncertainty Quantification in Computational Sciences and Engineering.

Two categories of benchmark classes are currently provided:

- reliability problems, i.e. estimating the probability that the output of a function is less than a threshold,
- sensitivity problems, i.e. estimating sensitivity indices, for example Sobol' indices.

Features:

- Most of the reliability problems were adapted from the RPRepo[6].
- Create a problem, run an algorithm and compare the computed probability with a reference probability.
- Loop over all problems and run several methods on these problems.
- **26 reliability problems** and **12 sensitivity analysis problems** so far
- Reference values either computed by exact quadrature methods or large Monte Carlo sampling.

[6] See Rozsas A., Slobbe A. (2019). Repository and Black-box Reliability Challenge 2019. https://rprepo.readthedocs.io/en/latest/.

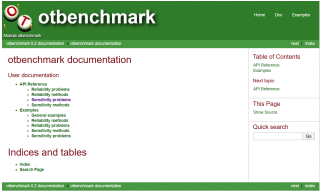
otbenchmark

Already presented[7] at OpenTURNS User's Day 2019.

New in 2025:

- Repo moved into openturns: https://github.com/openturns/otbenchmark
- Online documentation: https://openturns.github.io/otbenchmark/master/
- Conda and pip packaging[8]:

```
conda install otbenchmark  # ... or ....
pip install otbenchmark
```



[7] See Fekhari, E., Baudin, M., Chabridon, V., & Jebroun, Y. (2021, June 9). *OTBenchmark: An open source Python package for benchmarking and validating uncertainty quantification algorithms*. OpenTURNS User's Day 2019. jot_efekhari21.pdf.

[8] See here and there.