MSG400-TMS150

Stochastic data processing and simulation 2024

The Bootstrap for simulation-based uncertainty quantification

The exercises 1-2 constitute the assignment A2b. Notice the "recommended deadline" on the course webpage. The A2b assignment must be handed in as a LaTeX report and be submitted to Canvas together with the corresponding Matlab code.

Please use the recommended report template provided on the course page. You must also upload separately from the report all the code you produced. And you must also embed the code inside the report, as illustrated in the report template found on the course webpage. Recall that each submitted report and the code therein is INDIVIDUAL not group work.

Finally, since you have to write a proper report: as from the provided template, you are also asked to produce some background on the methodology you use. So do not just write answers to the exercise questions. See `https://chalmers.instructure.com/courses/31060/pages/guidelines-for-report-writing` for guidelines on report writing.

The report should not be longer than 10 pages including figures, but excluding appendices. Figures and axes labels should be big enough to be readable if printed. It is OK to use colors. For a given project report, 0.5 points will be deducted if the report is not clearly structured or is otherwise hard to understand. Likewise, 0.5 points will be deducted if the code attached to the report is not properly structured and commented.

Full details on grading are on the course webpage.

At the end of the document you find a short Matlab primer. Also see the file `demo_matlab.m` on Canvas.

Some of the sections that follow are denoted with an asterisk * when the topic can be skipped without much loss (those are left for the interested reader).

1 Introduction

In the previous two lectures we have focussed on the selection of appropriate linear regression models, their parameters estimation and the assessment of the estimates uncertainty. However, the estimation of parameters in general models and the assessment of the estimates' variability, is not a problem exclusively pertaining regression models (linear or nonlinear). In fact, the estimation of parameters in probability distributions is a central problem in statistics that one

tends to encounter already during the very first course on the subject. More generally, we are interested in inferences for *unknowns*: these are not just model parameters, for example when we predicted new observations, clearly those are unobserved quantities. Along with the estimate of some unobserved quantity we are (we should be!) also interested in its accuracy, which can be described in terms of the bias and the variance of the estimator, as well as confidence intervals around it. Sometimes, such measures of accuracy can be derived analytically. Often, they can not. The *bootstrap* is a technique that can be used to estimate them numerically[1].

2 The general idea

Let $X_1, ..., X_n$ be a i.i.d. sample from distribution F, that is $\mathbf{P}(X_i \leq x) = F(x)$, and let $X_{(1)}, ..., X_{(n)}$ be the corresponding ordered sample. For the purpose of this introduction, suppose we are interested in some *scalar* parameter θ which is associated with this distribution (mean, median, variance etc), the treatment of a multivariate θ being analogous. There is also an estimator $\hat{\theta} = t(\{X_1, ..., X_n\})$, with t denoting some function, that we can use to estimate θ from data. In this setting, it is the deviation of $\hat{\theta}$ from θ that is most interesting.

Ideally, to get an approximation of the estimator distribution we would like to repeat the data-generating experiment, say, B times, calculating $\hat{\theta}$ for each of the B data sets. That is, we would draw B samples of size n from the true distribution F (with replacement, if F is discrete). In practice, this is impossible.

The bootstrap method is based on the following simple idea: *Even if we do not know F we can approximate it from data and use this approximation, $\hat{F}$, instead of F itself.*

This idea leads to several flavours of bootstrap that deviate in how, exactly, the approximation $\hat{F}$ is obtained. Two broad areas are the *parametric* and *non-parametric* bootstrap.

The non-parametric estimate is the so called empirical distribution (you will see the corresponding pdf if you simply do a histogram of the data), that can be formally defined as follows:

Definition 2.1. *With # denoting the number of members of a set, the* empirical distribution function $\hat{F}$ *is given by*

$$\hat{F}(x) = \frac{\#\{i : X_i \leq x\}}{n} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}\{X_i \leq x\},$$

$$\mathbb{I}\{X_i \leq x\} = \begin{cases} 1 & \text{if } X_i \leq x \\ 0 & \text{otherwise} \end{cases}$$

That is, it is a discrete distribution that puts mass $1/n$ on each data point in your sample.

The parametric estimate assumes that the data comes from a certain distribution family (Normal, Gamma etc). That is, we say that we know the general functional form of the pdf, but not the exact parameters. Those parameters can then be estimated from the data (typically with Maximum Likelihood) and plugged in the pdf to get $\hat{F}$. This estimation method leads to

[1] More generally, the bootstrap is a method of approximating the distribution of functions of the data (statistics), which can serve different purposes, among others the construction of CI and hypothesis testing.

more accurate inference if we guessed the distribution family correctly but, on the other hand, $\hat{F}$ may be quite far from F if the family assumption is wrong.

2.1 Algorithms

The two algorithms below describe how the bootstrap can be implemented.

Non-parametric bootstrap
Assuming a data set $x = (x_1, ..., x_n)$ is available.

1. Fix the number of bootstrap re-samples B. Often $B \in [1000, 2000]$.
2. Sample a new data set x^* set of size n from x *with replacement* (this is equivalent to sampling from the empirical cdf $\hat{F}$).
3. Estimate θ from x^*. Call the estimate $\hat{\theta}_1^*$. Store.
4. Repeat step 2 and 3 further $B - 1$ times.
5. Consider the empirical distribution of $(\hat{\theta}_1^*, ..., \hat{\theta}_B^*)$ as an approximation of the true distribution of $\hat{\theta}$.

You may produce the histogram of $(\hat{\theta}_1^*, ..., \hat{\theta}_B^*)$. This histogram represents the pdf of $\hat{F}$.

Parametric bootstrap
Assuming a data set $x = (x_1, ..., x_n)$ is available.

1. Assume that the data comes from a known distribution family F_ψ described by a set of parameters ψ (for a Normal distribution $\psi = (\mu, \sigma)$ with μ being the expected value and σ the standard deviation).
2. Estimate ψ with, for example, Maximum likelihood, obtaining the estimate $\hat{\psi}$.
3. Fix the number of bootstrap samples B. Often $B \in [1000, 2000]$.
4. Sample a new data set x^* set of size n from $F_{\hat{\psi}}$.
5. Estimate θ from x^*. Call the estimate $\hat{\theta}_1^*$. Store.
6. Repeat 4 and 5 further $B - 1$ times.
7. Consider the empirical distribution of $(\hat{\theta}_1^*, ..., \hat{\theta}_B^*)$ as an approximation of the true distribution of $\hat{\theta}$.

Again, you may produce the histogram of $(\hat{\theta}_1^*, ..., \hat{\theta}_B^*)$, this representing the pdf of $\hat{F}$.

Concretely, let us say that $X_i \sim N(0, 1)$, θ is the median and it is estimated by $\hat{\theta} = X_{(n/2)}$, the $n/2$-th element in the ordered sequence. In Figure 1 the distribution of $\hat{\theta}$ approximated with the non-parametric and parametric bootstrap is plotted. Note that the parametric distribution is smoother than the non-parametric one, since the samples were drawn from a continuous distribution.

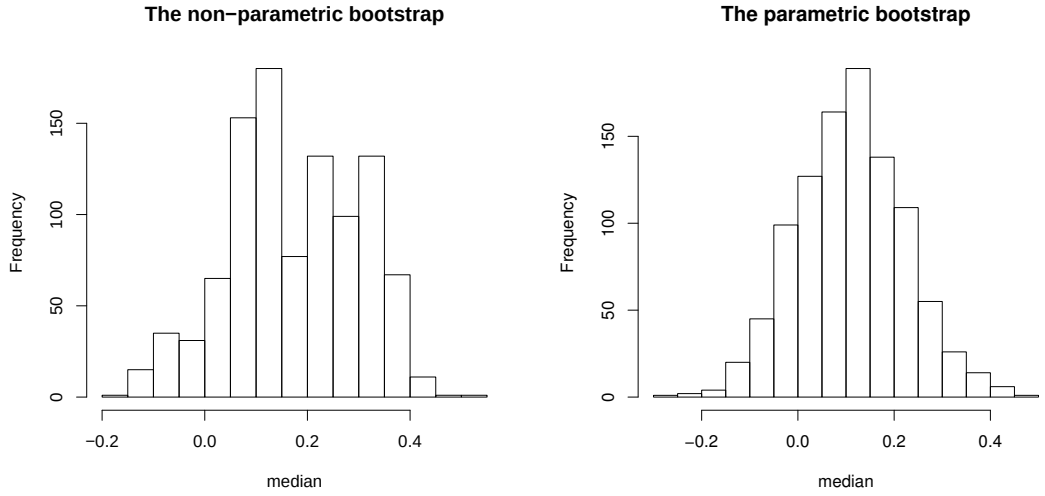


Figure 1: The non-parametric and the parametric bootstrap distribution of the median, $B = 1000$.

3 Bias and variance estimation (* can be skipped, only for those interested)

The theoretical bias and variance of an estimator $\hat{\theta}$ are defined as

$$\mathbf{Bias}(\hat{\theta}) = \mathbf{E}[\hat{\theta} - \theta] = \mathbf{E}[\hat{\theta}] - \theta$$
$$\mathbf{Var}(\hat{\theta}) = \mathbf{E}\left[(\hat{\theta} - \mathbf{E}[\hat{\theta}])^2\right]$$

In words, the bias is a measure of a systematic error ($\hat{\theta}$ tends to be either smaller of larger than θ) while the variance is a measure of random error.

In order to obtain the bootstrap estimates of bias and variance we plug in the original estimate $\hat{\theta}$ (which is a constant given data) in place of θ and $\hat{\theta}^*$ (the distribution of which we get from bootstrap) in place of $\hat{\theta}$. This leads us to the following approximations:

$$\mathbf{Bias}(\hat{\theta}) \approx \frac{1}{B}\sum_i \hat{\theta}_i^* - \hat{\theta} = \bar{\hat{\theta}}^* - \hat{\theta}$$

$$\mathbf{Var}(\hat{\theta}) \approx \frac{1}{B-1}\sum_i (\hat{\theta}_i^* - \bar{\hat{\theta}}^*)^2$$

That is, the variance is, as usual, estimated by the sample variance (but for the bootstrap sample of $\hat{\theta}$) and bias is estimated by how much the original $\hat{\theta}$ deviates from the average of the bootstrap sample denoted $\bar{\hat{\theta}}^*$.

4 Confidence intervals

There are several methods for CI construction with Bootstrap, the most popular being "normal", "basic" and "percentile". Let $\hat{\theta}^*_{(i)}$ be the ordered bootstrap estimates, with $i = 1, ..., B$ indicating the different samples. Let α be the significance level. In all that follows, $\hat{\theta}^*_\alpha$ will denote the

α-quantile of the distribution of $\hat{\theta}^*$. You can approximate this quantile with $\hat{\theta}^*_{([(B+1)\alpha])}$ with the $[x]$ indicating some interpolation or rounding procedure[2].

Basic	$[2\hat{\theta} - \hat{\theta}^*_{1-\alpha/2}, 2\hat{\theta} - \hat{\theta}^*_{\alpha/2}]$
Normal	$[\hat{\theta} - z_{1-\alpha/2}\hat{se}, \hat{\theta} - z_{\alpha/2}\hat{se}]$
Percentile	$[\hat{\theta}^*_{\alpha/2}, \hat{\theta}^*_{1-\alpha/2}]$

with z_α denoting an α quantile from a Normal distribution and $\hat{se}$ the estimated standard deviation of $\hat{\theta}$ calculated from the bootstrap sample.

Basic CI (* can be skipped, only for those interested)

To obtain this confidence interval we start with $W = \hat{\theta} - \theta$ (compare to the classic CI that correspond to a t-test). If the distribution of W was known, then a two-sided CI could be obtained by considering $\mathbf{P}(w_{\alpha/2} \leq W \leq w_{1-\alpha/2}) = 1 - \alpha$, which leads to CI $= [l_{low}, l_{up}] = [\hat{\theta} - w_{1-\alpha/2}, \hat{\theta} - w_{\alpha/2}]$. However, the distribution of W is not known, and is approximated with the distribution for $W^* = \hat{\theta}^* - \hat{\theta}$, with w^*_α denoting the corresponding α-quantile. The CI becomes $[\hat{\theta} - w^*_{1-\alpha/2}, \hat{\theta} - w^*_{\alpha/2}]$. Noting that $w^*_\alpha = \hat{\theta}^*_\alpha - \hat{\theta}$ and substituting this expression in the CI formulation leads to the definition given in the box.

This interval construction relies on an assumption about the distribution of $\hat{\theta} - \theta$, namely that it is independent of θ. This assumption, called pivotality, does not necessarily hold in most cases. However, the interval gives acceptable results even if $\hat{\theta} - \theta$ is close to pivotal.

Normal CI (* can be skipped, only for those interested)

This confidence interval probably looks familiar since it is almost an exact replica of the commonly used confidence interval for a population mean. Indeed, similarly to that familiar CI, we can get it if we have reasons to believe that $Z = (\hat{\theta} - \theta)/\hat{se} \sim N(0, 1)$ (often true asymptotically as the data size $n \to \infty$). Alternatively, we can again consider $W = \hat{\theta} - \theta$, place the restriction that it should be normal and estimate its variance with bootstrap. Observe that the normal CI also implicitly assumes pivotality.

Percentile CI

These type of CI may seem natural and obvious but actually requires a quite convoluted argument to motivate. Without going into details, you get this CI by assuming that there exists a transformation h such that the distribution of $h(\hat{\theta}) - h(\theta)$ is pivotal, symmetric and centered around 0. You then construct a Basic CI for $h(\theta)$ rather than θ itself, get the α quantiles and transform those back to the original scale by applying h^{-1}. Observe that although we do not need to know h explicitly, the existence of such a transformation is a must, which is not always the case.

[2]There is no general agreement on the rounding scheme to use. Some sources use ceiling (integer closest to x from above) and others use flooring (integer closest to x from below).

5 Using the nonparametric bootstrap to evaluate probabilities

We can use the bootstrap as a tool to approximate quantities via random simulations. Say that we wish to use it to approximate the probability of an event. Let's see a practical example of how we could do this.

Say that we have two theoretical populations $\mathcal{P}_1$ and $\mathcal{P}_2$ for which we want to learn some features. As an example $\mathcal{P}_1$ could represent subjects treated with drug A while $\mathcal{P}_2$ is subjects treated with drug B, and say that we wish to learn whether drug B is more effective than A for some pathology, where the mean efficacy rate for subject using A is θ_A and is θ_B for subjects using B. Or we have that $\mathcal{P}_1$ is the (theoretical) population of units of a product produced by a global company during dayshifts, while $\mathcal{P}_2$ is the theoretical population of units produced during nightshift in the same company, and the goal is to compare the mean number of daily faulty products produced in the two shifts (populations are "theoretical" because they include all potential units ever produced in the past, present and future). Say that here the mean number of faulty products is θ_1 for $\mathcal{P}_1$ and is θ_2 for $\mathcal{P}_2$. Both θ_1 and θ_2 are unknown to us.

In both cases above we want to understand the direction of the relationship $\Delta = \theta_A - \theta_B$ (positive, negative or zero?) and of $\Delta = \theta_1 - \theta_2$ (positive, negative or zero?), as of course having $\Delta > 0$ would mean that, in the first case, the efficacy rate for drug A is larger than drug B, etc.

Now, we need statistics and probability, as Δ itself cannot be observed *in the populations*, as the θ's are completely unknown to us. But we have some data. Now let's consider the first example and assume that we have n_A subjects treated with A and n_B subject treated with B. We also know how many of the n_A subjects have improved their condition due to assuming A, and we know how many of the n_B subjects have improved their condition due to assuming B. This way, we take $\bar{X}_A$ and $\bar{X}_B$ as *estimators* for (the unknown) θ_A and θ_B, respectively, where $\bar{X}_A = \#(\text{improved using A})/n_A$ and $\bar{X}_B = \#(\text{improved using B})/n_B$.

While we could just compare $\bar{X}_A$ and $\bar{X}_B$ and declare that, if $\bar{X}_A > \bar{X}_B$ (that is if $\hat{\Delta} = \bar{X}_A - \bar{X}_B > 0$), then "drug A is more effective than B", this would be actually hazardous as only based on the specific dataset we have happened to observe, it is not statistical *inference*. We would like to know more, as for example if we were given different datasets maybe the conclusions would be different. **What we want is to find an approximation to the distribution of $\hat{\Delta}$ (reflecting its variability as we pick different datasets and have different estimated efficacy rates for A and B) so that we can compute an approximation of the probability $Pr(\hat{\Delta} > 0)$.**
A non-parametric bootstrap procedure is as follows, which we execute R times, where here R is the number of bootstrap samples (we used B in previous sections):

- Set $r = 1$;
- From the set of n_A subject treated with A, sample subjects (with replacement) n_A times, to obtain a bootstrap sample, and in the latter count how many subjects have obtained an improvement due to using drug A. The ratio between such number and n_A is denoted $\bar{X}^*_A = \#(\text{improved using A})/n_A$.
- do the same as above, but for B: sample with replacement n_B times, ultimately to obtain $\bar{X}^*_B = \#(\text{improved using B})/n_B$.
- Compute and store $\hat{\Delta}^*_r = \bar{X}^*_A - \bar{X}^*_B$.
- increase r to $r + 1$, if $r = R$ stop.

In the end you have a collection of R values $(\hat{\Delta}^*_1, ..., \hat{\Delta}^*_R)$. You can now do several things with these value, such as producing an histogram for it, and you can also check how many bootstrap

values are larger than a certain number, see below. For example, you can use $(\hat{\Delta}_1^*, ..., \hat{\Delta}_R^*)$ to approximate the probability

$$Pr(\bar{X}_A - \bar{X}_B > 0) \approx \frac{\sum_{r=1}^{R} \mathbb{I}(\hat{\Delta}_r^* > 0)}{R}$$

where $\mathbb{I}$ is the indicator function such that $\mathbb{I}(x) = 1$ if x is true and 0 otherwise. That is $\sum_{r=1}^{R} \mathbb{I}(\hat{\Delta}_r^* > 0)$ is the number of bootstrap samples for which $\bar{X}_A^* > \bar{X}_B^*$. Therefore, the previous ratio gives an approximation to the probability[3] we were interested in, namely "what is the probability that the mean efficacy rate for drug A is larger than the mean efficacy rate for drug B"?

6 Limitations of the bootstrap

Yes, those do exist. Bootstrap tends to give results easily (too much so, in fact), but it is possible that those results are completely wrong. More than that, they can be completely wrong without being obvious about it. The following are some such situations.

6.1 Infinite variance

In the "general idea" of bootstrapping we plugged in an estimate $\hat{F}$ of F, and then sampled from it. This works only if $\hat{F}$ actually is a good estimate of F, that is it captures the essential features of F despite being based on a finite sample. This may not be the case if F is very heavy tailed, i.e. has infinite variance. The intuition is that in this case extremely large or small values can occur, and when they do they have a great effect on the bootstrap estimate of the distribution of $\hat{\theta}$, making it unstable. As a consequence, the measures of accuracy of $\hat{\theta}$, such as CI, will be unreliable.

The classic example of this is the mean estimator $\hat{\theta} = \bar{X}$ with the data generated from a Cauchy distribution. For it, both the first and the second moments (e.g. mean and variance) are infinite, leading to nonsensical confidence intervals even for large sample sizes. A less obvious example is a non-central Student t distribution with 2 degrees of freedom. This distribution has pdf

$$f_X(x) = \frac{1}{2\,(1 + (x - \mu)^2)^{3/2}} \quad \text{for} \quad x \in \mathbb{R}.$$

where μ is the location parameter, defined as $\mathbf{E}[X]$. So, the first moment is finite and its estimator $\bar{X}$ is consistent. The second moment, however, is infinite, and the right tail of the distribution grows heavier with increasing μ. This leads to the 95% CI coverage probabilities that are quite far from the supposed 95% even when the sample size n is as large as 500 (that is the percentage of confidence intervals that are supposed to contain the true parameter value would be far from the expected 95%).

6.2 Parameter on the boundary

The classical example here is the $U(0, \theta)$ distribution. The maximum likelihood estimate $\hat{\theta}$ is simply $\max(x_1, ..., x_n)$, which will always be biased ($\hat{\theta} < \theta$). In this case the non-parametric bootstrap leads to a very discrete distribution, with more than half of the bootstrap estimates

[3]For those who wonder: this is not equivalent to a "p-value". We haven't discussed p-values nor hypothesis testing in this course, but if we did, such p-value could be obtained via bootstrap using a slightly different procedure.

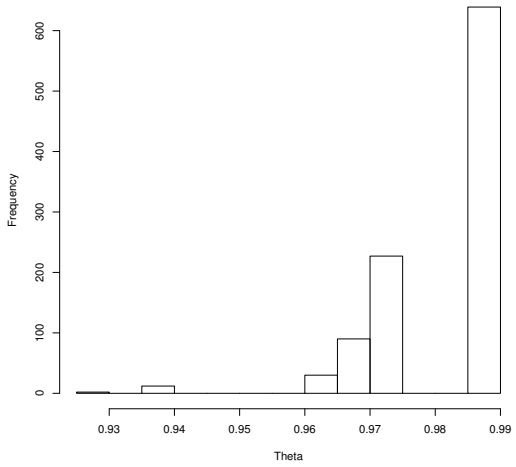


Figure 2: Histogram of 1000 non-parametric bootstrap samples of $\hat{\theta}^*$, the data from the uniform $U(0, \theta)$ distribution. $\theta = 1$.

$\hat{\theta}^*$ equal to the original $\hat{\theta}$ (Figure 2). Clearly, if the quantiles used in CIs are taken from this distribution the results will be far from accurate. However, the parametric bootstrap will give a much smoother distribution and more reliable results.

6.3 Lack of pivotality (* can be skipped, only for those interested)

In all the CI descriptions above the word "pivotality" shows up. So we can guess that it is a bad thing not to have. To circumvent this, something called "studentized bootstrap" can be used.

The idea behind the method is simple and can be seen as an extrapolation of the basic bootstrap CI. There, we looked at $W = \hat{\theta} - \theta$. Now, we instead consider the standardized version $W = (\hat{\theta} - \theta)/\sigma$, with σ denoting the standard deviation of $\hat{\theta}$. The confidence interval will then be calculated through

$$\mathbf{P}(w_{\alpha/2} \leq W \leq w_{1-\alpha/2}) = \mathbf{P}(\hat{\theta} - w_{1-\alpha/2}\sigma \leq \theta \leq \hat{\theta} - w_{\alpha/2}\sigma) = 1 - \alpha$$

As with the basic CI, the distribution of W is not known and has to be approximated, this time with the distribution of $W^* = (\hat{\theta}^* - \hat{\theta})/\hat{se}^*$. Here, $\hat{se}^*$ denotes the standard deviation corresponding to *each bootstrap sample*.

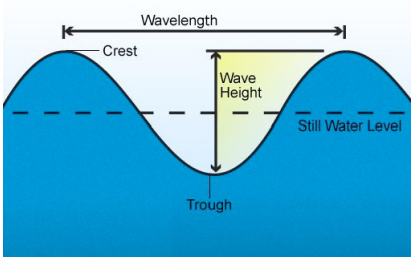
This CI is considered to be more reliable than the ones described earlier. However, there is a catch, and this catch is the estimate of the standard deviation of $\hat{\theta}^*$, $\hat{se}^*$. This estimate is not easily obtained. You can get it either parametrically (but then you need a model which you probably don't have) or through re-sampling. This means that you have to do an extra bootstrap for each of the original bootstrap samples. That is, you will have a loop within a loop, and it can become very heavy computationally.

6.4 Further reading

For a wider and deeper treatment of the bootstrap see chapters 10 and 11 in *Computer Age Statistical Inference* by Bradley Efron and Trevor Hastie. The PDF has been made freely available from the publisher at `https://web.stanford.edu/~hastie/CASI/`.

Exercise 1 (to be solved in MATLAB), xxx points

"The significant-wave-height is the average height of the highest one-third[4] *of all measured waves (measured from trough to crest), which is equivalent to the estimate that would be made by a visual observer at sea."* Imagine to observe for some time length all the waves from some location, and measure their heights, then sort those in decreasing order of height, then look at the initial one-third of the sorted heights (hence the one-third of largest heights) and compute the sample mean of that one-third. The result is the significant-wave-height of the observed waves. In the exercise, the data-file you are given does not contain the raw data of the waves, but is a collection of significant-wave-heights computed in different periods.



The data file `atlantic.txt` contains the significant-wave-height (I believe measurements are in feet) recorded 14 times a month during several winter months in the north Atlantic. To make inference for this type of data it is useful to look at a family of distributions that is known to be suitable for "extreme values".

It was found that a good fit to the empirical distribution of the data is given by a Gumbel distribution (also known as "type 1 extreme value distribution"). This is a distribution for rare events parametrised by two parameters, the "location" $\mu \in \mathbb{R}$ and the "scale" $\beta > 0$. The location μ also represents the mean and the mode of the distribution. For example, it is useful in predicting the chance that an extreme earthquake, flood or other natural disaster will occur. You can estimate the parameters from the data by maximum likelihood using the Matlab function `evfit`, however care has to be devoted to this, as I know explain.

In fact, the Gumble distribution can be written in two ways, as we can use it to describe both "large values of random variables" and "small ones". We are interested in modelling significant-wave-heights, and hence "tall" waves, so we use the version for "large quantities" that is given by the probability density function (pdf) in equation (1). However you can read a discussion (optional) for the other parametrization (small values) in `https://www.itl.nist.gov/div898/handbook/eda/section3/eda366g.htm`

$$f(x;\mu,\beta) = \frac{1}{\beta}\exp\left(-\frac{x-\mu}{\beta}\right)\exp\left(-\exp(-\frac{x-\mu}{\beta})\right), \qquad x \in \mathbb{R}. \tag{1}$$

The problem is that Matlab uses the other parametrization (the one to model small values) to find maximum likelihood estimates, to simulate random numbers etc, so here I give you tips to handle this, which basically amounts to changing the sign of the provided data when using `evfit`, and also the sign of the estimated μ.

[4] "Only the highest one-third is used, since this corresponds best with visual observations of experienced mariners, whose vision apparently focuses on the higher waves.", `https://en.wikipedia.org/wiki/Significant_wave_height`

To find the maximum likelihood estimates of μ and β from given data, when we wish to use the version in (1):

```
out = evfit(-data);  % notice I am flipping the sign of the data
mu_hat = -out(1); % changing the sign of the \mu estimate to be compatible with (1)
beta_hat = out(2);
```

The obtained estimates are now compatible with (1).

And here is another tip for generating pseudo-random numbers from the Gumbel in eq. (1) when using Matlab's `evrnd`.

```
% suppose mu_hat and beta_hat have been obtained as above:
draws = -evrnd(-mu_hat,beta_hat)
% the above generates a single pseudorandom number.
% type "help evrnd" to see how to sample vectors of iid Gumble values
```

(a) [xxx points] Find the maximum likelihood estimate $(\hat{\mu}, \hat{\beta})$ of (μ, β), using the given data, and using such estimates show that the plot of the corresponding pdf (1) adapts well to the empirical distribution of the data (for the latter you can use the `histogram` function, using an appropriate `'Normalization'` option for suitable comparison with the pdf, type `help histogram` for details).

(b) [xxx points] You are asked to use parametric bootstrap with $B = 2,000$ to estimate (i) the distribution of the estimated $\hat{\mu}$, that is the distribution of the estimated mean of the significant-wave-height (plot the histogram), and (ii) estimate the distribution of the maximum value of the significant-wave-height (plot the histogram). To clarify, with "distribution of the maximum value of the significant-wave-height" I mean the distribution of $x_{max} = \max\{x_1, ..., x_n\}$. [**Note:** for both (i) and (ii) please place `rng(123)` before the for-loops, in both cases, to ease grading.]

(c) [xxx points] Based on question b(ii), suppose we wish to advise the major of a nearby coastal city on building a wall that protects the dock from even the highest wave, how high should this wall be?

Exercise 2 (to be solved in MATLAB), xxx points

Assume we have have developed an app where our customers are exposed to adverts every 3 minutes. We want to study if they spend more in-game-time (IGT, in minutes) when we show them adverts every 4 minutes. We label the customers who see ads every 3 minutes as "Control", and those that see ads every 4 minutes as "Variant". Data pertaining 2,600 subjects are in the file `gametime.txt`, consisting of records of three variables: `UserID`, `Group` (Control/Variant) and `IGT`. To help you a little, below you are given some examples of commands to load and access this specific dataset. More general purpose Matlab commands are at the end of the document.

```
% load data
data=readtable('gametime.txt');

% this is a "table format"
%you may access data via column names
% for example
data.('UserID')  % all UserID values
% or
data.UserID
data.UserID(1:3)  % the first three UserID values

% Obtain all ''Control'' subjects by comparing strings (strcmp)
data(strcmp(data.Group,'Control'),:)
# and similarly for the Variant group
```

(i) Using a nonparametric bootstrap procedure with $B = 2,000$, compute the probability that the mean IGT for the Variant group is larger than the mean IGT for the Control group. *[**Note**: before running the bootstrap set the seed `rng(123)` in your script, so we get the same results.]*.

(ii) Show the histogram of the distribution you previously obtained from the bootstrap.

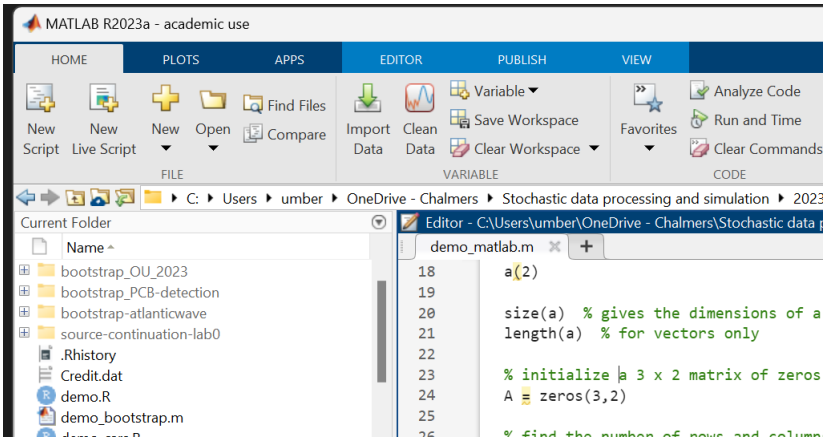
(iii) Using a procedure similar to what you did in (i), compute the probability that the mean IGT for the Variant group is at least one minute larger than mean IGT for the Control group. *[**Note**: before running the bootstrap set the seed `rng(123)` in your script, so we get the same results.]*.

7 Quick Matlab primer

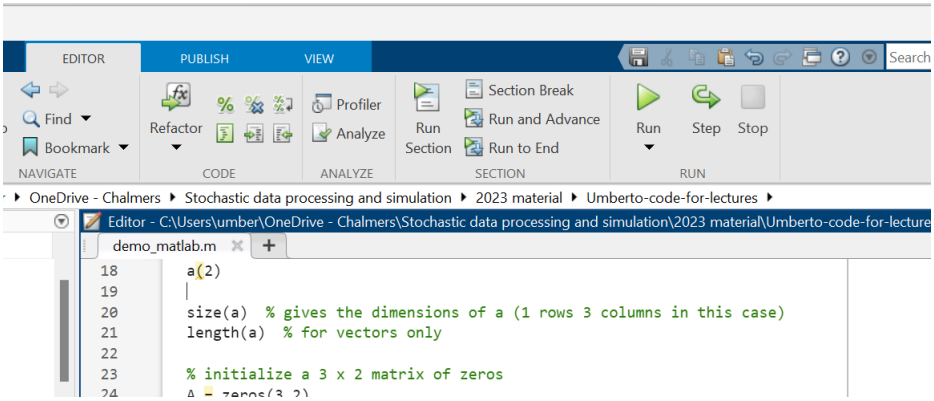
Just a few words on Matlab: if you got used to R and have never used Matlab, there is not an awful lot to learn, you can check cheat-sheets:

- MATLAB basic functions https://www.mathworks.com/content/dam/mathworks/fact-sheet/matlab-basic-functions-reference.pdf
- a MATLAB/R cheat sheet is at http://mathesaurus.sourceforge.net/octave-r.html
- a MATLAB/Python/R cheat sheet is http://mathesaurus.sourceforge.net/matlab-python-xref.pdf

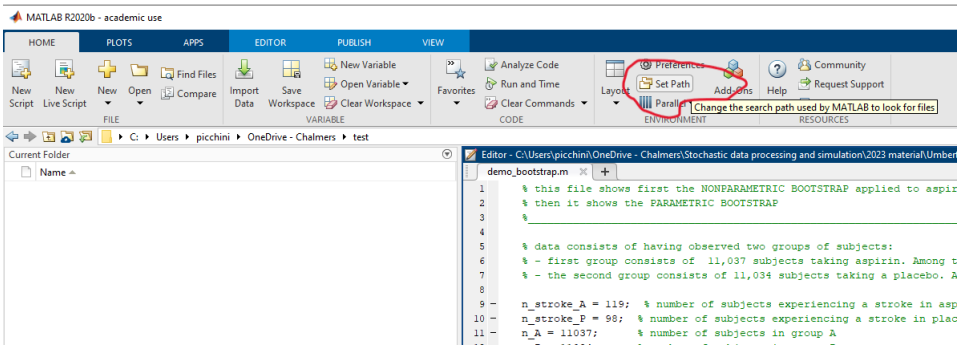
Create a script: Click on the HOME tab and click on New Script to the left:

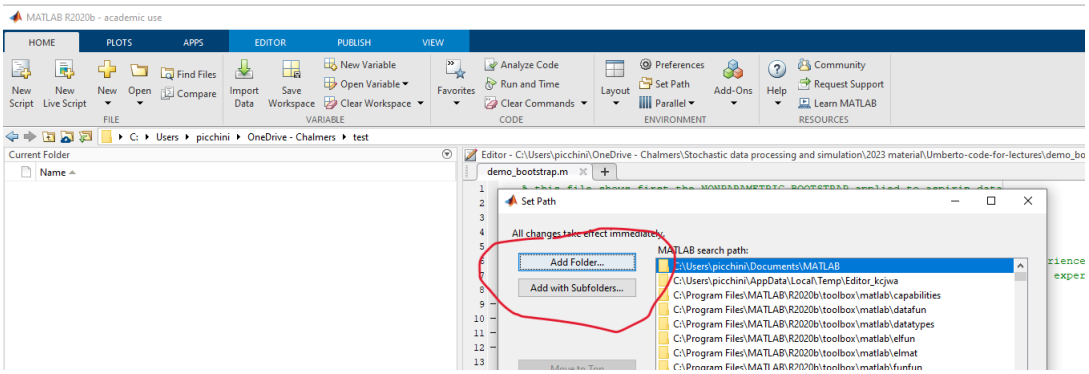


Run an entire script: First click o the script itself, so a new tab called EDITOR will appear (otherwise you won't see it if no script is open). Then click on the large green arrow "Run".



Another relevant thing is to know how to change your work directory:
click on the HOME tab on the top then select "Set Path", then click on either "Add folder" or even "Add with subfolders".





Basically you can use the previous procedure to add as many folders as you need, if your codes are scattered across different directories, so Matlab is aware of where to look to find them (I recommend to keep things tidy though).

Examples of things to know if you are new to Matlab: also check the file `demo_matlab.m`

- vectors are defined with square parentheses, eg

```
>> a=[1,2,3]
a =
     1     2     3
```

And you access vector's elements using round parenthesis, eg `a(3)`

```
>> a(3)
ans =
     3
```

- use semicolons ";" a lot or your scripts will print lots of things on screen , e.g.

```
>> a=[1,2,3]
a =
```

```
     1     2     3
% whereas

>> a=[1,2,3];  % this will not print anything on-screen
```

- Comments can be written via %
- Adding a plot to an already existing figure. Use `hold on` and `hold off`.

```
plot(randn(1,10),randn(1,10),'o')
hold on  % this will allow superimposing a plot on top of the existing plot
plot([1:0.01:10],[[-10:0.05:35]],'m--')  % added a dashed magenta line
hold off  % stop superimposing
```

Help/Documentation: for info on MATLAB's commands you may use both `help` and `doc` at the prompt, followed by an appropriate keyword, e.g. `help histogram` or `doc histogram`.

Setting the pseudorandom numbers seed: this can be done via `rng(123)`, where 123 is just an example (similarly to R's `set.seed()`).

Be careful when specifying dimensions for pseudorandom numbers: unlike in R, the function to produce a vector of uniform random numbers expects **two** dimensions to be provided. That is `rand(10000,1)` generates a 10000×1 column vector, while `rand(1,10000)` generates a row vector. However, if you were to write `rand(10000)`, this would create a $10,000 \times 10,000$ matrix.

Producing Gaussian pseudorandom numbers: you can use `randn` to produce a single draw from $N(0,1)$ (that is from the *standard Gaussian* aka *standard normal* distribution). To sample from a generic $N(\mu, \sigma^2)$ you can type $\mu + \sigma$`*randn` or use `normrnd(`μ`,`σ`)`. Notice σ^2 in $N(\mu, \sigma^2)$ is the variance, but software actually requires you to type in the standard deviation σ. How about dimensions? You have to be careful: if you wish to simulate a column vector of 10 elements from $N(0,1)$ then you can just type `randn(10,1)` *however* if you want to simulate a column vector of 10 elements from $N(\mu = 3, \sigma^2 = 4)$ then you have to write `normrnd(3,2,[10,1])` (notice the dimensions are specified between square brackets and the standard deviation is specified as 2, not the variance).

More examples in `demo_matlab.m`.