***Currently available SSD software tools***

At least nine software tools have been developed to fit a subset of 10 different distributions to data using one or more methods (Table 1). We consider Maximum-Likelihood to be the most suitable method from a regulatory perspective, because it is less biased than Moments Matching, does not require the specification of prior distributions (unlike Bayesian methods) can be used for model averaging (unlike Least Squares).

All the tools, which are free to use, estimate the HC5, and at least one other hazard concentration, with confidence intervals. The most common distributions are the log-logistic and log-normal which are each implemented in six of the nine tools. All the distributions are two parameter distributions (the log-triangular is symmetric) with the exception of the Burr Type III and possibly the log-t distributions.

INSERT TABLE FROM EXCEL FILE

Table 1. Software tools to fit distributions to data using Least Squares (LS), Moments Matching (MM), Maximum Likelihood (ML) and Bayesian (BY) analysis.

Only the SSD Toolbox (Etterson 2020) which has recently been released by USEPA and (shiny)ssdtools which was developed for the British Columbia Ministry of Environment and Climate Change Strategy (Thorley and Schwarz 2018; Dalgarno 2018) implement model averaging. It is important to be aware that (shiny)ssdtools consists of ssdtools - a stand-alone R package (Thorley and Schwarz 2018) - and shinyssdtools (Dalgarno 2018) a second R package which provides a bilingual (English and French) Graphical User Interfaces (GUI) to ssdtools. The advantage of this separation are discussed below. As only SSD Toolbox and (shiny)ssdtools fit six of the 10 distributions using Maximum Likelihood, run on all three major platforms, and have graphical user interfaces (GUI) we consider them to be the most useful of the nine software tools. Consequently, they are the focus for the remainder of this section.

SSD Toolbox is written in the commercial MATLAB® language and provided as a pre-complied binary that can be run by locally installing the free MATLAB Runtime libraries. ssdtools and shinyssdtools are both written in the open source R language (R Core Team 2020). The source code for both has been released under the open source Apache-2.0 Licence (<https://github.com/bcgov/ssdtools> and https://github.com/bcgov/shinyssdtools/) which allows users to modify and/or distribute the code under the same licence. We consider open source software to be preferable because it allows code validation and facilitates collaboration and replication .

SSD Toolbox allows distributions to be fitted using Bayesian methods and can statistically account for multiple datapoints for each species through the use of hierarchical models. Neither of these features are currently implemented in (shiny)ssdtools. However, by separating the scripting and GUI components into ssdtools and shinyssdtools, respectively, developers can readily extend (shiny)ssdtools functionality or incorporate it into their own software. shinyssdtools also provides an R script allowing the user to replicate the analysis they performed through the GUI. Finally, a web-based version of shinyssdtools which does not require the user to install R and runs on any browser is available at https://bcgov-env.shinyapps.io/ssdtools/.



While we are not advocating adoption of a single standard approach or tool, we think there is a need for closer jurisdictional collaboration, greater harmonisation of methods, and development of at least some benchmark data sets and reference results. The last of these is particularly pressing given the frequency with which we have observed noticeably different HCx values for the same data set from the different tools in Table 1. This is to be expected if different estimation strategies are employed (for example maximum likelihood versus method of moments or single SSD versus a model-averaged SSD) but all things being equal, all tools should give the same point estimates to within some nominally small tolerance (e.g.1-2%). Certainly, differences of a factor of 2 or more are indicative of flawed coding and/or numerical instabilities and convergence issues. This is not a new idea and indeed ‘reference data sets’ were commonly used in the early days of statistical computing to allow both software developers and end-users to assess the adequacy of numerical routines underpinning routine analyses such as ANOVA, regression, and correlation. Even today, the National Institute of Standards and Technology still maintains a number of statistical reference data sets at <https://itl.nist.gov/div898/strd/index.html>, including the famous Longley data set (Longley 1967). Several of the current authors are preparing a paper that looks at the performance of the various software tools with various case studies.