



Befitting bootstrap analysis: A case study

Ron S. Kenett, Chris Gotwalt, Laura Freeman, Peter Gedeck & Xinwei Deng

To cite this article: Ron S. Kenett, Chris Gotwalt, Laura Freeman, Peter Gedeck & Xinwei Deng (02 Sep 2025): Befitting bootstrap analysis: A case study, Quality Engineering, DOI: [10.1080/08982112.2025.2551756](https://doi.org/10.1080/08982112.2025.2551756)

To link to this article: <https://doi.org/10.1080/08982112.2025.2551756>



View supplementary material [↗](#)



Published online: 02 Sep 2025.



Submit your article to this journal [↗](#)



Article views: 35



View related articles [↗](#)



View Crossmark data [↗](#)



Befitting bootstrap analysis: A case study

Ron S. Kenett^a , Chris Gotwalt^b, Laura Freeman^c, Peter Gedeck^d, and Xinwei Deng^c 

^aThe KPA Group and the Samuel Neaman Institute, Technion, Israel; ^bJMP Division, SAS, Research Triangle, North Carolina;

^cDepartment of Statistics, Virginia Tech, Blacksburg, Virginia; ^dSchool of Data Science, University of Virginia, Charlottesville, Virginia

ABSTRACT

Bootstrapping is a popular method for inference and uncertainty quantifications. The key element of bootstrapping analysis is computing distributions of statistical estimators by resampling, with replacement, of a given data set. However, in practice, data often have some inherent data structure reflecting the data-generation process. The point in this article is that the corresponding bootstrap analysis needs to incorporate such information to enhance the quality of inference and uncertainty quantification. We propose applying a befitting bootstrap analysis (BBA) method reflecting the data generation structure. The proposed befitting bootstrap analysis method generalizes findings to a population frame with similar data generation processes. It is a follow up to the befitting cross validation (BCV) method proposed earlier by the same authors. A case study is used to elaborate the merits of the befitting bootstrap analysis method, in comparison with several conventional bootstrapping methods. The Python code used in the analysis is available in an openly available Github repository.

KEYWORDS

bootstrapping; befitting bootstrap analysis (BBA); befitting cross validation (BCV); statistical uncertainty; model goodness of fit





1. Introduction


Bootstrapping is a statistical procedure that resamples a single dataset to create many simulated samples. This process allows the user to calculate properties of statistical estimators such as standard errors, construct confidence intervals, and perform hypothesis testing. Bootstrap methods are model free alternatives to traditional hypothesis testing, notable for being easier to understand and requiring minimal assumptions, with the key assumption being that the sample data is representative of the population from which it was drawn. Bootstrapping can be used to analyze designed experiments. This needs to be considered with care. Designed experiments, typically, consist of balanced arrays of experimental runs that allow for efficient estimation of factor effects and their interactions. However, in running designed experiments one often meets unanticipated problems. Some of such issues can be dealt with in the design phase. For example, the impact of raw material batch or shifts can be accounted for by running experiments in separate blocks. Practical constraints may dictate that some factors are 'nested' within others or that there are limitations on the run order. In other examples, some

experimental points may turn out impossible to execute because of logistics or technological requirements. Unexpected problems may also arise when experiments are carried out (Kenett, Rahav, and Steinberg 2006). Other examples, where there is a structure generating the data, include students in schools where one needs to account for classes, teachers, schools and districts. In industry, products are often manufactured in batches affected by the production process, maintenance schedule or work shifts. The befitting bootstrap analysis (BBA) we introduce here provides a working approach to statistical inference accounting for the data generation structure in designed experiments or observable studies.

The original bootstrap methodology was introduced by Efron (1979) as a method of statistical inference, without the need for extensive assumptions and intricate theory. Generally, bootstrapping analysis (BA) is based on resampling with replacement. It was originally motivated as a general tool for uncertainty quantification (Efron and Hastie 2016).

There are several bootstrapping techniques. Generally, these can be classified as parametric or nonparametric methods. The parametric bootstrap

CONTACT Ron S. Kenett  ron@kpa-group.com  The KPA Group, PoBox 2525, Hataasia Street 25, Raanana 4365413, Israel.; Xinwei Deng  xdeng@vt.edu
 Department of Statistics, Virginia Tech, 250 Drillfield Drive, Blacksburg, VA 24061.

 Supplemental data for this article can be accessed online at <https://doi.org/10.1080/08982112.2025.2551756>.

© 2025 Taylor & Francis Group, LLC

generates new values of the responses from probability distributions characterized by the observed data. In contrast, the nonparametric bootstrap methods apply resampling with replacement to the observed data. With the generated bootstrapped datasets, one can conduct uncertainty quantification and model inference. See Kenett and Zacks (2021) and Kenett, Zacks, and Gedeck (2022b, 2023) for the implementation of bootstrap techniques using JMP, R and Python programming languages.

In the practice of using bootstrap analysis techniques in real situations, the data often has an inherent structure. In industrial applications, data is collected from products, produced in batches or continuously using specific equipment that operates in shifts. In a hospital, treated patients have medical records reflecting their clinical condition and lab test results. Other records document the treating medical staff, administered drug products and conditions of equipment. It is important to combine domain knowledge with statistical sampling methods to derive empirical distributions reflecting uncertainty in estimators. The befitting bootstrap analysis (BBA) approach introduced here enhances bootstrap techniques by matching the bootstrapping resampling techniques with the data structure, as derived from domain knowledge. The BBA is an effective approach to both assess model goodness of fit and to demonstrate significance of effects. It provides uncertainty quantification for both basic and complex predictive models.

BBA is based on principles similar to befitting cross validation (BCV) presented in Kenett et al. (2022a). BCV focuses on assessment of model prediction, while BBA emphasizes model inference and uncertainty quantification. This parallels the distinction made by W. E. Deming between analytic and enumerative studies (Deming 1953). The BBA strategy is a generalization of bootstrapping for designed experiment with replicated runs, as proposed in Kenett, Rahav, and Steinberg 2006. In designed experiments with replicates, BBA consists of resampling, separately, the replicates of individual experimental runs. The BBA approach extends this original method to more general conditions of observable data such as data stratification or hierarchical structures.

Throughout the article, we consider 1000 bootstrapped replicates by sampling with replacement. The remainder of this work is organized as follows. Section 2 covers methodological background and bootstrapping details. Section 2 provides details of the BBA method. Section 3 demonstrates the merits of BBA through a case study. A sensitivity analysis was

conducted and is presented in [Supplementary Material](#). The Python code is available in public link at the end of the article.

2. Background

The importance of the data-generating structure in statistical analysis can be traced back to Cornfield and Tukey (1956), where they consider the two-way classification model under different data structure. Tukey's (1958) jackknife method was based on the idea in Quenouille (1949) of using parts of a sample (i.e., sample with the i th observation omitted) to estimate bias and thus come up with an estimator with reduced bias. Efron (1979) introduced the "bootstrap" as a general method for estimating the sampling distribution of a statistic based on the observed data. In Benjamini and Fuchs (1990), the researchers emphasize the importance of considering the level of conditioning in regression analysis, particularly in complex models. They also suggest using bootstrap methods to estimate variability under different conditioning assumptions, providing valuable insights for researchers dealing with regression models. In this context resampling methods are used to estimate the variability of parameters, considering the level of conditioning in the data.

Olatayo, Amahia, and Obilade (2010) consider the application of a bootstrap method to a stochastic time series process with a non-parametric bootstrap method called a truncated geometric bootstrap method for stationary time series data. The procedure attempts to mimic the original model by retaining the stationarity property of the original series in the resample pseudo-time series.

Ghosh, Hastie, and Owen (2022) map a set of general data structures including balanced data, nested rows and nested columns. They focus on large scale crossed random effects regressions that account for the data structure. In general, bootstrap techniques were adapted to match the structure of the data. This includes block bootstrap for time series data (Dudek et al. 2014), multilevel bootstrap for data with nested structure (Modugno and Giannerini 2015; Saravanan, Berman, and Sober 2020), and spatial bootstrap for spatial data (Dumanjug, Barrios, and Lansangan 2010). Here, we emphasize the general importance of incorporating the information of data-generating structure to enhance bootstrap for inference and uncertainty quantification.

3. Befitting bootstrap analysis

Befitting bootstrap analysis (BBA) is a self-supervised bootstrap analysis method. The main characteristic of BBA is that it explicitly embraces the data-generating structure. It is a follow up to the befitting cross validation (BCV) proposal in Kenett et al. (2022a). Consider a data set $\mathbf{D} = (\mathbf{X}, \mathbf{y})$ where rows of \mathbf{X} correspond to n observations of p predictor variables, and a response vector \mathbf{y} , corresponds to the dependent variables. To enable uncertainty quantification in models of the observed data, one can use bootstrapping. The bootstrap method consists of a repeated sampling, with replacement, of the original data to generate a new dataset \mathbf{D}^* . The newly generated dataset, \mathbf{D}^* , is repeatedly used for model fitting and parameter estimation. To mirror the data-generating structure, BBA enforces the following properties for \mathbf{D}^* :

BBA Principle 1: The bootstrapped resampled dataset in BBA inherits the same data generation structure as the original dataset.

BBA Principle 2: The bootstrap resampling in BBA reflects the planned and unplanned constraints affecting the data collection.

BBA can be presented by a three-layer hierarchical structure of the data with variable Z_1 , Z_2 , and Z_3 , where Z_1 has I levels, Z_2 has J levels, and Z_3 has K levels. For each data point, the structure is

$$\mathbf{d}_{ijkl} = (\mathbf{x}_{ijkl}, y_{ijkl}) \text{ for } i \in \{1, \dots, I\}, \\ j \in \{1, \dots, J\} \text{ and } k \in \{1, \dots, K\},$$

where $l \in \{1, \dots, L\}$ represents the l -th replications. For notational convenience, we define $\mathbf{D}_{i\dots}$ to be the subset of data consisting of all data points corresponding to $Z_1 = i$. That is,

$$\mathbf{D}_{i\dots} = \{\mathbf{d}_{ijkl} : \text{for } j = 1, \dots, J; \\ k = 1, \dots, K, \text{ and } l = 1, \dots, L\}.$$

Similarly, we define $\mathbf{D}_{j\dots}$ to be the subset of data consisting of data points with corresponding $Z_2 = j$ and $\mathbf{D}_{k\dots}$ to be the subset of data consisting all data points with corresponding $Z_3 = k$. Furthermore, we define $\mathbf{D}_{ij\dots}$ to be the subset of data points consisting of all data points with $Z_1 = i$ and $Z_2 = j$. That is,

$$\mathbf{D}_{ij\dots} = \{\mathbf{d}_{ijkl} = (\mathbf{x}_{ijkl}, y_{ijkl}) : \text{for } k = 1, \dots, K, \text{ and} \\ l = 1, \dots, L\}.$$

Similarly, we define $\mathbf{D}_{i.k}$ to be the subset of data points consisting of all data points with $Z_1 = i$ and $Z_3 =$

k , and $\mathbf{D}_{j.k}$ to be the subset consisting of all data points with the corresponding $Z_2 = j$ and $Z_3 = k$.

Define $\mathbf{D}_{ijk\dots}$ to be the subset of L data points (i.e., replications) with the corresponding $Z_1 = i$, $Z_2 = j$, and $Z_3 = k$. That is,

$$\mathbf{D}_{ijk\dots} = \{\mathbf{d}_{ijkl} = (\mathbf{x}_{ijkl}, y_{ijkl}) : l = 1, \dots, L\}.$$

For BBA of data with a hierarchical structure, we first perform the sampling with replacement to the existing levels of Z_1 . This consists of resampling with replacement L sub-data sets, $(\mathbf{X}_{1\dots}, \mathbf{y}_{1\dots}), \dots, (\mathbf{X}_{I\dots}, \mathbf{y}_{I\dots})$. Then, for each selected level of Z_1 , we perform sampling with replacement for the existing level of Z_2 . Specifically, given a selected level i of Z_1 , its corresponding sub-data set $(\mathbf{X}_{i\dots}, \mathbf{y}_{i\dots})$ contains subsets of data $(\mathbf{X}_{i1\dots}, \mathbf{y}_{i1\dots}), \dots, (\mathbf{X}_{ij\dots}, \mathbf{y}_{ij\dots})$. Thus, we conduct resampling with replacement for K sub-data sets. We continue such a procedure until the sampling with replacement covers all the existing levels of Z_3 .

The impact of BBA relates to the structure of the data generation process. For data affected by a structure with significant effects, BBA will counteract the possible bias of ignoring this structure in bootstrapping.

Based on the data generation structure, and the planned or unplanned constraints in data collection, one can conduct resampling with replacement at different levels or different level combinations. For example, we may consider sampling with replacement for a subset $\mathbf{X}_{ij\dots}$, $i = 1, \dots, I$, and $j = 1, \dots, J$. In another example, one can consider sampling with replacement for data points in each subset $\mathbf{X}_{ijk\dots}$ such as in analyzing designed experiments with replicates, as presented in Kenett, Rahav, and Steinberg (2006).

BBA is not restricted to nonparametric bootstrap. It can also be applied in parametric befitting bootstrap analysis (pBBA). In pBBA, a parametric model is used to fit the whole data as

$$y_{ijkl} = f(\mathbf{x}_{ijkl}, \boldsymbol{\beta}) + \varepsilon_{ijkl},$$

where $\boldsymbol{\beta}$ represents the model parameters and ε_{ijkl} is the error term with zero mean. The key idea in pBBA is to conduct resampling based on means and standard deviations of replicates. In its simplest form, pBBA replaces the observed response with random numbers from a normal distribution fitting the replicates. Other distributions can also be considered. A case study with BA and BBA will be presented in the next section. Parametric bootstrap analysis (pBA) and Wild Bootstrap Analysis (wBA) are also introduced and applied in the next section, for comparison.

Comparing results of BBA and pBBA to pBA and wBA will show us the impact of BBA accounting for the data generation process.

4. A Case study

The case study is a piston operating in a combustion engine. The Piston Simulation function models the circular motion of a piston within a cylinder. It involves a chain of nonlinear functions. The performance of the piston is measured by the cycle time of a full revolution, in seconds. Thus, we consider the response y to be the cycle time. The piston simulation function is expressed as

$$y = 2\pi \sqrt{\frac{M}{k + S2 \frac{P_0 V_0 T_a}{T_0 V^2}}}$$

where $V = \frac{S}{2k} \left(\sqrt{A^2 + 4k \frac{P_0 V_0}{T_0} T_a} - A \right)$ and $A = P_0 S + 19.62M - \frac{kV_0}{S}$.

Listed below are the seven factors used to control the piston. In bold, the factors used in the BBA case study:

1. m : piston weight (Low = 30Kg, High = 60Kg),
2. **s : piston surface area** (Low = 0.005m², High = 0.2m²),
3. **v_0 : initial gas volume** (Low = 0.002m³, High = 0.01m³),
4. **k : spring coefficient** (Low = 1,000N/m, High = 5,000N/m),
5. P_0 : atmospheric pressure (Low = 90,000 N/m², High = 110,000N/m²),
6. t : ambient temperature (Low = 290⁰K, High = 296⁰K)
7. **t_0 : gas temperature** (Low = 340⁰K, High = 360⁰K).

The levels of these factors, shown in parentheses, represent extremes on the operating range that cannot be exceeded without affecting the smooth operation of the engine. The data is derived from simulator running at specific factor level combinations with uncertainty in the factors. The piston simulator code is available in R and JMP, Kenett and Zacks (2021), in Python, Kenett, Zacks, and Gedeck (2022b, 2023) and in Matlab, see Simon Fraser Virtual Lab (2023). Figure 1 presents a distribution of the piston running at nominal (center) level of all 7 factors. The average of 50 cycle time is 0.77 s and the standard deviation is 0.006 s.

To map a response surface enabling improvement and optimization, we run a Central Composite Design (CCD), with the 4 factors listed in Table 1. This

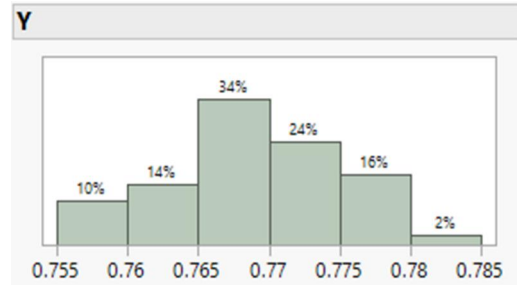


Figure 1. Distribution of cycle times for piston operating at nominal values (JMP 17.2 output).

Table 1. Central Composite Design on 4 factors of the piston simulator with 5 replicates (partial listing) and distribution of replicates (groups), the four factors are: x1 (piston surface area), x2 (initial gas volume), x3 (spring coefficient) and x4 (filling gas temperature).

Cycle time [ms]	Group	x1	x2	x3	x4
60.226485	1	-1.0	-1.0	-1.0	-1.0
65.020360	1	-1.0	-1.0	-1.0	-1.0
57.302374	1	-1.0	-1.0	-1.0	-1.0
50.978262	1	-1.0	-1.0	-1.0	-1.0
56.526159	1	-1.0	-1.0	-1.0	-1.0
...
68.963006	32	0.0	0.0	0.0	0.0
51.328098	32	0.0	0.0	0.0	0.0
51.372896	32	0.0	0.0	0.0	0.0
59.049678	32	0.0	0.0	0.0	0.0
67.187754	32	0.0	0.0	0.0	0.0

Table 2. Main effect OLS model with 4 factors of the piston simulator.

	Coeff	std err	t	P> t	[0.025	0.975]
Intercept	63.6518	0.823	77.302	0.000	62.025	65.278
x1	-15.5256	0.951	-16.329	0.000	-17.404	-13.647
x2	13.0152	0.951	13.689	0.000	11.137	14.893
x3	4.7356	0.951	4.981	0.000	2.857	6.614
x4	0.3212	0.951	0.338	0.736	-1.557	2.199

The .025 and .975 columns represent limits of a 95% confidence interval (from python available in open access link).

consists of 32 experimental points with 5 replicates each. Table 1 provides a partial listing of the experimental runs. In group 1 all factors are at level Low=-1. In group 32 all factors are at level Nominal = 0. The first column represents the piston cycle time for these conditions. For details on experimental designs see Kenett, Zacks, and Gedeck (2023). To fit a model, we use ordinary least squares regression (OLS). Our goal here is to demonstrate the application of BBA. In Table 2, we show the OLS fit to a model with only main effects. It indicates x4 as non-significant. In Table 3, we fit a model without x4, but with all two-way interactions and quadratic effects of factors x1, x2, x3.

To compare the proposed method (BBA) with other existing methods of bootstrap (BA, pBBA, pBA, wBA) we consider the methods defined below:

Table 3. Main effect, two-way interactions, and quadratic effect OLS model with 3 factors of the piston simulator.

	Coeff	std err	t	P> t	[0.025	0.975]
Intercept	63.3069	1.197	50.397	0.000	57.942	62.671
x1	-15.5256	0.788	-19.709	0.000	-17.082	-13.969
x2	13.0152	0.788	16.523	0.000	11.459	14.572
x3	4.7356	0.788	6.012	0.000	3.179	6.292
x1:x2	-4.9101	0.965	-5.089	0.000	-6.816	-3.004
x1:x3	-2.2820	0.965	-2.365	0.019	-4.188	-0.376
x2:x3	1.6966	0.965	1.759	0.081	-0.210	3.603
l(x1 ** 2)	4.5094	0.708	6.370	0.000	3.111	5.908
l(x2 ** 2)	0.2682	0.708	0.379	0.705	-1.131	1.667
l(x3 ** 2)	-0.3178	0.708	-0.449	0.654	-1.717	1.081

The .025 and .975 columns represent limits of a 95% confidence interval (from python available in open access link).

(M₁) Befitting Bootstrap analysis (BBA). In BBA, bootstrapping is applied with replicate sets only or following the data generation process structure.

(M₂) Bootstrap analysis (BA). In BA, bootstrapping is applied to the whole data set.

(M₃) Parametric Befitting Bootstrap analysis (pBBA). This is a parametric befitting bootstrap resampling based on means and standard deviations of replicates. In its simplest form, pBBA replaces the observed response with random numbers from a normal distribution fitting the replicates. Other distributions can also be considered but we do not show this here.

(M₄) Parametric bootstrap analysis (pBA) is based on bootstrapping residuals from a model fit to the data. This is model-dependent, as opposed to pBBA, which is model-independent and only and only assumes a normal distribution of the replicates.

(M₅) Wild Bootstrap Analysis (wBA) fits a model using the full original dataset. For each resampling set, one multiplies the residuals by a random value sampled from a normal distribution $N(0, 1)$ and adds it to the fitted values from the original model. wBA, like pBA, is also model-dependent (Mammen 1993).

These five approaches provide different forms of uncertainty in statistical estimators. They can be used to check the goodness-of-fit of a model by comparing the bootstrapped empirical variability of estimators and their theoretical variability. Discrepancies between these two values indicates lack of fit (Kenett, Rahav, and Steinberg 2006). To evaluate the model fitting, we focus on the standard errors of the effects. We then apply BA and BBA and compare the bootstrap standard errors of the effects to the regression OLS standard error and to each other. To facilitate this comparison, we define a relative measure of

Table 4. Standard errors of effects from OLS model and BBA.

	Main			Full		
	Regr.	BBA	Delta	Regr.	BBA	Delta
Intercept	0.82342	0.60483	-26.5	1.19664	1.03030	-13.9
x1	0.95080	0.76040	-20.0	0.78772	0.80110	1.7
x2	0.95080	0.68112	-28.4	0.78772	0.66288	-15.8
x3	0.95080	0.65951	-30.6	0.78772	0.67731	-14.0
x4	0.95080	0.63040	-33.7			
x1:x2				0.96476	0.92408	-4.2
x1:x3				0.96476	0.88620	-8.1
x2:x3				0.96476	0.91043	-5.6
l(x1 ** 2)				0.70794	0.73775	4.2
l(x2 ** 2)				0.70794	0.51710	-27.0
l(x3 ** 2)				0.70794	0.50657	-28.4

Main effect model on the left, full model on the right (from python available in open access link).

comparison, denoted as Δ (Delta):

$$\Delta = \frac{SE(M) - SE(OLS)}{SE(OLS)} 100\% \quad [1]$$

where $SE(M)$ is the bootstrap standard error for the method in comparison, and $SE(OLS)$ is the OLS standard error. Table 3 shows the results for BBA. Table 5 shows the results for BA. Comparing Table 3 and 5 shows that BBA is providing a sharp differentiation between the main effect and the full model, in comparison to BA. In this analysis we use delta in an exploratory sense without providing cutoff values like in tests of significance with a preset p-value.

From Table 4 and Table 2, one can see that the OLS main effect standard errors are 0.9508. The BBA standard errors are smaller. Specifically, for BBA, $\Delta = -20.0\%$, -28.4% , -30.6% , -31.7% for x_1 , x_2 , x_3 and x_4 , respectively. The empirical bootstrap 95% confidence interval for x_4 contains 0, corroborating the non-significant effect of x_4 found in the OLS fit. Consequently, we drop x_4 and fit a full model to x_1 , x_2 , x_3 with all two-way interactions and quadratic effects. With this full model we get, with BBA, $\Delta = +1.7\%$, -15.8% , -14.0% for x_1 , x_2 , and x_3 , respectively. This represents a clear drop in deltas from the reduced main effect model listed in the previous paragraph, indicating higher consistency between OLS standard errors and bootstrap standard errors.

One can see than there are large differences in standard errors between the regression and the bootstrap estimates. Such a qualitative observation serves as a diagnostic tool for model goodness of fit assessment. Large values of Δ indicates inadequate assumptions of the regression model. Here, a reduced model with only main effects is shown to imply a problematic assumption. Note that we provide only a qualitative assessment of this difference, in a similar spirit of the coefficient of determination R^2 . It is possible to consider a more elaborate study of this difference

Table 5. Standard errors of effects from OLS model and BA.

	Main			Full		
	Regr.	BA	Delta	Regr.	BA	Delta
Intercept	0.82342	0.80394	−2.4	1.19664	1.03789	−13.3
x1	0.95080	1.18924	25.1	0.78772	0.87550	11.1
x2	0.95080	0.94900	−0.2	0.78772	0.87109	10.6
x3	0.95080	0.93570	−1.6	0.78772	0.76117	−3.4
x4	0.95080	0.85582	−10.0			
x1:x2				0.96476	1.03973	7.8
x1:x3				0.96476	1.00755	4.4
x2:x3				0.96476	1.00718	4.4
l(x1 ** 2)				0.70794	0.78125	10.4
l(x2 ** 2)				0.70794	0.74039	4.6
l(x3 ** 2)				0.70794	0.56505	−20.2

Main effect model on the left, full model on the right (from python available in open access link).

Table 6. Standard errors of effects from OLS model and pBBA.

	Main			Full		
	Regr.	pBBA	Delta	Regr.	pBBA	Delta
Intercept	0.82342	0.60831	−26.1	1.19664	1.04301	−12.8
x1	0.95080	0.77756	−18.2	0.78772	0.77756	−1.3
x2	0.95080	0.66149	−30.4	0.78772	0.66149	−16.0
x3	0.95080	0.66926	−29.6	0.78772	0.66926	−15.0
x4	0.95080	0.64859	−31.8			
x1:x2				0.96476	0.89139	−7.6
x1:x3				0.96476	0.92286	−4.3
x2:x3				0.96476	0.92514	−4.1
l(x1 ** 2)				0.70794	0.69214	−2.2
l(x2 ** 2)				0.70794	0.54135	−23.5
l(x3 ** 2)				0.70794	0.50015	−29.4

Main effect model on the left, full model on the right (Python).

based on simulations of technical boundaries for interpretation of these gaps.

In Table 5, using BA, we get standard errors larger than OLS and an inconsistent picture. For example, with BA, the standard error of x3, is lower than OLS by 4% in both the reduced and full model. With x1, on the other hand, the standard error relative to OLS is 25.1% and 11.1% for the reduced and full model respectively. For x2 the values of delta are −0.2% and 10.6% for the reduced and full model, respectively. These are confusing results due to the random procedure not being in synch with the data structure.

In summary, with the piston example, we see that BBA-derived standard errors are more consistent estimators than the ones derived from BA. This is because BBA respects the structure in the original data. Tables 6–8 show these results for pBBA, pBA and wBA.

We proceed with a comprehensive analysis that compares BBA, BA, pBBA, pBA and wBA. In Figure 2, the bootstrap distributions of the coefficients are presented as boxplots. The first two boxplots correspond to BBA and BA. The remaining three are for pBBA, pBA and wBA, respectively. The effect of x4 is

Table 7. Standard errors of effects from OLS model and pBA.

	Main			Full		
	Regr.	pBA	Delta	Regr.	pBA	Delta
Intercept	0.82342	0.77803	−5.5	1.19664	1.11645	−6.7
x1	0.95080	0.93016	−2.2	0.78772	0.73997	−6.1
x2	0.95080	0.94416	−0.7	0.78772	0.75939	−3.6
x3	0.95080	0.92727	−2.5	0.78772	0.77580	−1.5
x4	0.95080	0.93728	−1.4			
x1:x2				0.96476	0.95261	−1.3
x1:x3				0.96476	0.92190	−4.4
x2:x3				0.96476	0.89047	−7.7
l(x1 ** 2)				0.70794	0.69731	−1.5
l(x2 ** 2)				0.70794	0.69909	−1.2
l(x3 ** 2)				0.70794	0.70454	−0.5

Main effect model on the left, full model on the right (from python available in open access link).

Table 8. Standard errors of effects from OLS model and wBA.

	Main			Full		
	Regr.	wBA	Delta	Regr.	wBA	Delta
Intercept	0.82342	0.82554	0.3	1.19664	1.04128	−13.0
x1	0.95080	1.17701	23.8	0.78772	0.85080	8.0
x2	0.95080	0.91483	−3.8	0.78772	0.80444	2.1
x3	0.95080	0.86632	−8.9	0.78772	0.74086	−5.9
x4	0.95080	0.85109	−10.5			
x1:x2				0.96476	0.98511	2.1
x1:x3				0.96476	0.97772	1.3
x2:x3				0.96476	0.97938	1.5
l(x1 ** 2)				0.70794	0.77889	10.0
l(x2 ** 2)				0.70794	0.69552	−1.8
l(x3 ** 2)				0.70794	0.53633	−24.2

Main effect model on the left, full model on the right (from python available in open access link).

judged nonsignificant by all methods. In all panels we observe that the spread of coefficients in BBA and pBBA is smaller.

Figure 3 is similar to Figure 2, for the full model. If we focus on interquartile range to assess significance, we notice that the interaction effect x2*x3 is judged significant by BBA, and not significant by the other methods. In the piston simulation function one can see that x2 (initial gas volume V_0) and x3 (spring coefficient k) often appear in a multiplicity in the function. This implies that the effects of x2 (initial gas volume) and x3 (spring coefficient) interact through the nonlinear equation for cycle time. Thus, changes in x3 can alter how x2 influences the response. The proposed BBA method, because of embracing the data structure to conduct uncertainty quantification, provides more appropriate inference on the significance of the interaction effect x2x3. Moreover, throughout the panels for main effects and interactions, we observe that the spread of coefficients in BBA and pBBA is not smaller. It is smaller for the quadratic effects.

In summary, the case study offers several key insights.

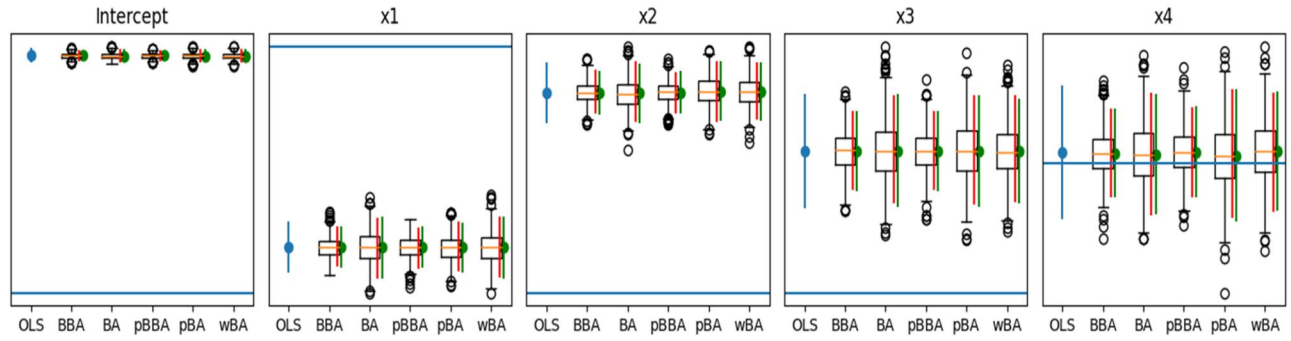


Figure 2. Distribution of main effects coefficient estimates for the piston simulation. Blue: ols estimate \pm std.dev. For each bootstrap approach, boxplot: distribution, red: interquartile range, green: mean \pm std.dev. Blue horizontal line is positioned at 0 (from python available in open access link).

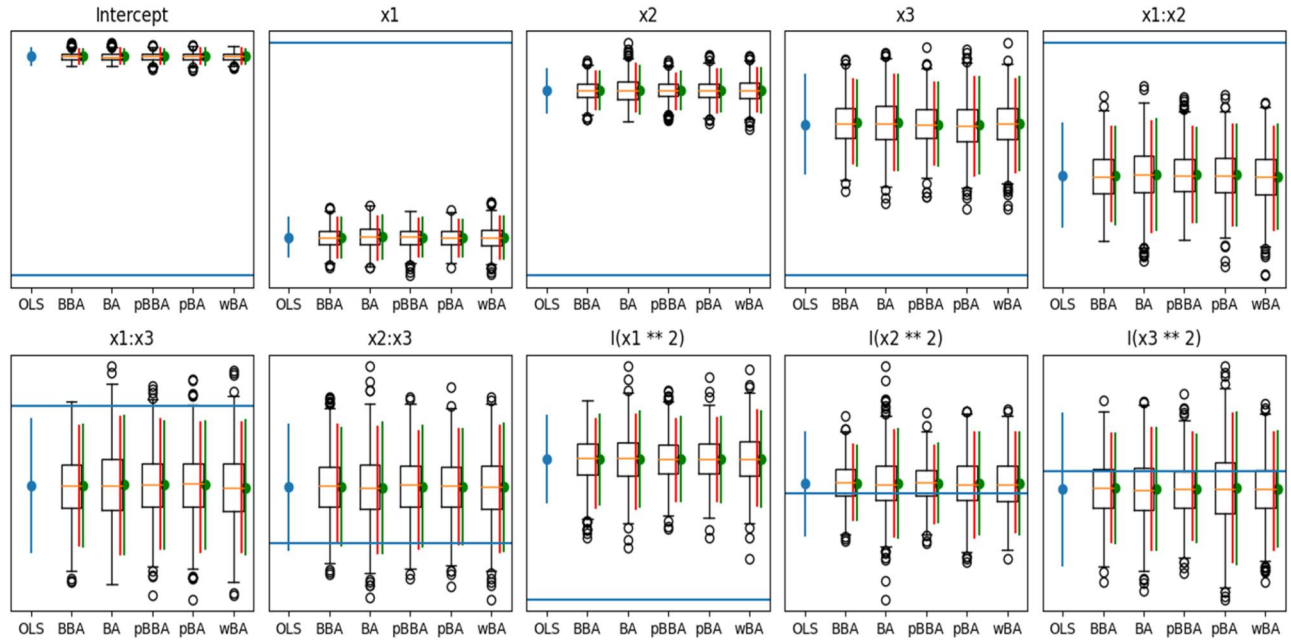


Figure 3. Distribution of main effects, interactions, and quadratic coefficient estimates for the piston simulation. Blue: ols estimate \pm std.dev. For each bootstrap approach, boxplot: distribution, red: interquartile range, green: mean \pm std.dev. Blue horizontal line is positioned at 0 (from python available in open access link).

First, if standard errors of coefficients from BBA are smaller than ordinary least squares (OLS), fit another model.

Here we first expand a model with only main effects to incorporate main effects, interactions and quadratic effects, as made possible by the central composite design (CCD).

Secondly, the BBA spread of effects will be consistently smaller than bootstrap analysis (BA), providing more power to BBA-based inference.

Thirdly, in the case study, BBA is applied to handle data structure set at the design stage. The example is a CCD design with 5 replicates at each run. This can be generalized to other designs with replicated runs and account for conditions where the number of replications is not balanced or if the actual experiment

deviated from the original design. This is demonstrated in [Supplementary Material](#).

5. Discussion

In this work, we introduce a befitting bootstrap analysis (BBA) approach to account for the inherent structure of the data. Compared to other bootstrapping methods, the key advantage of BBA is to take advantage of the data generation structure for more adequate uncertainty quantification and inference.

A case study based on a designed set of experiments with a piston simulator is used in comparing BBA with BA, pBBA, pBA and wBA.

The article presents an application of a BBA to data from a designed experiment, but the

methodology is applicable to data in general, gathered through a designed experiments or simply observed.

In some cases, the data generation structure is not understood. In these situations, a sensitivity analysis of BBA to various possible data generation structures can be carried out, and clustering can be considered for exploring the data generation structure (Li et al. 2022).

The BBA in bootstrapping, like BCV in cross validation, accounts for the data generation structure. It provides generalizability properties of computer-generated uncertainties in estimates.

With unbalanced data, BBA can be also applied when the number of replicates in designed experiments is uneven or when a characterized subset of the data contains very few observations. After introducing and demonstrating BBA, BA, pBA, pBBA and wBA, we focus on BBA and show how it can be used to validate a model fit to data. Further studies are needed for inference on the measure Δ in (1) enabling formal hypothesis testing. It will be also interesting to extend the proposed method to enhance the uncertain quantification in other applications, such as additive manufacturing (Chen et al. 2023; Kang et al. 2023; Wang et al. 2020), computer experiments with complex inputs (Xiao et al. 2021, 2024), and computational social experiments (Hu et al. 2024; Liu et al. 2023). A github repository with the Python code used to generate the results presented in the article is available in <https://github.com/gedeck/bba-case-study>.

About the authors

Professor Ron Kenett is Chairman of the KPA Group, Israel, Chairman of the Data Science Society at AEAI, Senior Research Fellow at the Samuel Neaman Institute, Technion, Haifa, Israel. and Research Professor at the University of Turin, Italy. He is an applied statistician combining expertise in academic, consulting and business domains. education, member of the INFORMS QSR advisory board, member of the advisory board of DSRC, the University of Haifa Data Science Research Center and member of the board of directors in several startup companies. He authored and coauthored over 250 papers and 18 books on topics such as data science, industrial statistics, biostatistics, healthcare, customer surveys, multivariate quality control, risk management, system and software testing, and information quality. He was awarded the 2013 Greenfield Medal by the Royal Statistical Society and, in 2018, the Box Medal by the European Network for Business and Industrial Statistics.

Dr. Chris Gotwalt leads the statistical software development and testing teams for JMP Statistical Discovery. His passion is developing new technologies that accelerate innovation in industry and science. Since joining the company as a PhD student intern in 2001, Gotwalt has

contributed many numerical algorithms and new statistical techniques. He has authored algorithms in JMP for fitting neural networks, linear mixed models, optimal design of experiments, analytical procedures for text analysis, and the algorithms for fitting structural equation models. Gotwalt is a principal investigator for Self-Validating Ensemble Models (S VEM), a procedure that makes machine learning possible for the small data sets often encountered in industry. He holds adjunct professorial positions at North Carolina State University, University of Nebraska and University of New Hampshire, and was the 2020 Chair of the Quality and Productivity Section of the American Statistical Association.

Professor Laura Freeman is Deputy Director, Virginia Tech National Security. Her research leverages experimental methods for conducting research that brings together cyber-physical systems, data science, artificial intelligence (AI), and machine learning to address critical challenges in national security. She is also a hub faculty member in the Commonwealth Cyber Initiative and leads research in AI Assurance. She develops new methods for test and evaluation focusing on emerging system technology. She is the Assistant Dean for Research for the College of Science; in that capacity she works to shape research directions and collaborations in across the College of Science. She previously served as the Intelligent Systems Division director.

Dr. Peter Gedeck is a Lecturer at the School of Data Science at the University of Virginia. His research interests include cheminformatics, research and development, life sciences, and molecular modeling. He has expertise in the intersection of data science and the pharmaceutical industry. Gedeck is also the Research Informatics Senior Scientist at Collaborative Drug Discovery, where he develops useful, production quality drug discovery software. He owns Peter Gedeck, LLC, which creates novel cheminformatics approaches for drug discovery. Custom software development for scientific software companies. He has coauthored four books, Practical Statistics for Data Scientists, Data Mining for Business Analytics: Concepts, Techniques, and Applications, Modern Statistics: A Computer Based Approach with Python with Ron Kenett and Shelemyahu Zacks and Industrial Statistics: A Computer Based Approach with Python with Ron Kenett and Shelemyahu Zacks, both published by Springer/

Professor Xinwei Deng is a professor of statistics and data science faculty fellow at Virginia Tech. He received his Bachelor's degree in mathematics from Nanjing University, and PhD degree in statistics from Georgia Tech. His research interests focus on statistical design, modeling, and decision-making, especially in design and analysis of computer experiments, high-dimensional classification, graphical model estimation, and the interface between experimental design and machine learning. He is an elected member of ISI, and a member of INFORMS and ASA.

Acknowledgements

The authors acknowledge the meticulous comments of two anonymous reviewers and the editor that helped improve the article. Their contribution is gratefully acknowledged.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Ron S. Kenett  <http://orcid.org/0000-0003-2315-0477>

Xinwei Deng  <http://orcid.org/0000-0002-1560-2405>

References

- Benjamini, Y., and C. Fuchs. 1990. Conditional versus unconditional analysis in some regression models. *Communications in Statistics - Theory and Methods* 19 (12):4731–56. doi: [10.1080/03610929008830471](https://doi.org/10.1080/03610929008830471).
- Chen, X., X. Kang, R. Jin, and X. Deng. 2023. Bayesian sparse regression for mixed multi-responses with application to run-time metrics prediction in fog manufacturing. *Technometrics* 65 (2):206–19. doi: [10.1080/00401706.2022.2134928](https://doi.org/10.1080/00401706.2022.2134928).
- Cornfield, J., and J. W. Tukey. 1956. Average values of mean squares in factorials. *The Annals of Mathematical Statistics* 27 (4):907–49. doi: [10.1214/aoms/1177728067](https://doi.org/10.1214/aoms/1177728067).
- Deming, W. E. 1953. On the distinction between enumerative and analytic surveys. *Journal of the American Statistical Association* 48 (262):244–55. doi: [10.1080/01621459.1953.10483470](https://doi.org/10.1080/01621459.1953.10483470).
- Dudek, A. E., J. Leśkow, E. Paparoditis, and D. N. Politis. 2014. A generalized block bootstrap for seasonal time series. *Journal of Time Series Analysis* 35 (2):89–114. doi: [10.1002/jtsa.12053](https://doi.org/10.1002/jtsa.12053).
- Dumanjug, C. F., E. B. Barrios, and J. G. Lansangan. 2010. Bootstrap procedures in a spatial-temporal model. *Journal of Statistical Computation and Simulation* 80 (7): 809–22. doi: [10.1080/00949650902785209](https://doi.org/10.1080/00949650902785209).
- Efron, B. 1979. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics* 7 (1):1–26. doi: [10.1214/aos/1176344552](https://doi.org/10.1214/aos/1176344552).
- Efron, B., and T. Hastie. 2016. *Computer age statistical inference*. New York: Cambridge University.
- Ghosh, S., T. Hastie, and A. B. Owen. 2022. Backfitting for large scale crossed random effects regressions. *The Annals of Statistics* 50 (1):560–83. doi: [10.1214/21-AOS2121](https://doi.org/10.1214/21-AOS2121).
- Hu, Z., X. Liu, X. Deng, and C. J. Kuhlman. 2024. An uncertainty quantification framework for agent-based modeling and simulation in networked anagram games. *Journal of Simulation* 18 (4):505–23. doi: [10.1080/17477778.2024.2313134](https://doi.org/10.1080/17477778.2024.2313134).
- Kang, S., R. Jin, X. Deng, and R. S. Kenett. 2023. Challenges of modeling and analysis in cybermanufacturing: A review from a machine learning and computation perspective. *Journal of Intelligent Manufacturing* 34 (2):415–28. doi: [10.1007/s10845-021-01817-9](https://doi.org/10.1007/s10845-021-01817-9).
- Kenett, R. S., and S. Zacks. 2021. *Modern industrial statistics: With applications in R, MINITAB, and JMP*. 3rd ed. UK: Wiley.
- Kenett, R. S., C. Gotwalt, L. Freeman, and X. Deng. 2022a. Self-supervised cross validation using data generation structure. *Applied Stochastic Models in Business and Industry* 38 (5):750–65. doi: [10.1002/asmb.2701](https://doi.org/10.1002/asmb.2701).
- Kenett, R. S., E. Rahav, and D. M. Steinberg. 2006. Bootstrap analysis of designed experiments. *Quality and Reliability Engineering International* 22 (6):659–67. doi: [10.1002/qre.802](https://doi.org/10.1002/qre.802).
- Kenett, R. S., S. Zacks, and P. Gedeck. 2022b. *Modern statistics: A computer-based approach with Python*. Switzerland: Springer.
- Kenett, R. S., S. Zacks, and P. Gedeck. 2023. *Industrial statistics: A computer-based approach with Python*. Switzerland: Springer.
- Li, Y., X. Deng, S. Ba, W. R. Myers, W. A. Brennenman, S. J. Lange, R. Zink, and R. Jin. 2022. Cluster-based data filtering for manufacturing big data systems. *Journal of Quality Technology* 54 (3):290–302. doi: [10.1080/00224065.2021.1889420](https://doi.org/10.1080/00224065.2021.1889420).
- Liu, X., Z. Hu, X. Deng, and C. J. Kuhlman. 2023. Uncertainty visualization for characterizing heterogeneous human behaviors in discrete dynamical system models. *Advances in Complex Systems* 26 (03):2340001. doi: [10.1142/S0219525923400015](https://doi.org/10.1142/S0219525923400015).
- Mammen, E. 1993. Bootstrap and wild bootstrap for high dimensional linear models. *The Annals of Statistics* 21 (1):255–85. doi: [10.1214/aos/1176349025](https://doi.org/10.1214/aos/1176349025).
- Modugno, L., and S. Giannerini. 2015. The wild bootstrap for multilevel models. *Communications in Statistics - Theory and Methods* 44 (22):4812–25. doi: [10.1080/03610926.2013.802807](https://doi.org/10.1080/03610926.2013.802807).
- Olatayo, T. O., Amahia, G. N., & Obilade, T. O. (2010). Bootstrap method for dependent data structure and measure of statistical precision. *Journal of Mathematics and Statistics*, 6(2), 84–88. doi: [10.3844/jmssp.2010.84.88](https://doi.org/10.3844/jmssp.2010.84.88).
- Quenouille, M. H. 1949. Problems in plane sampling. *The Annals of Mathematical Statistics* 355–375.
- Saravanan, V., G. J. Berman, and S. J. Sober. 2020. Application of the hierarchical bootstrap to multi-level data in neuroscience. *Neurons, Behavior, Data Analysis and Theory* 3 (5):1–29. <https://arxiv.org/pdf/2007.07797.pdf>.
- Simon Fraser Virtual Lab. 2023. <https://www.sfu.ca/~ssurjano/index.html> (accessed October 20, 2023).
- Wang, L., X. Chen, S. Kang, X. Deng, and R. Jin. 2020. Meta-modeling of high-fidelity FEA simulation for efficient product and process design in additive manufacturing. *Additive Manufacturing* 35:101211. doi: [10.1016/j.addma.2020.101211](https://doi.org/10.1016/j.addma.2020.101211).
- Xiao, Q., A. Mandal, C. D. Lin, and X. Deng. 2021. EZGP: Easy-to-interpret Gaussian process models for computer experiments with both quantitative and qualitative factors. *SIAM/ASA Journal on Uncertainty Quantification* 9 (2):333–53. doi: [10.1137/19M1288462](https://doi.org/10.1137/19M1288462).
- Xiao, Q., Y. Wang, A. Mandal, and X. Deng. 2024. Modeling and active learning for experiments with quantitative-sequence factors. *Journal of the American Statistical Association* 119 (545):407–21. doi: [10.1080/01621459.2022.2123335](https://doi.org/10.1080/01621459.2022.2123335).