Beyond visualizing catch-at-age models: lessons learned from the r4ss package about software to support stock assessments

Ian G. Taylor[a,*], Kathryn L. Doering[b], Kelli F. Johnson[a], Chantel R. Wetzel[a], Ian J. Stewart[c]

[a]Fishery Resource Analysis and Monitoring Division, Northwest Fisheries Science Center, National Marine Fisheries Service, National Oceanic and Atmospheric Administration, 2725 Montlake Blvd E, Seattle, WA 98112, USA

[b]Caelum Research Corporation under contract to Northwest Fisheries Science Center, National Marine Fisheries Service, National Oceanic and Atmospheric Administration, 2725 Montlake Blvd E, Seattle, WA 98112, USA

[c]International Pacific Halibut Commission, 2320 West Commodore Way, Suite 300, Seattle, WA 98199, United States

[*]Corresponding author. Email address: ian.taylor@noaa.gov

Keywords: r4ss, Stock Synthesis, Model diagnostics, Conditional age-at-length data, R packages

Abstract

Stock assessment analysts are exploring an increasingly diverse and complex range of models while also facing higher expectations for consistency, documentation, and transparency in reports and management advice, all within a tight timeline. Meeting these goals requires increased efficiency at all steps in the assessment process from data processing, through model development and selection, to report writing and review. Here, we describe one widely used tool that has proven successful in increasing the efficiency of the assessment process: the *r4ss* package, which supports the use of the Stock Synthesis modeling framework. What began 15 years ago as a tool to provide simple model diagnostics, including plots showing data and model results, has grown into a large collection of R functions to support many aspects of the assessment process. We provide an overview of the *r4ss* features and illustrate its utility with examples from recent applications. Finally, we discuss lessons learned from the ongoing development of *r4ss* that can be applied to similar efforts associated with the next generation of stock assessment packages.

1. Introduction

Assessment of fish stocks (hereafter referred to as “stocks”) is a necessary task, largely because of mandates by federal and regional governing bodies to provide information about stock status and apply harvest control rules to inform catch limits under harvest policies. While incorporating disparate data sources into a single population model (integrated analysis) to determine stock status is routine, understanding the fit to each data set and its associated influence on the model results can be challenging (Maunder and Punt, 2013; Maunder and Piner, 2015). A standardized set of visualization tools is key to providing understanding and transparency throughout this process for stock assessment analysts, reviewers, stakeholders, and managers. For example, standardized tools allow analysts to quickly understand model results and explore new model configurations during the model development and peer review processes; reviewers scrutinize the analyses and investigate other alternatives with the aid of visualization tools, ultimately deciding if the assessment results are appropriate for use by management; and lastly, stakeholders and managers need to understand the model results, and hence, need intuitive visualization tools to inform the range of management options and decide on which management measures to take.

Visualization tools can aid analysts throughout the assessment process. For example, Richards et al. (1997) found while developing a stock assessment for Pacific ocean perch (*Sebastes alutus*) that visualization tools allowed them to better understand their data sets and pinpoint data features that needed to be accommodated, develop a statistical catch-at-age model well suited to the data sets, and evaluate model output more thoroughly. Stock assessments often require hundreds of model runs. Tools for quickly visualizing model results allow analysts to more efficiently select among them. As an illustration of the power of automated workflows and visualization tools, calculating residuals by hand would take hours, while visualizing patterns in residuals already plotted can take just minutes. Visualization tools can also relieve the feeling of being time-poor when conducting stock assessments (Bentley, 2015). Aside from efficiency, a thorough and standardized set of tools for visualizing model output can help catch errors such as misspecified models and aid in the report writing process, as most stock assessment reports require numerous figures and tables.

The peer review process for stock assessments (e.g., Brown et al., 2020), to determine if assessment results can be used by management bodies for decision making, benefits from visualization tools. For example, Regular et al. (2020) found that interactive data and model dashboards improved their ability to communicate with stakeholders during the stock assessment review process for a northern cod stock in the northwest Atlantic. Producing standardized figures across assessments increases ease of understanding for readers and simplifies comparisons across assessments. Often, peer reviewers are tasked with evaluating modeling decisions and model results, ultimately deciding if the assessment results are appropriate for use by management. Requests for visualizations made during assessment review processes are often expected in subsequent reviews, especially if the same reviewers may be engaged in the future, and should be added to assessment analyst toolboxes, such that they can be better prepared for future reviews. Thus, this toolbox grows with each review and helps facilitate efficient reviews, because analysts are able to quickly produce desired output before it is asked for.

The Terms of Reference (ToR) for stock assessment reviews have also coevolved with visualization tools, increasing the value of standardization. For instance, 10 years ago the ToR for groundfish stock assessments conducted for the Pacific Fishery Management Council (PFMC 2009) had an eight-point bulleted list of general stock assessment team deliverables, while the ToR used in 2019 (PFMC 2019) had a check list of 74 elements within 18 sections with more specificity. These ToR changes have been driven in part by feedback from reviewers seeing the benefit of new visualizations and diagnostics for individual assessments as described above. The ToR changes, in turn, lead to wider adoption of the new approaches for analysts working to meet them, a shift which is easier when the analysts can use shared tools to meet the new standards.

Effectively translating complicated assessment models and results into an easily digestible form for fishery managers and stakeholders can be challenging, especially when presenting information across a large range of stocks (Dichmont et al. 2016). Presenting assessment results in a consistent manner across stocks can lessen the communication challenge, allowing for improved discussions between analysts, stakeholders, and managers. The development and application of an assessment toolbox for use by analysts facilitates this process without creating additional workload.

Communication methods for stock assessment results are not a frequent topic in fisheries science journals, but it is an area where new ideas are rapidly developing and which deserves greater prominence in the literature. The widespread adoption of the generalized integrated analysis platform Stock Synthesis (SS, Methot and Wetzel, 2013) provided an opportunity to develop a standardized set of visualization and automation tools, given a larger pool of potential applications, users, and contributors (Punt and Maunder, 2013). Here, we discuss how *r4ss*, an R package containing tools for working with SS models, has improved the stock assessment development and review processes for individual analysts, reviewers, and managers over its 15 years of active development. We also highlight lessons learned from developing and using *r4ss* that could be applied when developing new visualization tools for a new stock assessment modeling platform.

1.1 History

The *r4ss* package grew organically from a single code script written by a single author in 2005 for use in the R statistical programming language (R Core Team, 2020) to a large open-source R package with many contributors. Before *r4ss* was developed, the typical workflow for SS users was examining the output text files directly or importing them into Excel where figures were generated using Visual Basic scripts or created manually for each model. The figures were time-consuming to create, had limited reproducibility, and did not provide reviewers and managers with a consistent product with which they could become familiar as modifications for an individual model were rarely generalized for the benefit of other models. The original *r4ss* R script became widely used by the stock assessment team at the

National Oceanic and Atmospheric Administration (NOAA) Northwest Fisheries Science Center (NWFSC) and grew in complexity as members of the assessment team provided suggestions for additions. The increase in use also increased the burden associated with maintaining the code, and in 2008 the lead developer role was shifted to a postdoctoral researcher which allowed for more directed development, facilitating the growth and use of the code to function across SS-assessed stocks. Shortly thereafter, the code was put under version control and released as open source to facilitate distribution and development, to increase transparency, and to reduce the burden of maintenance on any individual developer.

Although, in the early years of its development, most of the code was written by just two people, feedback from users was essential to improving the package. In particular, conversations with participants at the annual Inter-American Tropical Tuna Commission (IATTC) Stock Assessment Workshop series (since succeeded by the Center for the Advancement of Population Assessment Methodology workshops) led to significant steps forward in the project. The initial public release of *r4ss* took place during the 2008 IATTC workshop (Maunder, 2008); discussions at the 2009 workshop inspired the conversion of the script into a formal R package available on the Comprehensive R Archive Network (CRAN); and a demonstration of the Javascript viewer for Multifan-CL (SPC, 2010) associated with the 2011 workshop led to the development of an HTML viewer for *r4ss* plots. Formatting the *r4ss* script as an R package brought the benefits of structured documentation for each function; making the *r4ss* package available on CRAN made it easier to find and install (as CRAN is the first source most users will look to for R packages). The number of authors, all of whom have made substantial code contributions, has also grown from 5 in 2009 to 29 in 2020. The methods used to incorporate code into the *r4ss* codebase have also evolved from contributors emailing files to the lead developer, to GitHub pull requests that get automatically checked and manually reviewed before merging. Although the development workflow has grown more sophisticated, the organic evolution of *r4ss* leaves many legacy aspects of the code and package structure, which are typical of research software (Ram et al. 2019), but would be designed differently if starting from scratch today.

1.2 Overview of the package

The *r4ss* package (github.com/r4ss/r4ss) includes functions designed to work with SS input and output files (Supplement 1). The main types of functions in the package are: 1) functions to read and plot information from SS output files to visualize model results; 2) functions to automate tasks associated with SS models that are routinely performed; and 3) functions to read, create or modify SS input files. In the examples, we will focus on functions to visualize model results and automate routine tasks.

2. Examples

2.1 Multimodel management (Pacific halibut)

The Pacific halibut (*Hippoglossus stenolepis*) stock assessment comprises four individual models which are used to create an ensemble for management use by the International Pacific Halibut Commission (IPHC; Stewart and Martell, 2015; Stewart and Hicks, 2018). Each of the models represent a different hypothesis regarding the best approach for modelling the stock dynamics. The four models vary in the length of the modelled period, the level of data aggregation, and data-weighting, among other factors (Stewart and Hicks, 2020). Use of special features (e.g., environmental covariates, time-varying catchability) in a subset of the models is automatically detected by *r4ss* functions and appropriate reporting is included in subsequent output. With four models to diagnose, it is efficient to run the same function calls and retrieve a complete set of output without having to adjust the code for each individual model. When suitable models have been identified and refined, the output can be easily summarized (Figure 1).

This application also highlights the benefit of standardized output for review purposes. For the 2019 stock assessment, the IPHC conducted a two-stage review, utilizing its standing Scientific Review Board, as well as a contracted external reviewer (IPHC 2020). Model input files and directories containing the *r4ss* HTML and individual output files from the *SS_output* function were provided electronically to all reviewers. This approach allowed for standardized reporting (reviewers were all familiar with *r4ss* output) and comprehensive diagnostics for all four models in a more detailed summary than could have been provided only in a standard written document. Reviewers were not required to re-run the *r4ss* code to explore the detailed results, but the HTML-approach re-created a user-friendly summary on demand.

2.2 Adoption of new modeling approaches and diagnostics (Big skate)

The 2019 assessment of big skate (*Beringraja binoculata*) off the U.S. west coast (Taylor et al., 2019) illustrates the efficiency gained by using tools like *r4ss*. Specifically, *r4ss* facilitated the use of conditional age-at-length (CAAL) data, exploration of numerous model configurations, timely model development during the review process, and an efficient development and revision of the assessment report.

The stock assessment of big skate used CAAL data (Figure 2), which has become a standard treatment of age- and length-composition data collected from the same samples used in integrated assessments to reduce biases associated with length-based selection and better inform parameters related to variability in length-at-age (Hoyle and Maunder, 2006; Stewart, 2005; Piner et al., 2016). However, adopting this approach to modeling compositional data was initially hampered by lack of associated model diagnostics and appropriate data weighting methods. In response to this concern new diagnostics were developed, including the calculation of the implied fit to the marginal age composition (called "ghost" age compositions within *r4ss*) and a new diagnostic figure to evaluate the fit of the model expectation to the mean and variability around the mean age at length (Figure 3). Data weighting approaches for CAAL data have likewise been the subject of ongoing research and new diagnostics. Francis (2011) proposed a new approach to tuning input sample sizes for marginal composition data. Two authors independently developed R code for applying the Francis (2011) algorithm and their functions were combined and contributed to *r4ss* in 2014. This facilitated more frequent use of this method and the discovery that an extension was needed for CAAL data (Punt 2017) which was subsequently incorporated into *r4ss.* The assessment of big skate used the Francis-Punt tuning method but also considered the sensitivity of the model results to the Dirichlet-Multinomial (Thorson et al., 2017) and McAllister-Ianelli (McAllister and Ianelli, 1997) methods of data weighting as alternatives. These sensitivity analyses were trivial to implement and easy to compare thanks to the support for all three approaches in SS and *r4ss*. Choosing among the methods unfortunately remains somewhat subjective, but the opportunity to quickly develop all three models and examine the associated fits to the CAAL data and other data types is essential for understanding the impact of data weighting choices on the perception of stock status and the structural uncertainty associated with assessment results.

The assessment of big skate also benefited from previous work to understand sex ratios in models fit to data for Pacific halibut. These explorations led to the inclusion of a new diagnostic figure within *r4ss* showing the sex ratio derived from the observed and expected age- or length-composition data (Figure 4). For big skate, the figure revealed unbalanced sex ratios at sizes where there was little evidence of dimorphic growth, suggesting that sex-specific differences in selectivity should be included in the model. The availability of a diagnostic previously developed for a different stock was vital to the model development during a period where the time available to explore new ways to understand patterns in the data was limited. After applying the diagnostic to big skate, discussions facilitated by the use of the GitHub issue tracker led to further refinement of the diagnostic. This collaborative effort highlights how open source code can increase efficiency, promote collective understanding, and allow for incremental progress.

2.3 Efficient exploration of alternative models (Big skate)

In the process of developing the big skate assessment, over 800 different model configurations were run during 2 months, almost 200 of which were run during the 5-day review meeting. These included seven new candidate "base" models, 108 models that were part of likelihood profiles, 72 sensitivity analyses, and three alternative forecast scenarios. Functions in *r4ss* were used to run the likelihood profiles and create the associated plots as well as automatically repeat sensitivity analyses for new candidate models. This volume of model examination, comparison, and selection has become a typical part of the assessment process in many regions, but thorough examination of this many models is only possible with tools to quickly compare and aggregate results.

In total, approximately 40,000 standard *r4ss* diagnostic figures were created during the 2-month model development period (~200 figures from each of ~200 models). The option to look at a suite of diagnostics for any given model allowed the authors to quickly determine whether the data were entered into the model input files correctly, look for outliers, and, as the models developed, evaluate whether alternative modeling approaches merited further consideration. The full collection of diagnostic figures and tables and the associated HTML viewer files (e.g., r4ss.github.io/r4ss/BigSkate) for models that were discussed in the review were shared with the review panel to allow them to further explore these models beyond the limited information that could be provided in a brief presentation.

2.4 Bayesian applications (Pacific hake)

The use of Bayesian methods for integrated assessment models have well-documented benefits (e.g. use of prior information and better characterization of posterior distribution) and challenges (slow run times and shifting focus from uncertainty in model structure) (Maunder 2003, Stewart et al. 2013; Monnahan et al. 2019). Analysts using Bayesian approaches are best served by having efficient approaches for comparing posterior distributions to maximum posterior density (MPD) estimates and associated asymptotic uncertainty (Fournier et al. 2012; Stewart et al. 2013). Examining the relationship between these posterior estimates and prior distributions is also vital to understanding the influence of the choice of prior. The annual stock assessment for Pacific hake (Grandin et al. 2020) has used Markov Chain Monte Carlo (MCMC) sampling to characterize posterior density distributions for nearly two decades. The *r4ss* package can be used to plot the prior and posterior distribution of all estimated parameters (Figure 5) as well as time series plots comparing quantiles from the posterior samples of derived quantities to the associated MPD estimates and their asymptotic uncertainty intervals. This functionality was initially developed for Pacific hake in 2011 but generalized to work for any SS model; it has allowed the assessment team for Pacific hake to demonstrate the qualitative similarities between the two estimation methods and use more computationally intensive MCMC sampling for a subset of the models explored while showing only the more rapidly available MPD results for numerous sensitivity analyses. However, many of the plots included in the Pacific hake assessment report require custom code (available at github.com/pacific-hake/hake-assessment) as the *r4ss* package does not contain the necessary tools to plot a variety of valuable diagnostics for Bayesian models, such as the fit of posterior distribution of expected values to observed data. Adoption of the no-U-turn sampler for SS and other models using ADMB and TMB (Monnahan and Kristensen, 2018), which decreases model runtime compared to the default Metropolis-Hasting algorithm used for MCMC in ADMB, may increase the use of posterior sampling in integrated models and inspire generalization of the custom Pacific hake code for more widespread use by users of *r4ss* or other generalized packages.

3. Collective experience of the authors

In addition to the examples above, *r4ss* has facilitated the formalization of many assessment authors' "tips and tricks" for efficiently building, diagnosing, and reporting stock assessment models. Sharing of collective experience reduces the learning curve for new assessment authors and also provides structure to remind experienced authors of perennial pitfalls. This section reports a series of problems that we the authors have collectively encountered across a large number of individual stock assessments that standardized and generalized tools like *r4ss* can solve (Table 1).

4. Discussion

Software packages are often described as black boxes (Dichmont et al., 2016) and fitting models to data has previously been described as an art rather than a science because of numerous non-trivial choices in the model development process (e.g., how to specify the model, how to weight the data). Fortunately, stock assessment scientists are formally trained in at least either model development or model fitting, helping to ensure that results fulfill mandates to provide best available science. Where stock assessment scientists typically lack formal training is in data visualization and how to effectively communicate science to stakeholders and fishery managers. Punt and Hilborn (1997) summed up the problem suggesting that "the art of stock assessment involves determining what to present to managers and the best way to summarize information." The importance of creating effective visualizations and effectively communicating science has also been noted in Management Strategy Evaluation processes (Miller et al.

2019). Because it is difficult to predict what decision makers know versus what they need to know (Fischhoff, 2013), leaning towards archiving vast amounts of easily accessible information is a typical practice among stock assessment scientists and their associated organizations. Key to this approach has been the development of open source tools such as *r4ss* that reduce the time needed to manipulate model-input files and run sensitivity analyses, summarize model output in a standardized way to minimize the time needed to interpret output, and openly document assumptions and protocols used along the way.

4.1 Successes

We attribute some of the success of *r4ss* to increases in the use and usability of open source tools for science (e.g., Stewart Lowndes et al., 2017; Boettiger et al., 2015; Huber et al., 2015). Software development tools like version control and automated testing have facilitated the continuous development process and increased the reliability of the code base. As more scientific disciplines model themselves after the geospatial community, a leader when it comes to open source and free software development (Bocher and Neteler, 2012), code sharing has become the norm. We recognize that exposing one's work in this way may be uncomfortable at first, but the benefits to the community far outweigh the costs. For example, the routine use of *r4ss* by different users with models fit to a diverse array of datasets leads to the discovery of bugs (whereas bugs may go unnoticed with personal code) that can be reported (to github.com/r4ss/r4ss/issues) and fixed. Many of the users also suggested fixes and improvements. Thus, the *r4ss* functions become more reliable and more useful than code used for one-off manipulations.

The *r4ss* project has been successful for the stock assessment process as a whole. For individual stock assessment analysts, using *r4ss* allows for a more automated workflow in which it is easier to pinpoint issues during the base model development process, run sensitivity analyses, and generate figures for reports. The package can also be used by analysts who lack the time or training to develop their own personal diagnostics code, thus empowering many to conduct higher-quality stock assessments. The availability of *r4ss* as a platform for distributing new diagnostics and tuning methods has contributed to collaboration among researchers and facilitated timely adoption of the new approaches within the stock assessment community. For reviewers, managers, and stakeholders, the standardized and extensive set of plots created by *r4ss* has made the stock assessment process and its results more transparent and easier to interpret. One unintentional consequence of the availability of automated visualization tools is that reviewers and managers often expect more output, which reduces some of the time-savings that the tools provide for individual analysts. However, movement toward more standardized model diagnostics for integrated fisheries models (Carvalho et al., this issue) mitigates this challenge.

Outside of the stock assessment workflow, *r4ss* can also be used as a dependency for other tools for working with SS models. For example, instead of maintaining similar functionality, the *ss3sim* R package (Anderson et al., 2014) imports *r4ss* as a dependency. This allows *ss3sim* to benefit from improvements and fixes added to *r4ss.* As the use of SS has grown, so has the ecosystem of tools that interact with it for conducting simulation analyses, MSEs, data-limited models and other tasks specific to a particular region or stock (Cope and Wetzel, this issue). We foresee a similar proliferation of tools associated with any widely used next generation assessment platform, where the basic functions of interpreting output files and interacting with input files can be contained within a single package on which others depend.

4.2 Challenges

The *r4ss* project also reveals the challenges of a standardized tool. It was created without foresight into what it would become, leading to a variety of problems. For example, a mismatch between the names used in *r4ss* and SS as well as non-standardized naming and output in SS are common issues that are difficult to remedy without a significant impact on the large set of additional R packages and individual scripts that depend on *r4ss*. Keeping up with the development of SS while maintaining backward compatibility with older versions of SS that are still in use is a challenge for the *r4ss* developers. While *r4ss* saves time for users, it creates more work for developers and maintainers.

Another challenge is the increasing scope of *r4ss* over time, as more functions and options have been added to the package. Generally, the scope of *r4ss* is “tools in R to work with Stock Synthesis”, but initially the scope of *r4ss* was “visualization tools in R to work with Stock Synthesis”. This creates challenges in balancing flexibility for advanced users with understandability for newer users, while maintaining interoperability with various dependent projects like *ss3sim* and automated assessment document writing approaches. “Scope creep” is a common problem in software projects, including R packages. For example, the R package *devtools,* which is widely used by R package developers (Wickham, 2015) was split into multiple smaller R packages as a solution to *devtools* becoming large and thus harder to develop and for other packages to depend on (Hester, 2018). A similar approach could be applied to *r4ss* to manage this balance.

In addition to the increase in scope associated with new functions added to r4ss, the range of uses for the figures created by the original r4ss code has grown over time. The original intent was to produce quick diagnostics for use by assessment analysts, but those figures (many of which are effectively unchanged in appearance since 2005), are now frequently used in assessment reports and presentations, contexts which would benefit from additional adjustments and refinements.

Finally, creating a collaborative community is challenging. While *r4ss* has successfully facilitated collaborations among stock assessment scientists working around the world, it is difficult to encourage contributions to a collaborative tool over individual development. This is especially challenging because most stock assessment scientists (like many scientists working in other disciplines) have little to no formal software development training and thus the hurdle of learning collaborative software development tools like version control is high. In cases where formally trained software developers (as opposed to scientists) are the ones writing code, these developers are often working alone, unlike in typical software engineering environments (Killcoyne and Boyle, 2009), so collaboration is not as common. The *r4ss* developers have encouraged collaboration by allowing contributions in multiple ways, from formal pull requests via a version control system to emailing code to *r4ss* maintainers. We also recognize that contributing code is not the only way to contribute to a tool; for example, bug reports and discussions taking place through the r4ss issue tracker also allow people who may never touch the *r4ss* codebase to contribute. Like the Ocean Health Index project (Stewart Lowndes et al., 2017) and many other open source projects, the *r4ss* community attempts to meet collaborators where they are in terms of understanding and using software development tools, while encouraging and empowering them to learn new approaches and practices.

4.3 Recommendations

The 15-year history of *r4ss* development provides insight for the future development of tools to work with stock assessment models. First, automating the creation of model diagnostics and plots is a fundamental aspect of developing a generalized model package, as it is too time consuming for analysts and reviewers to evaluate the validity of models using ad hoc approaches. The challenge of developing *r4ss* after SS highlights the difficulties in retrofitting tools to work with existing models. Retrofitting *r4ss* to SS has led to model inputs and outputs that change over time and are not standardized with other modeling platforms, thus creating an increase in development time for *r4ss*. Thus, we believe that development of future tools for stock assessment modeling workflows should be developed in tandem with any new stock assessment modeling framework and some standardization is needed between the stock assessment model framework (or frameworks) and tools for efficient workflow and model diagnostics. However, the stock assessment modeling framework and associated tools should be kept somewhat independent, as the software requirements and development processes for assessment modeling frameworks, which depend on computational efficiency and reliability, are different from those that best suit associated tools like *r4ss*. While long-term planning is helpful, it will not be possible to perfectly plan the tools because future needs are unpredictable. Adopting a flexible approach will be necessary in a future tool.

We recommend using R as the basis for similar tools associated with the next generation of integrated models. While there are other languages that could be faster or better-suited to some of the tasks, R is widely used in fisheries science (Schnute et al., 2007, Anderson et al. 2020) and thus there is a larger pool of potential users and developers. We argue that having a large group of users and developers is more important than computational efficiency for this aspect of the stock assessment process.

We also think that the open source development model and creating an environment in which users' contributions are encouraged is key to the success of a future diagnostic tool. Once the core elements are in place for a tool, users can contribute functions like new plots without having to understand much about the package. Open source leads to more bugs being caught and fixed quickly while also facilitating the growth of the tool to fit the ever changing needs of stock assessment science. While open source software may be "free", it still has maintenance costs, typically in the form of maintainers' time. In the case of *r4ss*, there is a primary maintainer who typically spends about 8 hours a week working on the package. Maintenance is necessary for the long-term sustainability of software (Ram et al. 2019). Who (organizations or individuals) will maintain the software and how much time they can devote to maintenance should be carefully considered during the planning stages of a new stock assessment modeling framework and its associated diagnostic tool.

Standard software development tools and practices are key for creating scientific software that works well (Wilson et al., 2014; Wilson et al., 2017) and will be important to use in tools designed to work with the next generation of integrated models. At a bare minimum, the software should be under version control, hosted in a public repository and provide standard workflows for collaborative coding. Other tools that are essential to maintaining code that works are a testing framework for running unit and integration tests (and writing tests as code is added) and a continuous integration tool to automatically build the software and run the testing framework often.

One limitation of *r4ss* is it is designed to only work with a single stock assessment platform, SS. Non-standardized inputs and outputs among assessment model frameworks create challenges for comparing modeling approaches. A generalized package not specific to any assessment model framework would eliminate this problem while also allowing analysts to consider other potential model configurations not available in a single framework and more easily compare models developed in different frameworks. However, to our knowledge, no package which has sought to provide this type of generality has been widely used for more than one assessment platform. There are multiple challenges with creating a generalized package for model diagnostics. First, the differences in input and output formats used in different stock assessment platforms (such as those listed in Punt et al., this issue) would add overhead associated with translating into a common framework. Second, development of a generalized tool would require expertise in multiple assessment platforms (e.g., an attempt to create a function within *r4ss* to convert SS output into the format required by another diagnostic package was unsuccessful because no one was adequately familiar with both platforms). Rather than focus on development of a more generalized version of a tool like *r4ss*, we believe that the effort would be better invested in the design and use of more standard formats in the next generation of stock assessment platforms and their associated tools for input and output process and model diagnostics. However, tools to translate input files between assessment packages have been successfully developed largely as one-off projects when the comparison of results across platforms is desired. For example, tools to compile results across assessment platforms were developed while conducting a project to formally compare some stock assessment platforms used in the U.S. (Li, 2020). These tools have the potential to become more generalized for future projects. Unfortunately, it is normally faster to get code to work only for what is directly needed rather than creating generalized code. As ensemble modelling in stock assessment (Stewart and Martell, 2015) becomes more popular, efforts to translate inputs and output between assessment platforms may become increasingly worthwhile.

Tools similar to *r4ss* which are developed in the future should also consider the divergent needs for interactive exploration of model results and production of written reports. Regular et al. (2020) argue that interactive visualizations support deeper understanding of models, but we have found that the consistent set of figures as stand-alone image files (and associated captions) created by *r4ss* has been valuable for automating the compiling of assessment reports. It is difficult to create a tool that works well for all purposes, so whether interactive or static visualizations (or both) are needed given the purpose of the tool should be carefully considered.

Development of a consistent format for assessment reports across regions and agencies could further simplify the report-writing process by providing all the benefits of shared code discussed above as well as improve understanding of the reports for the many reviewers and researchers who work with stock assessments from multiple regions. For example, standardized static visualizations of patterns in data across 113 rockfish species developed by Anderson et al. (2020) have made these data more accessible to more people through two page reports that can be quickly explored. However, standardizing reports would require a significant effort as the processes for changing the Terms of Reference for stock assessment reports and reviews, the stock assessment modeling platforms, and the tools like *r4ss* that provide the bridge between them, differ among regions and agencies.

Finally, although the *r4ss* code is open source and available for use in any future project, we recommend that any successor to the *r4ss* package should be developed from the ground up, learning from the successes and challenges or *r4ss*, and copying the useful features of *r4ss*, but not utilizing the existing code. This type of restart occurred when Stock Synthesis was converted from FORTRAN to ADMB during Hurricane Isabel in 2003, providing the basis for a long period of utility for a large community of stock assessment scientists. Replacing *r4ss* will take a huge effort, but we have found that the value of *r4ss* has not just come from the tool itself, but the understanding and experience gained and community built by all those who have contributed to this effort.

5. Acknowledgements

We thank Mark Maunder and Simon Hoyle for their role in organizing the IATTC and CAPAM workshops that have led to so much development of *r4ss* as well as our co-authors of the *r4ss* package: Z. Teresa A'mar, Sean C. Anderson, Andrew B. Cooper, LaTreese S. Denson, Robbie L. Emmet, Tommy M. Garrison, Andrea M. Havron, Allan C. Hicks, Watal M. Iwasaki, Neil L. Klaer, Gwladys I. Lambert, Carey R. McGilliard, Cole C. Monnahan, Iago Mosqueira, Kotaro Ono, André E. Punt, Megan M. Stachura, Christine C. Stawitz, Andi Stephens, Yukio Takeuchi, James T. Thorson, Nathan Vaughan, John R. Wallace, and Athol R. Whitten. Finally, we thank the NOAA staff who provided an internal review and the three anonymous reviewers for their suggestions and comments.

6. References

Anderson, S.C., Keppel, E.A., and Edwards, A.M., 2020. Reproducible visualization of raw fisheries data for 113 species improves transparency, assessment efficiency, and monitoring. Fisheries. https://doi.org/10.1002/fsh.10441.

Anderson, S. C., Monnahan, C. C., Johnson, K. F., Ono, K., and Valero, J. L., 2014. ss3sim: an R package for fisheries stock assessment simulation with Stock Synthesis. PLoS One. 9 (4), e92725. https://doi.org/10.1371/journal.pone.0092725

Bentley, N., 2015. Data and time poverty in fisheries estimation: potential approaches and solutions. ICES Journal of Marine Science. 72 (1), 186-193. https://doi.org/10.1093/icesjms/fsu023.

Bocher, E., and Neteler, M., 2012. Geospatial Free and Open Source Software in the 21st Century. Springer, Berlin, Heidelberg. 261 p. https://doi.org/10.1007/978-3-642-10595-1.

Boettiger, C., Chamberlain, S., Hart, E., and Ram, K., 2015. Building software, building community: lessons from the rOpenSci project. Journal of open research software, 3 (1), e8. https://doi.org/10.5334/jors.bu

Brown, L. D., Cai, T. T., and DasGupta, A., 2001. Interval Estimation for a Binomial Proportion. Statistical Science. 16 (2), 101-117. https://www.jstor.org/stable/2676784.

Brown, S. K., Shivlani, M., Koeneke, R. F., Agnew, D., Byrd, J., Cryer, M., Dichmont, C., Die, D., Michaels, W., Reid, K., Rice, J., Sparholt, H., and Weinberg, J., 2020. Patterns and practices in fisheries assessment peer review systems. Marine Policy. 117 (2020), 103880. https://doi.org/10.1016/j.marpol.2020.103880.

Carvalho, F. et al. (submitted to this issue). A Cookbook for Using Model Diagnostics in Integrated Stock Assessments.

Dichmont, C.M., Deng, R.A., Punt, A.E., Brodziak, J., Chang, Y.-J., Cope, J.M., Ianelli, J.N., Legault, C.M., Methot, R.D., Porch, C.E., Prager, M.H., and Shertzer, K.W. 2016. A review of stock assessment packages in the United States. Fisheries Research 183: 447-460.

Fischhoff, B., 2013. The sciences of science communication. Proceedings of the National Academy of Sciences of the United States of America. 110 (3), 14033-14039. https://doi.org/10.1073/pnas.1213273110.

Fournier, D.A., Skaug, H.J., Ancheta, J., Ianelli, J., Magnusson, A., Maunder, M.N., Nielsen, A. and Sibert, J., 2012. AD Model Builder: using automatic differentiation for statistical inference of highly parameterized complex nonlinear models. Optimization Methods and Software, 27(2), pp.233-249. https://doi.org/10.1080/10556788.2011.597854.

Francis, R. C., 2011. Data weighting in statistical fisheries stock assessment models. Canadian Journal of Fisheries and Aquatic Sciences. 68 (6), 1124-1138. https://doi.org/10.1139/F2011-025

Grandin, C.J., K.F. Johnson, A.M. Edwards, and A.M. Berger. 2020. Status of the Pacific Hake (whiting) stock in U.S. and Canadian waters in 2020. Prepared by the Joint Technical Committee of the U.S. and Canada Pacific Hake/Whiting Agreement, National Marine Fisheries Service and Fisheries and Oceans Canada. 273 p.

Hester, J. 2018. “devtools 2.0.0” https://www.tidyverse.org/blog/2018/10/devtools-2-0-0/ (last accessed: 4 August 2020).

Hoyle, S. and M. Maunder, 2006. "Status of yellowfin tuna in the eastern Pacific Ocean in 2004 and outlook for 2005". Inter-American Tropical Tuna Commission Stock Assessment Report 6, pp. 3-102.

Huber, W., Carey, V. J., Gentleman, R., Anders, S., Carlson, M., Carvalho, B. S., Bravo, H. C., Davis, S., Gatto, L. , Girke, T., Gottardo, R., Hahne, F., Hansen, K. D., Irizarry, R. A., Lawrence, M., Love, M. I., MacDonald, J., Obenchain, V., Oles, A., K., Pages, H., Reyes, A., Shannon, P., Smyth, G. K., Tenenbaum, D., Waldron, L., and Morgan, M., 2015. Orchestrating high-throughput genomic analysis with Bioconductor. Nature Methods. 12, 115–121. https://doi.org/10.1038/nmeth.3252.

IPHC 2020. "Stock Assessment" https://www.iphc.int/management/science-and-research/stock-assessment (last accessed: 5 August 2020).

Killcoyne, S., Boyle, J., 2009. Managing chaos: lessons learned developing software in the life sciences. Computing in Science and Engineering. 11 (6), 20-29. https://doi.org/10.1109/MCSE.2009.198.

Li, B. 2020. Stock Assessment Model Comparison Project https://github.com/Bai-Li-NOAA/Stock-Assessment-Model-Comparison-Project (last access: 13 August 2020).

Stewart Lowndes, J. S., Best, B. D., Scarborough, C., Afflerbach, J. C., Frazier, M. R., O'Hara, C. C. Jiang, N., and Halpern, B. S., 2017. Our path to better science in less time using open data science tools. Nature Ecology and Evolution. 1 (6), 1-7. https://doi.org/10.1038/s41559-017-0160.

Maunder, M.N., 2003. Paradigm shifts in fisheries stock assessment: from integrated analysis to Bayesian analysis and back again. Natural Resource Modeling. 16, 465-475. https://doi.org/10.1111/j.1939-7445.2003.tb00123.x

Maunder, M. 2008. Workshop on spatial analysis for stock assessment. Inter-American Tropical Tuna Commission, La Jolla, California, 14–17 October 2007

Maunder, M. N., Piner, K. R., 2015. Contemporary fisheries stock assessment: many issues still remain. ICES Journal of Marine Science. 72 (1), 7-18. https://doi.org/10.1093/icesjms/fsu015

Maunder, M. N., Punt, A. E., 2013. A review of integrated analysis in fisheries stock assessment. Fisheries Research. 142, 61-74. https://doi.org/10.1016/j.fishres.2012.07.025

McAllister, M. K., Ianelli, J. N., 1997. Bayesian stock assessment using catch-age data and the sampling - importance resampling algorithm. Canadian Journal of Fisheries and Aquatic Sciences. 54 (2), 284-300. https://doi.org/10.1139/f96-285

Methot, R. D., Wetzel, C. R., 2013. Stock synthesis: A biological and statistical framework for fish stock assessment and fishery management. Fisheries Research. 142, 86-99. https://doi.org/10.1016/j.fishres.2012.10.012.

Miller, S. K., Anganuzzi, A., Butterworth, D. S., Davies, C. R., Donovan, G. P., Nickson, A., Rademeyer, R. A., Restrepo, V., 2019. Improving communication: the key to more effective MSE processes. Canadian Journal of Fisheries and Aquatic Sciences, 76 (4), 643-656. https://doi.org/10.1139/cjfas-2018-0134

Monnahan, C. C., Branch, T. A., Thorson, J. T., Stewart, I. J., & Szuwalski, C. S., 2019. Overcoming long Bayesian run times in integrated fisheries stock assessments. ICES Journal of Marine Science, 76 (6): 1477-1488. https://doi.org/10.1093/icesjms/fsz059

Monnahan, C. C., and Kristensen, K., 2018. No-U-turn sampling for fast Bayesian inference in ADMB and TMB: Introducing the adnuts and tmbstan R packages. PloS one, 13 (5): e0197954. https://doi.org/10.1371/journal.pone.0197954

PFMC 2009. Terms of reference for the groundfish stock assessment and review process for 2009-2010. Pacific Fishery Management Council, Portland, OR.

PFMC 2019. Terms of reference for the groundfish and coastal pelagic species stock assessment review process for 2019-2020. Pacific Fishery Management Council, Portland, OR. Available from https://www.pcouncil.org/documents/2019/04/terms-of-reference-for-the-groundfish-and-coastal-pelagic-species-stock-assessment-review-process-for-2019-2020-april-2019.pdf/ (last accessed 5 August 2020).

Piner K. R., Lee, H.H., and Maunder, M. N. 2016. Evaluation of using random-at-length observations and an equilibrium approximation of the population age structure in fitting the von Bertalanffy growth function. Fish. Res. 180: 128-137. https://doi.org/10.1016/j.fishres.2015.05.024.

Punt, A. E. 2017. Some insights into data weighting in integrated stock assessments. Fish. Res. 192: 52-65. https://doi.org/10.1016/j.fishres.2015.12.006.

Punt, A. E., Dunn, A., Elvarsson, B., Hampton, J., Hoyle, S. D., Maunder, M. N., Methot, R. D. and Nielsen, A., 2020. Essential features of the next-generation integrated fisheries stock assessment package: A perspective. Fisheries Research 229: 105617. https://doi.org/10.1016/j.fishres.2020.105617.

Punt, A. E., and Hilborn, R., 1997. Fisheries stock assessment and decision analysis: the Bayesian approach. Reviews in Fish Biology and Fisheries. 7, 35–63. https://doi.org/10.1023/A:1018419207494

Punt, A. E., and Maunder, M. N., 2013. Stock Synthesis: Advancing stock assessment application and research through the use of a general stock assessment computer program. Fisheries Research 142, 1-2. https://doi.org/10.1016/j.fishres.2012.11.001

R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Ram, K., Boettiger, C., Chamberlain, S., Ross, N., Salmon, M., and Butland, S., 2019. A Community of practice around peer review for long-term research software sustainability. Computing in Science and Engineering. 21 (2), 59-65. https://doi.org/10.1109/MCSE.2018.2882753.

Regular, P. M., Robertson, G. J., Rogers, R., and Lewis, K. P., 2020. Improving the communication and accessibility of stock assessment using interactive visualization tools. Canadian Journal of Fisheries and Aquatic Sciences. https://doi.org/10.1139/cjfas-2019-0424.

Richards, L. J., Schnute, J. T. and Olsen, N., 1997. Visualizing catch–age analysis: a case study. Canadian Journal of Fisheries and Aquatic Sciences, 54 (7), 1646-1658. https://doi.org/10.1139/f97-073.

Schnute, J. T., Maunder, M. N., and Ianelli, J. N. 2007. Designing tools to evaluate fishery management strategies: can the scientific community deliver? ICES Journal of Marine Science. 64 (6), 1077-1084. https://doi.org/10.1093/icesjms/fsm109

SPC 2010. “MFCL Viewer” https://oceanfish.spc.int/ofpsection/sam/research/272-mfcl-viewer?lang=en (last accessed 4 August 2020).

Stewart, I.J., Hicks, A.C., Taylor, I.G., Thorson, J.T., Wetzel, C. and Kupschus, S., 2013. A comparison of stock assessment uncertainty estimates using maximum likelihood and Bayesian methods implemented with the same model framework. Fisheries Research 142, 37-46.

Stewart, I. J., 2005. Status of the U.S. English sole resource in 2005. In Status of the Pacific Coast Groundfish Fishery through 2005, Stock Assessment and Fishery Evaluation: Stock Assessments and Rebuilding Analyses (Volume II). Pacific Fishery Management Council, Portland, OR. 221 p.

Stewart, I. J., and Hicks, A. C., 2018. Interannual stability from ensemble modelling. Canadian Journal of Fisheries and Aquatic Sciences. 75 (12), 2109-2113. Available from https://doi.org/10.1139/cjfas-2018-0238.

Stewart, I. J., and Hicks, A. 2020. Assessment of the Pacific halibut (*Hippoglossus stenolepis*) stock at the end of 2019. International Pacific Halibut Commission, Seattle, WA. 32 p. Available from https://www.iphc.int/uploads/pdf/sa/2020/iphc-2020-sa-01.pdf (last accessed: 4 August 2020)

Stewart, I. J., and Martell, S. J. D., 2015. Reconciling stock assessment paradigms to better inform fisheries management. ICES Journal of Marine Science. 72 (8), 2187-2196. https://doi.org/10.1093/icesjms/fsv061.

Taylor, I. G., Gertseva, V., Stephens, A. and Bizzarro, J. 2019. Status of Big Skate (*Beringraja binoculata*) Off the U.S. West Coast. Pacific Fishery Management Council, Portland, OR. 184 p. Available from https://www.pcouncil.org/documents/2019/10/status-of-big-skate-beringraja-binoculata-off-the-u-s-pacifc-coast-in-2019-october-2019.pdf (last accessed: 4 August 2020)

Thorson, J. T., Johnson, K. F., Methot, R. D. and Taylor, I. G., 2017. Model-based estimates of effective sample size in stock assessment models using the Dirichlet-multinomial distribution. Fisheries Research. 192, 84-93. https://doi.org/10.1016/j.fishres.2016.06.005

Wickham, H., 2015. R packages: organize, test, document, and share your code. "O'Reilly Media, Inc.". ISBN:978-1-4919-1056-6.

Wilson, G., Aruliah, D. A., Brown, C. T., Hong, N. P. C., Davis, M., Guy, R. T., Haddock, S. H. D., Huff, K. D., Mitchell, I. M., Plumbley, M. D., Waugh, B., White, E. P., and Wilson, P., 2014. Best practices for scientific computing. PLOS Biology, 12 (1), e1001745. https://doi.org/10.1371/journal.pbio.1001745

Wilson, G., Bryan, J., Cranston, K., Kitzes, J., Nederbragt, L., and Teal, T. K., 2017. Good enough practices in scientific computing. PLOS computational biology. 13 (6), e1005510. https://doi.org/10.1371/journal.pcbi.1005510

Tables

Table 1. Issues related to the stock assessment modeling process and potential solutions to alleviate or minimize the issues.

Issue	Solution
Placeholder values for data not available at the start of the assessment process inadvertently retained in the final model.	Frequently make and view plots of data included in input files.
Parameters on bounds in the final model.	The *r4ss* HTML viewer includes a table that highlights parameters on bounds in red so they are easily identified and a plot of the estimated parameter values and associated uncertainty relative to the parameter bounds and prior distribution; consider reconfiguring the model.
Incorrect parameterization of processes because it is difficult to see the resulting form from model input files, e.g., selectivity curve.	Plots of processes resulting from input parameters such as selectivity are automatically available in the *r4ss* HTML viewer; change input parameters until the desired shape is found.
Confounded parameters.	A correlation check is included in the *r4ss* HTML viewer; consider reconfiguring the model.
Small model changes can have cascading effects on the model results that go unnoticed.	The *r4ss* HTML viewer allows examination of the holistic model results and comparison functions

	allow for easy visualization of how results change with each change to input files.
Using custom plotting code not under version control edited by multiple analysts independently can lead to confusion and makes splitting up work difficult.	Use standardized tools under version control.
Sensitivities requested during a review process can be difficult to set up and implement quickly without errors.	Use generalized and standardized tools to modify the model files increases the robustness of results and decreases the likelihood of errors.
Need to quickly update figures and tables when models change.	Use code (e.g., R Markdown or LaTeX) with a standardized plotting tool to generate reports.

Figures

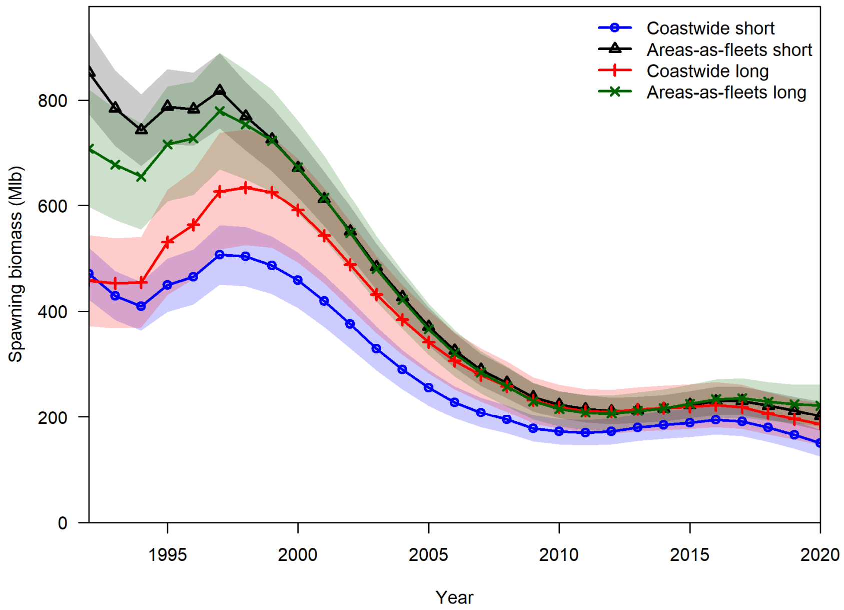

Figure 1. Recent time-series of individual model estimates for female spawning biomass for Pacific halibut created using the *SSplotComparisons* function in *r4ss*.The shaded polygons represent the approximate 95% confidence intervals derived from asymptotic uncertainty estimates from the models.

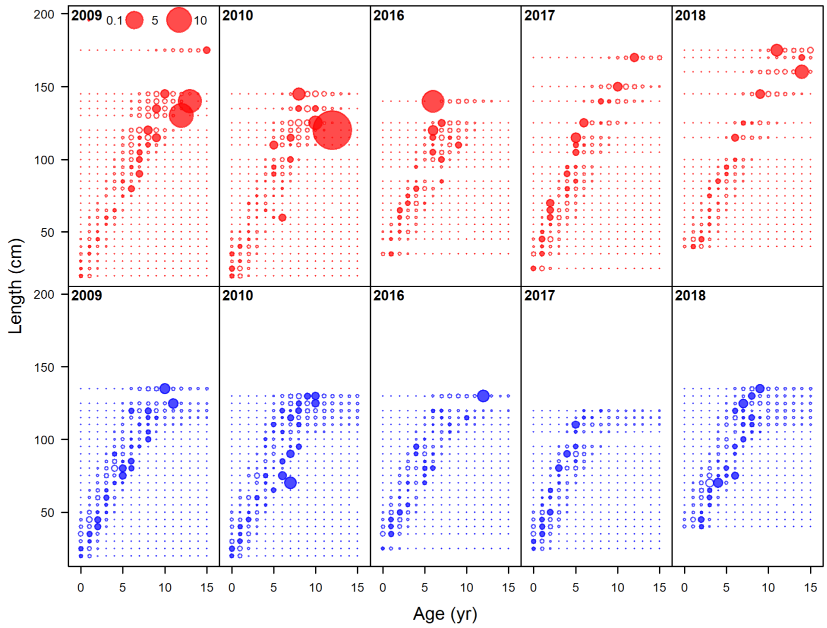

Figure 2. Pearson residuals showing fit of big skate model to conditional age-at-length data. Upper panels are females (red circles) and lower panels males (blue circles). Filled circles represent observations greater than the expectation, while unfilled circles represent observations less than the expectation. Size of the circles (key at top of upper left panel) represent the magnitude of the residual. The year in which the conditional age-at-length data were collected is shown in the upper left corner of each panel.

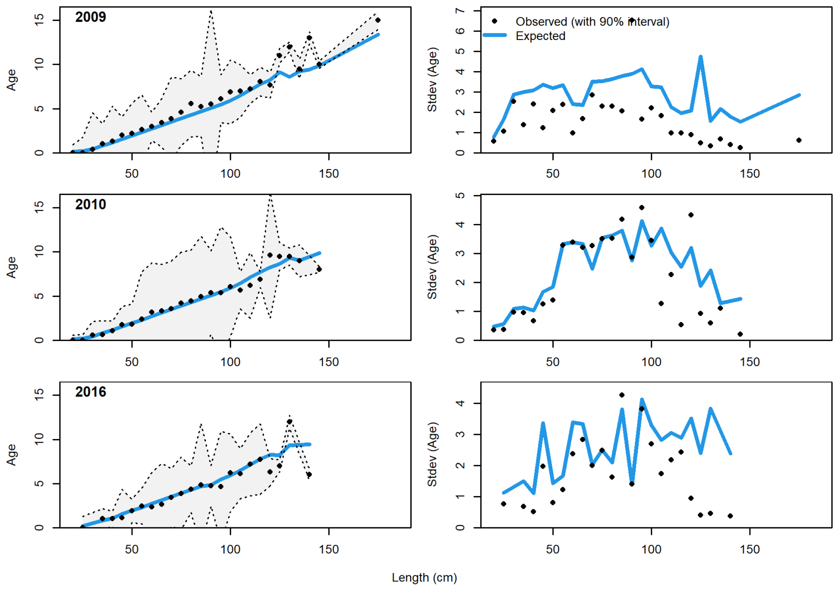

Figure 3. Application of the conditional-age-at-length diagnostic plot for three years of big skate age and length data. The left panels are mean observed age at length by length bin (points) with 90% confidence intervals (shading) calculated from the asymptotic standard errors (SE) compared to the expected mean age at length (solid line) for each year (upper left corner of each left panel). The right panels show the corresponding mean age at length SE values (points) compared to the expected standard deviation (line).

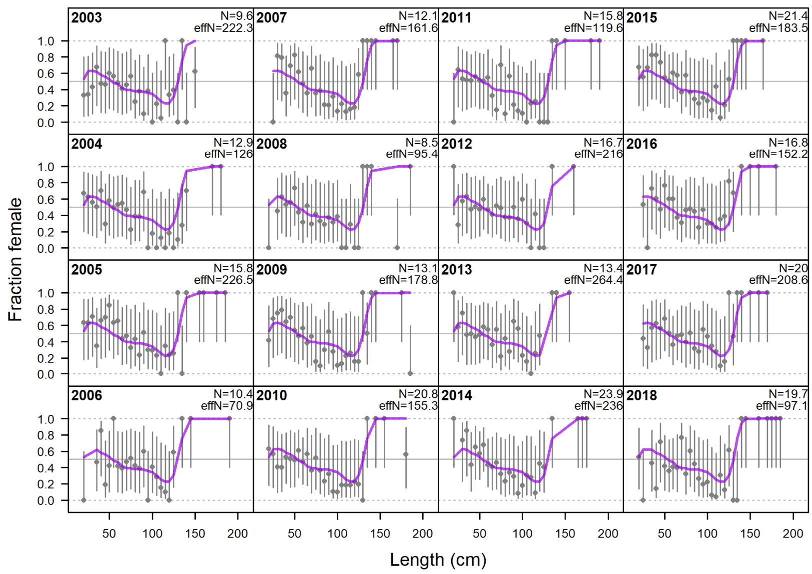

Figure 4. Sex ratios calculated from the bottom trawl survey length composition samples of big skate. Observed sex ratios (points) with 75% intervals (vertical lines) calculated as a Jeffreys interval (Brown et al., 2001). The model expectation is shown in the purple line. The year in which the length composition samples were collected is shown in the upper left corner of each panel. The adjusted input sample size (N) and effective sample size associated with the McAllister-Ianelli tuning (effN) are shown in the upper right corner of each panel.

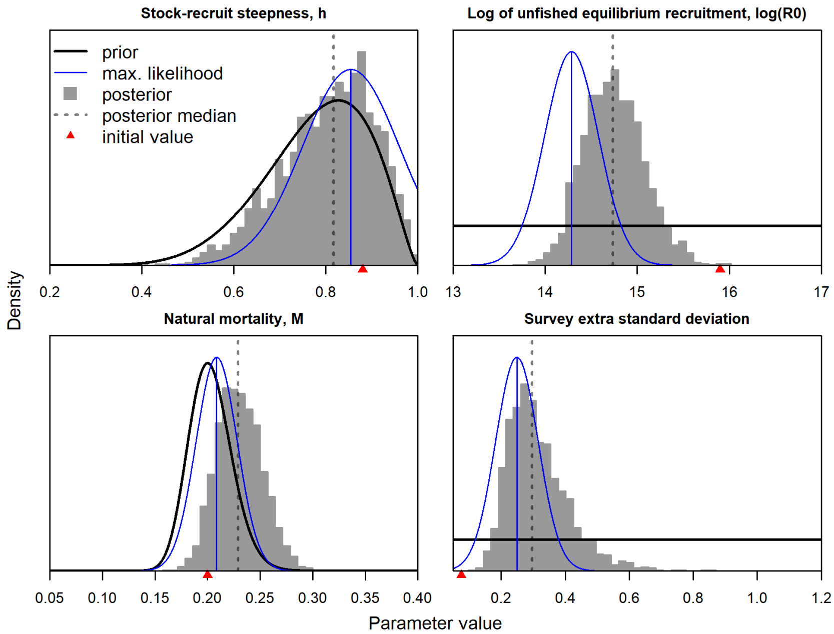

Figure 5. Prior and posterior density estimates for four parameters in the Pacific hake stock assessment. The gray polygons show the distribution of values from 2000 MCMC samples while the blue lines show the maximum posterior density estimate and the normal approximation to the posterior based on the asymptotic approximation of the parameter variance. The black lines show the prior distributions (a beta distribution for steepness, a lognormal distribution for natural mortality, and uniform distributions for the other two parameters). Starting values for the MPD estimation are shown in the red triangles.