Tutorial overview of simple, stratified, and parametric bootstrapping

P. M. Shankar

Department of Electrical and Computer Engineering, Drexel University, Philadelphia, Pennsylvania

Correspondence
P. M. Shankar, Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA 19104. Email: shankapm@drexel.edu

Abstract
Students pursuing baccalaureate degrees in electrical engineering and computer engineering are required to take a course in probability and statistics. While the course continues to be mostly conceptual, author started initiatives to introduce data analytics in this course with special emphasis on machine vision applications. Topics such as receiver operating characteristics curves and hypothesis testing are covered through examples and exercises with students having individual datasets. Continuing with this theme, bootstrapping and associated methodologies have now been introduced to facilitate interpretation of machine vision experiments. A demo created that illustrates simple, stratified, and parametric bootstrapping as a means to understand the statistics of a machine vision sensor is presented. It encompasses a number of conceptual topics such as random variables, densities, parameter estimation, chi square testing, etc. alongside data analytics offering a holistic picture of machine learning and machine vision to the undergraduate students.

Keywords
bootstrapping, chi square tests, receiver operating characteristics, resampling, tutorial

1 | INTRODUCTION

A course in engineering probability is a requirement for the successful completion of the baccalaureate programs in engineering.1 This traditional course covering Bayes’ rule and associated topics along with single and multiple random variables is very conceptual and theoretical.2,3 Coupled with the fact that students generally do not see exercises and examples involving data, the course becomes very challenging and less interesting to students as they do not see any direct link to applications in business, industry, and medicine, specifically in the areas of robotics, machine learning, and medical diagnostics. One way to bridge the gap between the conceptual and practical coverage is to incorporate data analysis and interpretation in the course. At Drexel University, author started the initiative by introducing topics of confusion and transition matrices, receiver operating characteristics (ROC) curves, area under the ROC curve (AUC), and parametric hypothesis testing through the analysis of machine vision data.4-6 Every student was provided with a unique dataset for these assignments. Bootstrapping, which involves sampling with replacement, in its very elemental form, was also introduced as a means to examine the statistics of the mean of a set of measurements along with an assignment.7-9
Bootstrapping involving two cohorts of data as in the case of the ROC analysis of a machine vision receiver needs a much more detailed approach. Pedagogy and didactics of bootstrapping of AUC are not generally covered in courses because of the reliance on software packages to carry out bootstrapping. Coverage of bootstrapping is also missing from the standard textbooks used in engineering probability courses. In addition, literature is abound with terms such as stratified bootstrapping and parametric bootstrapping making it necessary to offer a detailed pedagogical perspective of bootstrapping so that it is well understood. The applications of bootstrapping can be found in many areas of science, engineering, medicine, social sciences, etc. It is essential that students be exposed to the details of bootstrapping so that it becomes easy to transition to undertake statistical analysis when they enter their chosen professions.

This manuscript reports on a demo created for the students in an undergraduate course in electrical engineering and computer engineering probability at Drexel University.

2 | BACKGROUND

Inferences and predictions are often made on the basis of a single experiment. Of specific interest in machine learning problems is the ability to predict the presence or the absence of a target in front of a machine vision sensor based on the data collected. This is accomplished by conducting an experiment which consists of collecting data from a sensor when a target is in its field of view as well as collecting data when the target is absent. The sensor may be wireless, radio frequency, infrared, or ultrasonic. In all these cases, because of the uncertainties either from multipath scattering or other forms of unwanted interference, the data from the two sets, target present and target absent, will overlap. A statistical approach is needed so that one can have an optimum threshold (value of the sensor output) for decision making later to predict whether the target is present or absent on the basis of a single observation. In another instance, the efficacy of a new medical screening device may need to be tested. A number of subjects (having no illness and having illness) are recruited and screening is administered with the goal determining an optimal threshold to predict (on the basis of a single screening) whether the subject suffers from the illness or not. In both cases, the ROC quantified through the metric, AUC is used to establish the performance of the screening device or the sensor. The range of the AUC lies between 0.5 and 1, the ideal sensor with no errors in prediction corresponding to the value of AUC of unity. In practical cases, AUC values never reach unity and there is a need to know the statistics of AUC to access the efficacy. Very few formulaic tools are available to examine the statistics of the AUC to draw inference on its variability (variance) and its 95% confidence level. If there is a way to undertake the experiment multiple times, we may be able to study the inferential statistics. Because of difficulties in recruiting large number of subjects available for screening or the high costs of setting up machine vision testing multiple times, we need to explore other ways to study the efficacy of the devices and sensors.

A direct way to reproduce the experimental results is through the use of resampling. Bootstrapping is a simple technique allowing empirical regeneration of samples of the outcomes of the experiment through continual resampling with replacement regardless of the underlying statistics of the data. While resampling is relatively simple, its application to the ROC is not straightforward because of the existence of two cohorts of data. The two cohorts of data (one from the target present and the other from the target absent) also offer two ways of resampling. One is to resample the composite data from the two cohorts constituting simple bootstrapping while retaining their labels or tags (target absent and target present) or resample the individual cohorts separately constituting stratified bootstrapping. These two forms of bootstrapping are nonparametric because we are not estimating any specific parameters associated with the data. If we know the underlying statistics of the two cohorts, we may generate multiple sets of samples satisfying the underlying statistics. In other words, we can reproduce the values of data through random number simulations. The creation of datasets based on underlying statistics constitutes the technique of parametric bootstrapping even though no resampling is used. Parametric bootstrapping may also address one of the issues associated with traditional bootstrapping, namely the existence of nonindependent samples arising from sampling with replacement.

A number of publications offer insight into bootstrapping. But, very few offer a detailed step-by-step approach that can be followed by an undergraduate student taking the first course in probability. As the students are starting to understand the concepts of data analytics through examples and exercises dealing with terms such as confusion matrices, probabilities of false alarm and miss, ROC etc. and exposure to another new methodology requires a careful and detailed approach. The implementation of bootstrapping and the nuanced differences among the three forms of bootstrapping must be undertaken at a very basic level making it possible for them to appreciate its practical implications. The ideal time for demonstrating this to the students may be when they have been exposed a number of data analytic applications.
Students were familiar with data analytics because they were required to solve one homework problem in Matlab (www.mathworks.com) every week with a unique dataset for every student (besides a set of common problems for the class). Students had already completed exercises on positive predictive values, confusion matrix, AUC, chi square testing involving multiple densities to determine the best fit, maximum likelihood estimation of parameters of densities, and statistical analysis of improvement in performance achieved through signal processing algorithms, namely arithmetic mean, geometric mean, maximum.2,3,29 The topic of bootstrapping was introduced after students were exposed to mathematical statistics (marginal, conditional, joint probabilities, and Bayes’ rule), one and two random variables, parametric hypothesis testing (chi square tests), ROC curves, etc.

At Drexel University, every quarter consists of 10 weeks of classes followed by examinations during the 11th week. The engineering probability course is offered every quarter (required course for students pursuing baccalaureate degrees in electrical engineering as well as computer engineering) as a 4-credit course with 3 hours of lecture followed by 1 hour of recitation every week. For the recitation sessions, the class is split into smaller sections (less than 30 students). Lectures and recitations (three different nonoverlapping sessions) were covered by the author while teaching assistants were responsible for grading the homework submissions. For every topic covered in data analytics, demos were created and students were provided with datasets ahead of time so that they could follow the procedure during the lecture.2,3

The demo created is described next. It was implemented in Matlab (version 2019a).

### 3 | CREATION OF THE DEMOS AND RESULTS

A simple model of a machine vision sensor is used in the demo with samples of data collected with a target present in the field of view of a sensor and target absent in the field of view. The dataset used in the demo is shown in Table 1. It consisted of \( N_0 = 70 \) samples (target absent: hypothesis \( H_0 \)) and \( N_1 = 60 \) samples (target present: hypothesis \( H_1 \)) of the data collected.

| Target absent | 1.065 | 1.4527 | 1.2136 | 0.5438 | 0.9607 | 1.4839 | 2.4966 |
|---------------|-------|--------|--------|--------|--------|--------|--------|
| 1.848         | 3.4215| 0.9317 | 2.7098 | 1.4702 | 2.7796 | 1.2094 |
| 1.4237        | 1.9893| 0.3166 | 2.9643 | 1.1842 | 0.6276 | 2.7992 |
| 1             | 1.5359| 0.8195 | 2.3391 | 3.3296 | 0.3461 | 2.3408 |
| 1.1051        | 1.1607| 2.116  | 3.1226 | 3.4398 | 1.5508 | 0.9616 |
| 1.4548        | 0.988 | 1.958  | 3.2232 | 0.994  | 1.6618 | 3.8522 |
| 2.2179        | 2.6836| 0.9415 | 0.5999 | 3.1532 | 1.206  | 1.311  |
| 0.1206        | 1.1292| 1.0089 | 1.9378 | 2.4539 | 0.9694 | 0.7312 |
| 0.7583        | 1.5243| 0.988  | 2.212  | 1.4416 | 0.878  | 1.2381 |
| 0.5812        | 0.5502| 0.3405 | 0.9917 | 0.4564 | 1.7705 | 2.0065 |

| Target present | 2.3833 | 2.2629 | 4.2882 | 3.447 | 0.9503 | 2.0914 |
|----------------|--------|--------|--------|-------|--------|--------|
| 3.8669         | 4.1612 | 2.7489 | 4.3879 | 2.7874| 4.4084 |
| 3.1586         | 6.364  | 2.0118 | 4.3661 | 7.1283| 2.0643 |
| 2.8864         | 2.0157 | 2.3335 | 1.0294 | 6.0826| 1.2949 |
| 4.2997         | 3.3567 | 6.8318 | 2.3034 | 1.6827| 7.8573 |
| 3.38           | 3.2767 | 3.8012 | 3.7993 | 0.9925| 5.0523 |
| 3.0062         | 2.757  | 3.1757 | 5.9732 | 3.15  | 3.4679 |
| 2.4024         | 0.3027 | 3.3986 | 4.7591 | 2.6701| 4.9292 |
| 4.2595         | 1.6926 | 2.312  | 1.4589 | 1.2391| 4.5932 |
| 4.0767         | 2.8594 | 1.9243 | 1.7278 | 6.2024| 3.5736 |
The generation of this dataset is described elsewhere. The datasets are merged into a single column, with the top 70 values belonging to hypothesis H₀ and the rest belonging to hypothesis H₁. This step is followed by concatenating a second column of containing the labels with ‘0’ for H₀ and ‘1’ for H₁. This means that we have a new matrix, [data, labels] of size [N × 2], the first column of the data (values) and the second column with the corresponding labels. The total number of samples is N = N₀ + N₁. We can examine the ROC of the machine vision sensor to characterize its performance in terms of the AUC. With the results from a single experiment, we need to use techniques such as bootstrapping that implies sampling of the data with replacement to properly understand and interpret the statistics of AUC. Bootstrapping thus allows replication of experiments. In the absence of the two cohorts (target absent and target present) of data, bootstrapping is simple and straightforward with sampling the data with replacement and a single boot sample (or boot set) is created by sampling N times if N is the number of original samples of a single cohort. By repeating the resampling process M times, we can get M boot sets (each being a matrix of size [N × 1] that may be used to study the characteristics of the original cohort, for example, the statistics of the mean. Students had already seen similar analysis with a single cohort of data in lecture and completed an assignment to estimate the statistics of the mean of a measurement. But, the machine vision system has two cohorts of data, each distinctly identified by two labels. This means that the simple resampling of the composite or pooled data from two cohorts set is not sufficient.

Bootstrapping of the machine vision data is undertaken on the basis of rows of the [N × 2] matrix. This means that resampling results in the selection of a row of data (value) and the corresponding label. The sampling with replacement is undertaken M times creating M boot sets, each resulting in a matrix of size [N × 2]. It can easily be understood that this sampling may result in more than the original sample sizes of cohort H₁ or H₀ because resampling is nothing but choice of a number between 1 and N (repetition allowed). Resampling may be done in Matlab directly using the command `randsample(.)`. It can also be done using a uniform random number generator as

\[ \overrightarrow{k} = \text{ceil}(\text{rand}(1, N) \times N). \]  

\[ (1) \]

In Equation (1), `rand(1,N)` generates N uniform random number in [0,1] and `ceil(.)` produces the integer value (maximum) and \( \overrightarrow{k} \) is an [N × 1] vector of integers between 1 and N (130 in the present case) with repetitions. Each of these M boot sets is used to obtain the AUC providing M boot samples of the AUC. Indeed, this means that we expect to get the statistics of the AUC from the 5000 boot samples of the AUC.

One of the issues raised with regards to bootstrapping described above (also identified as the simple or direct bootstrapping) is the fact that the relative counts of the two cohorts vary with every bootstrap set as explained in the previous paragraph because resampling is undertaken on the [N × 2] matrix. One way to mitigate the variation in relative counts is to perform stratified bootstrapping. This means that bootstrapping is done for each cohort. The resampling is done first on the [N₀ × 2] matrix and then on the [N₁ × 2] matrix. This means that we have two vectors,

\[ \overrightarrow{k}_0 = \text{ceil}(\text{rand}(1, N_0) \times N_0) \]

\[ \overrightarrow{k}_1 = \text{ceil}(\text{rand}(1, N_1) \times N_1). \]  

\[ (2) \]

In Equation (2), \( \overrightarrow{k}_0 \) is an [N₀, 1] vector, with integers between 1 and N₀ and \( \overrightarrow{k}_1 \) is an [N₁, 1] vector, with integers between 1 and N₁. With each cohort, M = 5000 boot samples (sets) are created. These sets result in M samples of the AUC are used for the analysis just as in the case of simple bootstrapping. It is clear that with stratified bootstrapping, the number of samples of target absent cohort will always be N₀ and the other cohort will always be N₁, identical in each case to the original dataset.

While the simple and stratified bootstrapping presented above do not require any knowledge of the underlying statistics of the data, we may explore another option if the statistics of the two cohorts are available. This is accomplished in two steps. First a goodness fit test is conducted on each cohort. Students had already undertaken parametric hypothesis tests as a part of the homework assignments. Since both sets of datasets represented either power or amplitude values collected through the sensor, the hypothesis tests were conducted to see the closest density fit to the data (significance level of 5%) among Rayleigh, Rician, gamma, Nakagami, and Weibull densities. Once the density fits are obtained, M sets of each cohort (target absent and target present) were created and in each case, ROC analysis was carried out providing M samples of AUC. This constitutes parametric bootstrapping even though the M sets were not obtained through traditional bootstrapping modality.
As mentioned above, chi square tests were carried out first on each of the datasets. Figure 1 displays the histograms of the data obtained using the `ksdensity(.)` command in Matlab and the closest fit based on the chi square test along with the parameters (\(a\) and \(b\)) of the gamma density,

\[
f(x|H_j) = \frac{x^{a_j-1}}{(b_j)^{a_j}\Gamma(a_j)} \exp \left( -\frac{x}{b_j} \right), \quad x \geq 0, \quad j = 0, 1.
\] (3)

In Equation (3), \(\Gamma(.)\) is the gamma function.\(^3,15\) The parameters were estimated using `fitdist(.)` and chi square testing was undertaken using `chi2gof(.)` in Matlab. Figure 2 also displays the test statistic (\(X^2_T\)) and the degrees of freedom associated with the test.\(^3\)

Figure 2 shows the ROC curves for the data and the ROC curve obtained from the theoretical density fits (in this case, both fits are gamma). For any arbitrary threshold \(x_T\), the probability of false alarm (\(P_F\)), and probability of detection (\(P_D\)) are

\[
P_F = \int_{x_T}^{\infty} f(x|H_0)dx.
\] (4)

\[
P_D = \int_{x_T}^{\infty} f(x|H_1)dx.
\] (5)
The theoretical ROC plot is obtained from calculating the probabilities in Equation (4) by varying the threshold $x_T$ from 0 to $\infty$. The AUC can be obtained either using `trapz(.)` or `polyarea(.)` commands in Matlab. While AUC may be obtained using `perfcurve(.)` in Matlab, a simple script was written to accomplish this in the demo. One can see that the two curves fit well with AUC values very close. Even though we only have a single set of data, it is possible to estimate the SD of the AUC obtained from the data using the formula in literature as

$$\sigma(AUC) = \sqrt{\frac{(1 - AUC)AUC + (N_1 - 1)(A_1 - AUC^2) + (N_0 - 1)(A_2 - AUC^2)}{N_1 N_0}}. \quad (6)$$

In Equation (6),

$$A_1 = \frac{AUC}{2 - AUC}. \quad (7)$$

$$A_2 = \frac{2AUC^2}{1 + AUC}. \quad (8)$$

The value of the SD is also shown in Figure 2. Figure 3 shows the summary results of bootstrapping. Top portion displays the histogram of the boot samples of AUC (with a Gaussian fit) for the case of simple bootstrapping. The mean ($\mu$), SD ($\sigma$), and the 95% confidence interval are also given. The middle portion displays the results for the case of stratified bootstrapping. The characteristics of both plots (simple and stratified) appear to be matching.

The variation in AUC values with both simple and stratified bootstrapping appears to be in the same range as reflected in the matching statistics in both cases (top two displays). The results of parametric bootstrapping are displayed on the bottom. The statistics are not much different from those of the simple bootstrapping (top left) or stratified bootstrapping (top right).

To determine whether the data sizes influence the results of bootstrapping, the procedure was repeated for another set (this was also provided to the students). The dataset is shown in Table 2.

The dataset consists of 40 samples of target absent and 30 samples of target present. The best fits were Nakagami and Rayleigh as seen in Figure 4. The two densities are

$$f(x) = \frac{2}{\Gamma(m)} \left(\frac{m}{\Omega}\right)^m x^{2m-1} \exp\left(-\frac{m x^2}{\Omega}\right), \quad x \geq 0, m \geq \frac{1}{2}, \Omega > 0. \quad (9)$$

$$f(x) = \frac{x}{b^2} \exp\left(-\frac{x^2}{2b^2}\right), \quad x \geq 0, b > 0. \quad (10)$$
**TABLE 2** Dataset used

| Target absent         | 4.1265 | 3.1262 | 4.545 | 2.47 |
|-----------------------|--------|--------|-------|------|
|                       | 3.0827 | 1.7475 | 1.4253| 2.7553|
|                       | 1.2586 | 2.0172 | 1.9097| 0.9492|
|                       | 0.571  | 3.8455 | 4.2976| 1.5789|
|                       | 0.1814 | 2.2919 | 3.9989| 2.4109|
|                       | 3.0508 | 3.0734 | 1.5875| 2.0464|
|                       | 0.9669 | 1.8482 | 1.2809| 5.5544|
|                       | 0.6888 | 2.3737 | 2.3115| 3.005 |
|                       | 0.1599 | 2.6668 | 1.7248| 2.9118|
|                       | 2.0744 | 1.9203 | 0.6482| 1.9811|

| Target present        | 4.2917 | 5.7518 | 6.8372|
|-----------------------|--------|--------|-------|
|                       | 9.7598 | 6.8042 | 2.1683|
|                       | 9.6515 | 6.7231 | 4.6538|
|                       | 6.4053 | 13.6268| 15.9081|
|                       | 4.8132 | 2.1271 | 2.4463|
|                       | 10.5349| 3.1468 | 7.4671|
|                       | 11.8775| 3.8767 | 4.229 |
|                       | 4.2091 | 5.6213 | 10.4387|
|                       | 3.5992 | 8.4665 | 6.0601|
|                       | 5.1248 | 2.7738 | 4.4685|

**FIGURE 4** Histograms of the data and the best density fits associated with the data in Table 2

Equation (9) is the Nakagami density with the Nakagami parameter $m$ and second moment of $\Omega$. Equation (10) is the Rayleigh density with the parameter $b$. Figure 5 shows three ROC plots. In addition to the ROC from the data and the best fit, Figure 6 shows the ROC curve with a bigamma fit by treating both sets as gamma distributed. The lower sample size clearly leads to slightly different AUC values for each of the three cases, a departure from the results in Figure 2. Figure 6 shows the results of bootstrapping with the dataset in Table 2. The problems with parametric bootstrapping (more appropriately random number simulations) are seen from the departure of the AUC compared to the simple and
4 | DISCUSSION AND CONCLUSIONS

A demo was created to familiarize students with applications of probability and statistics to data analytics. The goal was to expand the scope of the traditional course in engineering probability to provide insight into machine vision applications to gain a better understanding of the current trends. While exploring data analytics is important, the association between it and the conceptual topics in statistics should not be ignored. The pedagogic and didactic strengths of demo described in this manuscript are that it offers a means to combine multiple topics of interest generally covered in a statistics course on probability and random variables to the analysis and interpretation of data. The concepts of hypothesis testing, probability density, and distribution functions of random variables, parameter estimation, probabilities of false alarm and miss based on models are reinforced in the demo. Bootstrapping, more specifically, parametric bootstrapping bridges the gap between topics in probability and applications to machine vision and machine learning. The use of bootstrapping as a
computational tool allows us to replicate the experiments. Bootstrapping also makes it possible to evaluate the efficacy of the machine vision system quantified through the AUC.

Thus, the panoply of the topics encompassed in this demo allows the students to understand the importance of the topics covered in the lecture to practical aspects in engineering. In addition, extensive use of computational approaches in the demo provides a window into importance of software packages and computational tools in engineering education. Together with other efforts reported by the author and the students’ responses, all the demos including the one described here, made the course relevant to the current topics in engineering applications to industry and medicine. Starting with fall 2019 to 2020, the course is being renamed as Probability and Data Analytics for Engineers, taking into account the importance of data analytics in probability as evident from the demos and exercise problems introduced in the past 2 years. During the upcoming academic year, author also plans to expand the demos to include tests such as $z$ tests, $t$ tests, and Wilcoxon rank sum tests to expose the students to other practical aspects of engineering probability.

CONFLICT OF INTEREST
Author has no conflicts of interest relevant to this article.

ORCID

P. M. Shankar  https://orcid.org/0000-0002-9719-9700

REFERENCES

1. ABET (2019), ABET Accreditation. Baltimore, MD: Accreditation Board for Engineering and Technology, Inc. www.abet.org/accreditation/
2. Shankar PM. Pedagogy of Bayes’ rule, confusion matrix, transition matrix, and receiver operating characteristics. Comput Appl Eng Educ. 2019;27:510-518.
3. Shankar PM. Pedagogy of chi square goodness of fit test for continuous distributions. Comput Appl Eng Educ. 2019;27:679-689.
4. Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology. 1982;143:29-36.
5. Fawcett T. An introduction to ROC analysis. Pattern Recogn Lett. 2006;27:861-874.
6. Kohavi R, Provost F. Glossary of terms. Mach Learn. 1998;30:271-274.
7. Efron B, Tibshirani R. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. Stat Sci. 1986;1:54-75.
8. Hu N, Dannenberg RB. Bootstrap learning for accurate onset detection. Mach Learn. 2006;65:457-471.
9. Wehrens R, Putter H, Buydens LM. The bootstrap: a tutorial. Chemom Intel Lab Syst. 2000;54(1):35-52.
10. Bertail P, Clémençon SJ, Vayatis N. On bootstrapping the ROC curve. Paper presented at: Proc of the 21st International Conf on Neural Information Processing Systems (NIPS2008); 2008;137–144.
11. Hesterberg TC. What teachers should know about the bootstrap: resampling in the undergraduate statistics curriculum. Am Stat. 2015;69:371-386.
12. Hardin J, Hoerl R, Horton NJ, Nolan D. Data science in statistics curricula: preparing students to think with data. Am Stat. 2015;69:343-353.
13. Baumer B. A data science course for undergraduates: thinking with data. Am Stat. 2015;69:334-342.
14. Yates RD, Goodman DJ. Probability and Stochastic Processes: A Friendly Introduction for Electrical and Computer Engineers. Hoboken, NJ: Wiley; 2014.
15. Papoulis A, Pillai SU. Probability, Random Variables and Stochastic Processes. 4th ed. McGraw-Hill; 2002.
16. Hall P, Maiti T. On parametric bootstrap methods for small area prediction. J R Stat Soc B Stat Methodol. 2006;68(2):221-238.
17. Pons O. Bootstrap of means under stratified sampling. Electron J Stat. 2007;1:381-391.
18. Hesterberg T. Bootstrap. Wiley Interdisciplinary Reviews: Computational Statistics. 2011;3(6):497-526.
19. Bradley AP. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn. 1997;30:1145-1159.
20. Swets JA. ROC analysis applied to the evaluation of medical imaging techniques. Invest Radiol. 1979;14:109-121.
21. Brown CD, Davis HT. Receiver operating characteristics curves and related decision measures: a tutorial. Chemom Intel Lab Syst. 2006;80:24-38.
22. Mecatti F. Bootstrapping unequal probability samples. Stat Appl. 2000;12:67-77.
23. Liu H, Li G, Cumberland WG, Wu T. Testing statistical significance of the area under a receiving operating characteristics curve for repeated measures design with bootstrapping. J Data Sci. 2005;3:257-278.
24. Liu X. Classification accuracy and cut point selection. Stat Med. 2012;31:2676-2686.
25. Datta J, Ghosh JK. Bootstrap—an exploration. Stat Methodol. 2014;20:63-72.
26. Bickel PJ, Freedman DA. Asymptotic normality and the bootstrap in stratified sampling. Ann Stat. 1984;12(2):470-482.
27. Engel J. On teaching bootstrap confidence intervals. Paper presented at: Proc. 10th International Conference on Teaching Statistics; 2010; Voorburg, The Netherlands: International Statistics Institute.
28. Preston J. Rescaled bootstrap for stratified multistage sampling. Surv Methodol. 2009;35(2):227-234.
29. Shankar PM. Pedagogy of diversity and data analytics: theory to practice. Comput Appl Eng Educ. 2019;27:1227-1285.
P. M. Shankar received his MSc (1972) in Physics from Kerala University, India, MTech (1975) in Applied Optics, and PhD in Electrical Engineering (1980) from the Indian Institute of Technology, Delhi, India. He was a visiting scholar at the School of Electrical Engineering, University of Sydney, Australia, from 1981 to 1982. He joined Drexel University in 1982 and is currently the Allen Rothwarf Professor of Electrical and Computer Engineering. He is the author of the textbook “Introduction to Wireless Systems,” published by John Wiley & Sons, 2002 and “Fading and Shadowing in Wireless Systems,” published by Springer, 2012 (First Ed.) and 2017 (second Ed.). He also authored the book, “Differential Equations: A Problem Solving Approach Based on MATLAB,” published by CRC Press, April 2018. His interests are in statistical signal processing for medical applications, modeling of fading channels, wireless communications, medical ultrasound, optical fiber biosensors, Matlab for pedagogy, etc.

How to cite this article: Shankar PM. Tutorial overview of simple, stratified, and parametric bootstrapping. Engineering Reports. 2020;2:e12096. https://doi.org/10.1002/eng2.12096