A framework for an open source economic model of the systems shaping the mental health of young people

Matthew P Hamilton[1,*] Caroline X Gao[1,2,3] Glen Wiesner[4] John Gillam[2]
Kate M Filia[1,2] Jana M Menssink[1,2] Petra Plencnerova[5] David Baker[1,2]
Patrick D McGorry[1,2] Alexandra Parker[6] Jonathan Karnon[7] Sue M Cotton[1,2]
Cathrine Mihalopoulos[3]

Abstract

Summary: There is strong in principle support for open source health economic models, but practical barriers limit their availability. We propose a set of principles and standards for the implementation of open source health economic models that are TIMELY - Transparent, Iterative, Modular, Epitomised and Yielding. We then describe a software framework that we have developed for developing TIMELY models in youth mental health and illustrate this framework with an open source utility mapping project.

Data: Data

[1] Orygen, Parkville, Australia
[2] Centre for Youth Mental Health; University of Melbourne, Parkville, Australia
[3] School of Public Health and Preventive Medicine, Monash University, Clayton, Australia
[4] Heart Foundation, Melbourne, Australia
[5] headspace National Youth Mental Health Foundation, Melbourne, Australia
[6] Victoria University, Footscray, Australia
[7] Flinders University, Adelaide, Australia

[*] Correspondence: Matthew P Hamilton <matthew.hamilton@orygen.org.au>

1 Introduction

Health economic computational models have become essential tools for health policy development [1,2]. Although influential and widely used, these models routinely contain errors [3], are rarely adequately validated [4], can be difficult to reproduce [5–7] and are likely to be infrequently updated or revised [8]. To help address these issues, there is growing support for greater use of open source health economic models (OSHEMs) that grant liberal permissions to access and re-use model source code [9]. However, to date actual implementations of OSHEMs are rare [10–12]. Barriers to adoption include concerns about intellectual property, confidentiality, model misuse and the resources required to support open source implementations [9,13]. As many health economic models are owned by pharmaceutical companies and consultancies, commercial considerations may also limit the uptake of OSHEMs [12].

Better guidance about how to implement OSHEMs is required [14]. Adherence to explicit standards is as essential requirement for modelling in healthcare [2], but current best practice recommendations for OSHEMs are scarce and piecemeal. Guidelines on health economic model transparency were published ten years ago [15] and made recommendations on documenting models but notably not on the sharing of model code and data. More recent and more general modelling guidance [2] recommends the sharing of code and data through platforms such as GitHub [16] and Zenodo [17] and the use of version control systems such as Git [18]. A coding framework for OSHEMs developed in the language R includes standardised approaches to directory structure and naming conventions [19].

We have consolidated and refined these and other recommended standards for OSHEMs as part of a framework for developing an open source model of youth mental health. In this paper, we describe our motivation for developing the framework, the rationale for each included standard, the software toolkits we have developed to help meet each standard and a worked example of a modelling project developed with the framework.

2 Motivation

2.1 Why develop OSHEMs in mental health

Mental disorders impose high health, social and economic burdens worldwide [20,21]. Much of this burden is potentially avertable [22], but poorly financed and organised mental health systems are ill-equipped for this challenge[23,24]. The large and widespread additional mental health burdens recently observed during the COVID pandemic[25] and predicted as a potential future consequence of global heating [26], highlight the need to improve the resilience and adaptability of these systems. To help stem growing demand for mental health services, policymakers have also been encouraged to place greater emphasis on tackling the social determinants of mental disorder[27].

Open source frameworks have been previously recommended for the development of mental health modelling field [28] but, as with health economics more generally, OSHEMs remain rare. Currently there is only one mental health related model (in Alcohol Use Disorder [29]) that is indexed in the Open Source Models Clearinghouse [10,30]. A Major Depressive Disorder reference model for the United States [31] is also being developed as part of the Open Value Initiative [32]. Greater use of open source approaches could help improve the scope, validity and usefulness of mental health economic models.

Major mental health reform programs, such as those currently being implemented in Victoria, Australia [33], can involve the identification, prioritisation, sequencing, targeting and monitoring of multiple interdependent initiatives. Single purpose models that assume static systems may be inadequate for the decision support needs of policymakers and service planners [34]. Currently, mental health economic models predominantly address issues relating to the affordability and value for money of individual programs [35] with mental health simulation studies rarely modelling complex systems [28]. Systems modelling approaches, that have been recommended for public health economics more generally [36] and for mental health specifically [37], could provide greater insight about inter-dependencies between candidate policies and the dynamic nature of the mental health systems planning context. Dynamic systems methods might also be the basis for developing reference models [38] of mental health systems that are intended for multiple-applications and re-use by different modelling teams. However, more complex models may be more prone to error [39] and models designed for multiple purposes require greater investments in model transparency and validation [12,15].

To retain and improve their utility as decision aids, health economic models require updating and refinement as new evidence emerges and decision contexts change [40]. There are significant deficits in our understanding of the systems in which mental disorder emerges and is treated [41] and longer term project horizons could allow mental health economic models to progressively improve validity as these gaps are addressed. Currently, the theoretical basis for understanding complex mental health systems is weak [42]. Strikingly, it remains unclear why increased investments in mental health care have yet to discernibly reduce the prevalence and burden of mental disorders[43]. The literature about how the requirements, characteristics and performance of mental health services are shaped by spatiotemporal context is underdeveloped [44]. There is insufficient evidence to identify the social determinants of mental disorders most amenable to preventative interventions, and for which population sub-groups such interventions would be most effective [45].

The development, validation and updating of more complex mental health economic models implemented over longer time frames may be too onerous a burden for a single modelling team. Developing networks of modellers working on common health conditions has been recommended as a strategy for improving model validity [14] and a some of us are part of a nascent initiate of this type in mental health [46]. Furthermore, collaborations across multiple modelling teams that include the ability to re-use and extend each others work can make complex modelling projects more tractable [47]. Similarly, developing partnerships between modellers and decision-makers across the life-cycle of a modelling project can help ensure models are appropriately conceptualised and implemented and improve their practical utility as decision aids [36,48].

2.2 readyforwhatsnext

We are currently developing readyforwhatsnext, a reference OSHEM that aims to examine multiple potential population level strategies for promoting mental wellbeing and preventing and treating mental disorders in young people.

Our approach to model development is to undertake a number of discrete modelling projects of the people, places, platforms and programs that shape the mental health and wellbeing of young people and to progressively link them together by means of a common framework. To model people we are developing synthetic representations of populations of interest [49] that describe relevant individual characteristics and their household relationships, algorithms that map psychological measures to health utility [50] and choice models for predicting the helpseeking behaviour of young people. Our in development model of places [51] aims to synthesise geometry and spatial attribute data to characterise the geographic distribution of relevant demographic, environmental, epidemiological and service infrastructure features and our first model of a service platform will represent the processes and operations of a complex primary youth mental health service. We also plan to extend our prior work reviewing economic evidence relating to youth mental health programs [52] so that it can be integrated with the model.

Our initial work on readyforwhatsnext is focused on Victoria, Australia but the framework we are using to develop it is designed to facilitate extension by ourselves and others to different decision contexts. Progress is reported on a project website [53].

3 Framework

The framework we have developed to implement readyforwhatsnext specifies standards for OSHEMs and provides tools for meeting those standards.

3.1 Standards

We have identified 20 standards that we believe are important for quality implementations of OSHEMs, each described under one of following six principles for making models TIMELY:

- **Transparent**: people can easily see how a model has been implemented and tested;
- **Iterative**: a model is routinely updated to maintain and improve validity;
- **Modular**: models and their components can be combined to extend their scope;
- **Epitomized**: a model and its components can be used in multiple decision contexts;
- **Licensed**: a model, its components and derivatives are persistently re-usable by other modellers; and
- **Yielding**: a model can be simply, flexibly and reliably used to inform decision-making.

3.1.1 Transparent Models

A range of tools and practices are available to help make model code and data accessible, comprehensible and citable. The most efficient way to widely disseminate code and data may be to use existing open science infrastructure [2]. Repositories such as Zenodo [17] and Dataverse [54] provide persistent storage solutions that generate a Digital Object Identifier (DOI) for each unique item. These repositories are a preferable solution for sharing citable code and data than transitory repositories such as corporate websites or GitHub where items can be deleted or relocated at any time [55]. Zenodo includes tools that automate integration with GitHub, which makes it easy for developers to maintain parallel code repositories - one for disseminating the most up to date development code and the other for archiving citable code releases.

Model code and data also need to be clearly documented, potentially with different versions for technical and non-technical users [15]. Developers storing data in a Dataverse installation have access to multiple meta-data fields to document both a data collection and its constituent individual files. In R, code manuals and websites can be created with the aid of tools such as devtools [56], sinew [57], roxygen2 [58] and pkgdown [59].

Consistent use of meaningful naming conventions when authoring code is recommended [19,60]. Code can be made easier to follow by using the practices of abstraction [61], where only simple, high level commands are routinely exposed to reviewers, and polymorphsim [62], where the same command (e.g. "simulate") can be reused to implement different algorithms of the same type. Programs to implement model analyses can be made comprehensible to even non-technical users though the use of literate programming techniques that use tools like RMarkdown [63] to render documents that integrate computer code with plain English descriptions.

An essential component of quality assuring health economic models is verification - ensuring that calculations are correct and consistent with model specifications [64]. One useful concept for informing model users about the extensiveness of verification checks is code coverage [65] - the proportion of model code that has been explicitly tested. In R, the testthat [66] and covr [67] tools can be used in conjunction with GitHub to define tests and report coverage metrics.

Finally, transcription errors - mistakes introduced when transferring data between sources, models and reports - are very common in health economic models [3]. The risk of these errors might be lower if there was full transparency across all steps in a study workflow. Scientific computing tools now make it relatively straightforward to author programs that reproducibly execute all steps in data ingest, processing and reporting [60].

Standards:

- **T1: Uniquely identified copies of model code and data are permanently archived in open online repositories**
- **T2: Model code and data are documented**
- **T3: Model code uses a simple and consistent syntax**
- **T4: Model analyses and reporting are implemented using literate programming**
- **T5: Code coverage is reported**
- **T6: All parts of a study analysis and reporting workflow can be reproduced and/or replicated**

3.1.2 Iterative Models

To avoid OSHEMs going stale - losing validity and usefulness with time - they should be routinely updated. A number of tools and approaches can make the process of implementing and curating changes to model code and data more coherent and efficient. Repositories such as Zenodo [17] and Dataverse [54] provide persistent access to all published versions of a dataset, each uniquely identifiable. For code, use of version control tools like Git [18] can ensure that the entire development history of a project is organised so that each version is distinguishable and retrievable by developers. The online platform GitHub [16] can make this version history accessible to anyone.

Adopting semantic versioning [68] conventions can be an efficient way to provide users of model code and data with information about the potential importance of an update. For R code, the usethis [69] package can be used to partially automate version number increments using the convention Major.Minor.Patch.Development. Datasets stored on the Harvard Dataverse use the simpler Major.Minor convention. Continuous integration [70] tools can help verify that each code update passes multiple quality tests. OSHEMs developed in R can take advantage of templates provided by devtools [56] and pkgdown [59] to run continuous integration checks on GitHub. These tests can include those of units (do individual functions produce expected output?), documentation (does documentation render correctly?, can all example workflows be executed?) and installation (can the software be successfully deployed on multiple types of operating system?).

Finally, using deprecation conventions that take an informative and staged approach to retiring old code and data reduces the risk that model revisions have unintended consequences on third party users. The package lifeycle [71] provides tools for R developers to consistently deprecate their code.

Standards:

- **I1: Model code is version controlled**
- **I2: Model code and data use semantic versioning**
- **I3: Continuous integration is used to verify model code updates**
- **I4: Deprecation conventions are used to retire model code and data**

3.1.3 Modular Models

Modular health economic models link multiple self-contained components that can be independently reused and extended by other projects [72,73].

Many types of mental health data are highly sensitive with strict confidentiality requirements. For this reason, not all data included in some mental health models can be made widely available for others to re-use. A modular approach that ensures that model code and data are decoupled (stored in different files) can help model developers to restrict access to confidential model data, while providing open access to all other model components.

An important consideration when combining model components (or modules) is to ensure that interactions between two modules do not compromise the validity of either. Using the coding practice of encapsulation [61] can help ensure that model modules can be safely combined [74].

Standards:

- **M1: Model code and data are stored and managed separately**
- **M2: Model code defines encapsulating data structures**

3.1.4 Epitomised Models

A key challenge to generalising health economic models is that they are typically developed to inform a decision problem with a highly specific jurisdictional context. However, a number of choices about how these models are implemented can significantly increase the re-usability of model code in other contexts.

Writing code as collections of functions (short, self-contained and reusable algorithms that each perform a discrete task) is recommended as good practice for scientific computing [60]. When distributed as libraries (for example, as R packages), functions have the potential to be widely re-used, often in contexts very different than those they were originally developed for. A special type of function called a method can only be applied to a pre-defined class of data structure. Due to the coding concept of inheritance [61], the more restricted nature of methods can be used to enhance the re-usability of model code in different decision contexts [74]. For example, when generalising a model developed for the Australian context to a UK context, one could create a class that initially inherits all of the methods defined for the Australian model and then write new or replacement methods as needed for the UK model.

Whatever type of functions are written for a modelling project, it is good practice to make available test or toy data to demonstrate their use [60].

Standards:

- **E1: Model code is distributed as libraries of classes and functions**
- **E2: Model code defines inheriting data-structures**
- **E3: Test data is available to demonstrate generalised applications of model code**

3.1.5 Licensed Models

To make model code and data widely re-usable by others, it is important to provide users with appropriate and explicit permissions. In the context of open source models, there are two broad categories of licensing options. Some guidance strongly recommends the use of permissive licensing [60] that provides users with great flexibility as to the purposes (including commercial) for which the content could be re-used. An alternative approach is to use copyleft licenses [75] that can require content users to distribute any derivative works they create under similar open source arrangements.

For code, it may be appropriate to adopt the prevailing open source licensing practice within the programming language being used. Applying a previously published algorithm [76] to analyse the most comprehensive archive of released R packages [77] finds that `r`gpl_pc_dbl‘% are distributed under various forms of General Public License (GPL) [78], a copyleft license.

For data, it may not be sufficient to simply choose between a permissive license like the Public Domain Dedication (CC0) [79] or a copyleft option such as the Attribution-Share Alike (CC-BY-SA) [80]. Responsible custodianship of some de-identified mental health data may involve using or adapting template terms of use [81] which have a number of ethical clauses (for example, prohibiting efforts to re-identify research participants).

Licenses may or may not specify that model re-users must give appropriate acknowledgement to model authors. Citation tools can be distributed with each cite individual code or data items to inform re-users of the desired attribution. In R, including a CITATION file in the `inst` directory of a package will enable users of that package to retrieve citation information by running a command of the format `citation("Package Name")` in the R console [82]. More generally, including a CITATION.cff file at the top level of your code repository will enable GitHub and Zenodo repositories hosting that item to include the relevant information in their citation tools [83]. Datasets hosted on Dataverse installations have metadata fields that, once completed by authors, generate citation files for dataset viewers.

Standards:

- **L1: Model code is made available for re-use under copyleft licenses**
- **L2: Non-confidential model data is available for liberal re-use (subject to additional terms for de-identified human data)**
- **L3: Model code and data are distributed with tools to support appropriate citation**

3.1.6 Yielding Models

OSHEMs can be time and skills intensive for modellers to develop - but they should be easy for others to use.

Statistical models are a common output of health economic evaluations, but they are often not reported in a format that enables others to confidently and reliably re-use them [84]. Open source approaches can help address this by disseminating code artefacts that enable easy and appropriate use of a statistical model to make predictions with new data. However, great care must be exercised when doing so if models are derived from data on human subjects. In the R language, model objects by default typically contain a copy of the source dataset. Such dataset copies must therefore be replaced (for example, with fake data) and the amended artefact's predictive performance then retested before any public release.

Another way to make OSHEMs easier to use is to develop simple user-interfaces for non-technical users. In R, such user-interfaces are typically developed with the Shiny package, for which a tutorial aimed at health economists is available [85].

Standards:

- **Y1: Statistical models are distributed with validated tools to support their safe and appropriate re-use**
- **Y2: Simple user-interfaces allow non-technical users to configure and run models**

3.2 Modelling tools

We have developed a toolkit to help streamline the process of developing OSHEMs that meet the TIMELY standards. The toolkit is comprised of online repositories and software.

3.2.1 Framework repositories

We created a GitHub organisation where all framework software source code is stored, documented, version controlled and continuously integrated [86]. To store citable archived copies of release copies of our software, we created a Zenodo community [87]. Finally, to manage datasets for use in models developed with the framework, we created a dedicated collection within the Harvard Dataverse [88].

3.2.2 Framework software

As a foundation for implementing the framework, we authored a development version R package that defines a novel syntax and a template class for model modules [89]. To enable the syntax and module template be applied to modelling projects, we then created five additional development version R packages that provide tools for authoring:

- self-documenting model modules [90];
- self-documenting functions (including methods), written in a consistent house style [91];
- citable, quality assured R packages [92];
- model datasets [93];
- model analyses and reports [94].

The six R packages, their primary focus, the TIMELY standards they support and the third party packages they depend on are summarised in Table 1. When used in conjunction with framework repositories, the six packages currently provide strong support for implementing 17 of the TIMELY standards, weak support for implementing the standards relating to disseminating statistical models (Y1) and user-interface development (Y2) and no support for implementing standard T5 (reporting code coverage). Our future development priorities for these packages include adding functionality so that all 20 TIMELY standards are well supported and to then submit production versions of each package for review by and archiving on the Comprehensive R Archive Network (CRAN) [77].

Table 1: Framework software to help implement TIMELY standards

Package	Ref	Focus	Standard	Depends on
ready4	[89]	Syntax	T3 M3	assertthat bib2df dataverse dplyr fs generics Hmisc kableExtra knitr lifecycle magrittr methods piggyback purrr rlang rvest stats stringi stringr testit testthat tibble tools utils
ready4fun	[91]	Modules	T2-3 I4 E1	desc devtools dplyr Hmisc knitr lifecycle lubridate magrittr methods piggyback pkgdown purrr readxl ready4 ready4show ready4use rlang sinew stats stringi stringr testit testthat tibble tidyr tools usethis utils xfun
ready4class	[90]	Functions	T2-3 M2 E1-2	devtools dplyr fs gtools Hmisc knitr lifecycle magrittr methods purrr ready4 ready4fun ready4show rlang stats stringi stringr testit testthat tibble tidyr usethis utils
ready4pack	[92]	Libraries	T1 I1-3 M1-2 E1 L1,3	dataverse dplyr knitr lifecycle magrittr methods purrr ready4 ready4class ready4fun rlang stringr testthat tibble tidyr utils
ready4use	[93]	Data	T1-2 M1 E3 L2-3	data.table dataverse dplyr fs Hmisc knitr lifecycle magrittr methods piggyback purrr readxl ready4 ready4show rlang stats stringi stringr testit testthat tibble tidyr utils
ready4show	[94]	Reporting	T2,4,6 Y1-2	dataverse DescTools dplyr flextable grDevices here Hmisc kableExtra knitr knitrBootstrap lifecycle magrittr methods officer purrr ready4 rlang rmarkdown stringi stringr testthat tibble tidyr utils xtable

4 Application

Worked example

Table 2: TIMELY Checklist applied to utility mapping study

	Standard	Met?	Description
T1	Publicly archived	F	5 libraries (youthvars, scorz, specific, TTU and youthu) [95–99], 3 programs [100–102], 2 sub-routines [103,104] and 2 datasets [49,105] are permanently archived with unique identifiers.
T2	Documented	F	All code libraries have documenting websites with URLs that concatenate 'https://ready4-dev.github.io/' and the package name (e.g. https://ready4-dev.github.io/youthvars). All three Markdown programs are self-documenting, with one [100] including additional instructions in a README file. Only one sub-routine [104] is documented with a meaningful README file. All datasets have meaningful metadata descriptors.
T3	Consistent syntax	F	All libraries, programs and sub-routines use the same house style. All libraries except [99] use framework syntax, as does one program [100].
T4	Literately programmed	F	All programs use literate programming.
T5	Code coverage	F	No current reporting of code coverage.
T6	Reproducible	F	All parts of the study workflow from raw data ingest through to data processing, analysis, reporting and dissemination of study outputs can be reproduced (if granted access to source data) or replicated (using supplied synthetic data) with one program [100].
I1	Version controlled	F	All code is version controlled using Git and GitHub. All source code is available in one of two GitHub organisations [86,106].
I2	Semanticly versioned	F	Semantic version is used in all code. As no code library has yet been submitted to CRAN, only the development version extensions of each version number have been incremented to date.
I3	Continuously integrated	F	All six libraries use continuous integration (CI). CI results for each library can be viewed at URL that concatenates 'https://github.com/ready4-dev/' the package name and '/actions' (e.g. https://github.com/ready4-dev/youthvars/actions)
I4	Deprecation	F	Retired code is deprecated using lifecycle package tools (e.g. review everything after "## DEPRECATED FNS" in https://github.com/ready4-dev/youthvars/blob/main/data-raw/fns/add.R . Package vignettes and datasets are also deprecated e.g. https://ready4-dev.github.io/youthvars/articles/Replication_DS.html)
M1	Separate code and data	F	All development code is stored on repos in two GitHub organisations [86,106] and all archived releases are available in aZenodo community [87]. All non-confidential data is stored in repositories within a Harvard Dataverse collection [88].
M2	Encapsulated	F	Four [95–98] out of five libraries include encapsulated modules. Examples are the items beginning with Scorz, Specific and Youthvars that are listed in this table: https://ready4-dev.github.io/ready4/articles/V_01.html#current-ready4-framework-modules as well as the S4 classess from the TTU package listed here: https://ready4-dev.github.io/TTU/reference/index.html
E1	Libraries	F	All code libraries include functions. The most complete list of functions for each library is available by clicking the 'Manual - Developer (PDF)' link on each package's documentation homepage (see item T2 above).

	Standard	Met?	Description
E2	Inheriting	F	All modules (see item M2) inherit from the Ready4Module class.
E3	Test data	F	Two synthetic datasets and their data dictionaries are publicly available in a data repository [49]. One (ymh_clinical_tb.RDS) closely resembles the study dataset and was released so that the main study algorithm [100] can be rerun by those without access to the confidential data. The other (eq5d_ds_dict.RDS) is deliberately different to the source dataset in both variable naming convention and the concepts used for predictors and outcome measures and was created to demonstrate generalised applications of study algorithms.
L1	Copyleft code	F	All code libraries, programs and sub-routines use GPL-3 licenses.
L2	Liberal, safe data terms	F	Datasets use amended version of template provided by Harvard Dataverse [81].
L3	Citation tools	F	All libraries have CITATION file in inst directory. All code repositories have CITATION.cff file. All datasets have citation generating metadata.
Y1	Prediction tools	F	Model catalogues (PDF files beginning with 'AAA_TTU_MDL_CTG') describe the predictive performance of all models under a variety of usage regimes and are available in the study results dataset [105]. The youthu library [99] includes tools for searching for and applyihg models compatible with different types of input data. An example program to demonstrate this functionality is available in RMarkdown [101] and rendered PDF formats (the 'Application.pdf' file in the study results dataset [105]).
Y2	User interface	F	No Shiny app user interface has yet been developed.

5 Discussion

MH systems design is not a pharma led project - less concerns about commercial ownership

greater use of these types of models may require adaptation on the part of funders, modellers and decision-makers. T

Availability of data and materials

Ethics approval

Details on ethics approvals go here.

Funding

The study was funded by Orygen, VicHealth and Victoria University.

Conflict of Interest

None declared.

References

1. Dakin H, Devlin N, Feng Y, Rice N, O'Neill P, Parkin D. The influence of cost-effectiveness and other factors on nice decisions. Health economics. Wiley Online Library; 2015;24: 1256–1271.
2. Erdemir A, Mulugeta L, Ku JP, Drach A, Horner M, Morrison TM, et al. Credible practice of modeling and simulation in healthcare: Ten rules from a multidisciplinary perspective. Journal of translational medicine. 2020;18: 369. doi:10.1186/s12967-020-02540-4
3. Radeva D, Hopkin G, Mossialos E, Borrill J, Osipenko L, Naci H. Assessment of technical errors and validation processes in economic models submitted by the company for NICE technology appraisals. International Journal of Technology Assessment in Health Care. 2020;36: 311–316. doi:10.1017/S0266462320000422
4. Ghabri S, Stevenson M, Möller J, Caro JJ. Trusting the results of model-based economic analyses: Is there a pragmatic validation solution? Pharmacoeconomics. 2019;37: 1–6. doi:10.1007/s40273-018-0711-9
5. Jalali MS, DiGennaro C, Guitar A, Lew K, Rahmandad H. Evolution and reproducibility of simulation modeling in epidemiology and health policy over half a century. Epidemiologic Reviews. 2021;43: 166–175. doi:10.1093/epirev/mxab006
6. McManus E, Turner D, Sach T. Can you repeat that? Exploring the definition of a successful model replication in health economics. Pharmacoeconomics. 2019;37: 1371–1381. doi:10.1007/s40273-019-00836-y
7. Bermejo I, Tappenden P, Youn J-H. Replicating health economic models: Firm foundations or a house of cards? PharmacoEconomics. 2017;35: 1113–1121. doi:10.1007/s40273-017-0553-x
8. Sampson CJ, Wrightson T. Model registration: A call to action. PharmacoEconomics - Open. 2017;1: 73–77. doi:10.1007/s41669-017-0019-2
9. Pouwels X, Sampson CJ, Arnold RJG. Opportunities and barriers to the development and use of open source health economic models: A survey. Value Health. 2022;25: 473–479. doi:10.1016/j.jval.2021.10.001
10. Emerson J, Bacon R, Kent A, Neumann PJ, Cohen JT. Publication of decision model source code: Attitudes of health economics authors. PharmacoEconomics. 2019;37: 1409–1410. doi:10.1007/s40273-019-00796-3
11. Michalczyk J, Clay E, Pochopien M, Aballea S. PRM123 - AN OVERVIEW OF OPEN-SOURCE MODELS IN HEALTH ECONOMICS. Value in Health. 2018;21: S377. doi:10.1016/j.jval.2018.09.2243
12. Feenstra T, Corro-Ramos I, Hamerlijnck D, Voorn G van, Ghabri S. Four aspects affecting health economic decision models and their validation. PharmacoEconomics. 2022;40: 241–248. doi:10.1007/s40273-021-01110-w
13. Wu EQ, Zhou Z-Y, Xie J, Metallo C, Thokala P. Transparency in health economic modeling: Options, issues and potential solutions. PharmacoEconomics. 2019;37: 1349–1354. doi:10.1007/s40273-019-00842-0
14. Sampson CJ, Arnold R, Bryan S, Clarke P, Ekins S, Hatswell A, et al. Transparency in decision modelling: What, why, who and how? PharmacoEconomics. 2019;37: 1355–1369. doi:10.1007/s40273-019-00819-z
15. Eddy DM, Hollingworth W, Caro JJ, Tsevat J, McDonald KM, Wong JB. Model transparency and validation: A report of the ISPOR-SMDM modeling good research practices task force-7. Med Decis Making. 2012;32: 733–43. doi:10.1177/0272989x12454579
16. github. GitHub [Internet]. 2007. Available: https://github.com/
17. European Organization For Nuclear Research, OpenAIRE. Zenodo [Internet]. CERN; 2013. doi:10.25495/7GXK-RD71

18. git. Git [Internet]. Available: https://git-scm.com/

19. Alarid-Escudero F, Krijkamp EM, Pechlivanoglou P, Jalal H, Kao S-YZ, Yang A, et al. A need for change! A coding framework for improving transparency in decision modeling. PharmacoEconomics. 2019;37: 1329–1339. doi:10.1007/s40273-019-00837-x

20. Bloom DE, Cafiero ET, Jané-Llopis E, Abrahams-Gessel S, Bloom LR, Fathima S, et al. The global economic burden of noncommunicable diseases. 91-93 route de la Capite,CH-1223 Cologny/Geneva,Switzerland: World Economic Forum.; 2011.

21. Global, regional, and national burden of 12 mental disorders in 204 countries and territories, 1990–2019: A systematic analysis for the global burden of disease study 2019. The Lancet Psychiatry. 2022;9: 137–150. doi:10.1016/S2215-0366(21)00395-3

22. Chisholm D, Sweeny K, Sheehan P, Rasmussen B, Smit F, Cuijpers P, et al. Scaling-up treatment of depression and anxiety: A global return on investment analysis. The Lancet Psychiatry. 2016; doi:10.1016/s2215-0366(16)30024-4

23. Saxena S, Thornicroft G, Knapp M, Whiteford H. Resources for mental health: Scarcity, inequity, and inefficiency. The Lancet. 370: 878–889. doi:10.1016/S0140-6736(07)61239-2

24. Whiteford H, Ferrari A, Degenhardt L. Global burden of disease studies: Implications for mental and substance use disorders. Health Affairs. 2016;35: 1114–1120. doi:10.1377/hlthaff.2016.0082

25. Santomauro DF, Mantilla Herrera AM, Shadid J, Zheng P, Ashbaugh C, Pigott DM, et al. Global prevalence and burden of depressive and anxiety disorders in 204 countries and territories in 2020 due to the COVID-19 pandemic. The Lancet. 2021;398: 1700–1712. doi:https://doi.org/10.1016/S0140-6736(21)02143-7

26. Page LA, Howard LM. The impact of climate change on mental health (but will mental health be discussed at copenhagen?). Psychological Medicine. Cambridge University Press; 2010;40: 177–180. doi:10.1017/S0033291709992169

27. Organization WH, Foundation CG. Social determinants of mental health. Geneva: World Health Organization; 2014.

28. Long KM, Meadows GN. Simulation modelling in mental health: A systematic review. Journal of Simulation. 2017; doi:10.1057/s41273-017-0062-0

29. Clearinghouse C. Basu, kim: Alcohol use disorder [Internet]. OSF; 2018. Available: osf.io/jvayu

30. Evaluation of Value C for the, Health R in. Open-source model clearinghouse [Internet]. Tufts Medical Center; Available: http://ghcearegistry.org/orchard/open-source-model-clearinghouse

31. Innovation T, Initiative V. IVI-MDD value model [Internet]. 2022. Available: https://www.thevalueinitiative.org/ivi-mdd-value-model/

32. Jansen JP, Incerti D, Linthicum MT. Developing open-source models for the US health system: Practical experiences and challenges to date with the open-source value project. PharmacoEconomics. 2019;37: 1313–1320. doi:10.1007/s40273-019-00827-z

33. Victoria S of. Royal commission into victoria's mental health system final report [Internet]. State of Victoria; 2021. Available: https://finalreport.rcvmhs.vic.gov.au/download-report/

34. Commission P. Mental health: Productivity commission inquiry report [Internet]. Productivity Commission; 2020. Available: https://apo.org.au/node/309475

35. Knapp M, Wong G. Economics and mental health: The current scenario. World Psychiatry. 2020;19: 3–14. doi:10.1002/wps.20692

36. Squires H, Chilcott J, Akehurst R, Burr J, Kelly MP. A framework for developing the structure of public health economic models. Value in Health. 2016;19: 588–601. doi:https://doi.org/10.1016/j.jval.2016.02.011

37. Occhipinti JA, Skinner A, Doraiswamy PM, Fox C, Herrman H, Saxena S, et al. Mental health: Build predictive models to steer policy. Nature. 2021;597: 633–636. doi:10.1038/d41586-021-02581-9

38. Afzali HH, Karnon J, Merlin T. Improving the accuracy and comparability of model-based economic evaluations of health technologies for reimbursement decisions: A methodological framework for the development of reference models. Med Decis Making. 2013;33: 325–32. doi:10.1177/0272989x12458160

39. Saltelli A. A short comment on statistical versus mathematical modelling. Nature Communications. 2019;10: 3870. doi:10.1038/s41467-019-11865-8

40. Jenkins DA, Martin GP, Sperrin M, Riley RD, Debray TPA, Collins GS, et al. Continual updating and monitoring of clinical prediction models: Time for dynamic prediction systems? Diagnostic and Prognostic Research. 2021;5: 1. doi:10.1186/s41512-020-00090-3

41. Fried EI, Robinaugh DJ. Systems all the way down: Embracing complexity in mental health research. BMC Medicine. 2020;18: 205. doi:10.1186/s12916-020-01668-w

42. Langellier BA, Yang Y, Purtle J, Nelson KL, Stankov I, Diez Roux AV. Complex systems approaches to understand drivers of mental health and inform mental health policy: A systematic review. Administration And Policy In Mental Health. 2018; doi:10.1007/s10488-018-0887-5

43. Jorm AF, Patten SB, Brugha TS, Mojtabai R. Has increased provision of treatment reduced the prevalence of common mental disorders? Review of the evidence from four countries. World psychiatry : official journal of the World Psychiatric Association (WPA). 2017;16: 90–99. doi:10.1002/wps.20388

44. Furst MA, Gandré C, Romero López-Alberca C, Salvador-Carulla L. Healthcare ecosystems research in mental health: A scoping review of methods to describe the context of local care delivery. BMC Health Services Research. 2019;19: 173. doi:10.1186/s12913-019-4005-5

45. Alegría M, NeMoyer A, Falgàs Bagué I, Wang Y, Alvarez K. Social determinants of mental health: Where we are and where we need to go. Current Psychiatry Reports. 2018;20: 95–95. doi:10.1007/s11920-018-0969-9

46. Mental Health Research QC for. National mental health systems modelling network [Internet]. 2022. Available: https://qcmhr.org/research/research-streams/mental-health-services-research/national-mental-health-systems-modelling-network/

47. Arnold RJG, Ekins S. Time for cooperation in health economics among the modelling community. PharmacoEconomics. 2010;28: 609–613. doi:10.2165/11537580-000000000-00000

48. Zabell T, Long KM, Scott D, Hope J, McLoughlin I, Enticott J. Engaging healthcare staff and stakeholders in healthcare simulation modeling to better translate research into health impact: A systematic review. Frontiers in Health Services. 2021;1. doi:10.3389/frhs.2021.644831

49. Hamilton MP. Synthetic (fake) youth mental health datasets and data dictionaries [Internet]. Harvard Dataverse; 2021. doi:10.7910/DVN/HJXYKQ

50. Hamilton MP, Gao CX, Filia KM, Menssink JM, Sharmin S, Telford N, et al. Predicting quality adjusted life years in young people attending primary mental health services. medRxiv. Cold Spring Harbor Laboratory Press; 2021; doi:10.1101/2021.07.07.21260129

51. Hamilton M. Springtides reports for Local Government Areas in the North West of Melbourne [Internet]. Harvard Dataverse; 2022. doi:10.7910/DVN/V3OKZV

52. Hamilton MP, Hetrick SE, Mihalopoulos C, Baker D, Browne V, Chanen AM, et al. Identifying attributes of care that may improve cost-effectiveness in the youth mental health service system. Med J Aust. 2017;207: S27–S37. doi:10.5694/mja17.00972

53. Orygen. ready4-dev - documenting the development of an open souce youth mental health systems model [Internet]. Available: https://ready4-dev.com/

54. Quantitative Social Science I for. Dataverse [Internet]. Harvard University; 2007. Available: https://dataverse.org

55. Data, Data Editors CG by. Frequently asked questions [Internet]. Available: https://social-science-data-editors.github.io/guidance/FAQ.html

56. Wickham H, Hester J, Chang W, Bryan J. Devtools: Tools to make developing r packages easier [Internet]. 2021. Available: https://CRAN.R-project.org/package=devtools

57. Sidi J. Sinew: Package development documentation and namespace management [Internet]. 2022. Available: https://CRAN.R-project.org/package=sinew
58. Wickham H, Danenberg P, Csárdi G, Eugster M. roxygen2: In-line documentation for r [Internet]. 2021. Available: https://CRAN.R-project.org/package=roxygen2
59. Wickham H, Hesselberth J, Salmon M. Pkgdown: Make static HTML documentation for a package [Internet]. 2022. Available: https://CRAN.R-project.org/package=pkgdown
60. Wilson JAC Greg AND Bryan. Good enough practices in scientific computing. PLOS Computational Biology. Public Library of Science; 2017;13: 1–20. doi:10.1371/journal.pcbi.1005510
61. Hourani H, Wasmi H, Alrawashdeh T. A code complexity model of object oriented programming (OOP). 2019 IEEE jordan international joint conference on electrical engineering and information technology (JEEIT). 2019. pp. 560–564. doi:10.1109/JEEIT.2019.8717448
62. Milojkovic N, Caracciolo A, Lungu MF, Nierstrasz O, Röthlisberger D, Robbes R. Polymorphism in the spotlight: Studying its prevalence in java and smalltalk. 2015 IEEE 23rd international conference on program comprehension. 2015. pp. 186–195. doi:10.1109/ICPC.2015.29
63. Xie Y, Allaire JJ, Grolemund G. R markdown: The definitive guide. Chapman; Hall/CRC; 2018.
64. Büyükkaramikli NC, Rutten-van Mölken MPMH, Severens JL, Al M. TECH-VER: A verification checklist to reduce errors in models and improve their credibility. PharmacoEconomics. 2019;37: 1391–1408. doi:10.1007/s40273-019-00844-y
65. Eric Wong W, Debroy V, Choi B. A family of code coverage-based heuristics for effective fault localization. Journal of Systems and Software. 2010;83: 188–208. doi:https://doi.org/10.1016/j.jss.2009.09.037
66. Wickham H. Testthat: Get started with testing. The R Journal. 2011;3: 5–10. Available: https://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf
67. Hester J. Covr: Test coverage for packages [Internet]. 2020. Available: https://CRAN.R-project.org/package=covr
68. Preston-Werner T. Semantic versioning 2.0.0 [Internet]. 2022. Available: https://semver.org
69. Wickham H, Bryan J, Barrett M. Usethis: Automate package and project setup [Internet]. 2021. Available: https://CRAN.R-project.org/package=usethis
70. Shahin M, Ali Babar M, Zhu L. Continuous integration, delivery and deployment: A systematic review on approaches, tools, challenges and practices. IEEE Access. 2017;5: 3909–3943. doi:10.1109/ACCESS.2017.2685629
71. Henry L, Wickham H. Lifecycle: Manage the life cycle of your package functions [Internet]. 2021. Available: https://CRAN.R-project.org/package=lifecycle
72. Trauer JM, Ragonnet R, Doan TN, McBryde ES. Modular programming for tuberculosis control, the "AuTuMN" platform. BMC Infectious Diseases. 2017;17: 546. doi:10.1186/s12879-017-2648-6
73. Urach C, Zauner G, Endel G, Wilbacher I, Breitenecker F. A modular simulation model for assessing interventions for abdominal aortic aneurysms. 2013 winter simulations conference (WSC). 2013. pp. 66–76. doi:10.1109/WSC.2013.6721408
74. Hamilton M. Apply an object-oriented paradigm to computational models of mental health systems [Internet]. 2022. Available: https://ready4-dev.github.io/ready4/articles/V_03.html
75. Foundation TFS. What is copyleft? [Internet]. Available: https://www.gnu.org/copyleft/
76. Kross S. How-r-packages-are-licensed [Internet]. 2016. Available: https://seankross.com/2016/08/02/How-R-Packages-are-Licensed.html
77. Statistical Computing RF for. The comprehensive r archive network [Internet]. 2022. Available: https://cran.r-project.org

78. Foundation TFS. Licenses [Internet]. 2022. Available: https://www.gnu.org/licenses

79. Commons C. CC0 1.0 universal [Internet]. 2022. Available: https://creativecommons.org/publicdomain/zero/1.0/legalcode

80. Commons C. Attribution-ShareAlike 4.0 international [Internet]. 2022. Available: https://creativecommons.org/licenses/by-sa/4.0/legalcode

81. Quantitative Social Science I for. Sample data usage agreement [Internet]. 2022. Available: https://support.dataverse.harvard.edu/sample-data-usage-agreement

82. Salmon M, Chamberlain S, Ram K. Make your r package easier to cite [Internet]. 2021. Available: https://ropensci.org/blog/2021/02/16/package-citation/

83. Druskat S, Spaaks JH, Chue Hong N, Haines R, Baker J, Bliven S, et al. Citation File Format. 2021. doi:10.5281/zenodo.5171937

84. Kearns B, Ara R, Wailoo A, Manca A, Alava MH, Abrams K, et al. Good practice guidelines for the use of statistical regression models in economic evaluations. PharmacoEconomics. 2013;31: 643–652. doi:10.1007/s40273-013-0069-y

85. Smith R, Schneider P. Making health economic models shiny: A tutorial. Wellcome Open Res. 2020;5: 69. doi:10.12688/wellcomeopenres.15807.2

86. Orygen. ready4: A suite of authoring, modelling and prediction tools for exploring topics in young people's mental health [Internet]. 2022. Available: https://github.com/ready4-dev/

87. Orygen. ready4: Open and modular mental health systems models [Internet]. 2022. Available: https://zenodo.org/communities/ready4

88. Orygen. ready4: Open and modular mental health systems models [Internet]. 2022. Available: https://dataverse.harvard.edu/dataverse/ready4)

89. Hamilton MP. ready4: Implement open science computational models of mental health systems [Internet]. 2021. doi:10.5281/zenodo.5606250

90. Hamilton M, Wiesner G. ready4class: Author Ready4 framework modules [Internet]. 2022. doi:10.5281/zenodo.5640313

91. Hamilton M, Wiesner G. ready4fun: Author and document functions that extend the Ready4 framework [Internet]. 2022. doi:10.5281/zenodo.5611779

92. Hamilton M. ready4pack: Author r packages that extend the Ready4 framework [Internet]. 2022. doi:10.5281/zenodo.5644322

93. Hamilton M, Wiesner G. ready4use: Author, label and share Ready4 framework datasets [Internet]. 2022. doi:10.5281/zenodo.5644336

94. Hamilton M, Wiesner G. ready4show: Author literate programs to share insights from applying the Ready4 framework [Internet]. 2022. doi:10.5281/zenodo.5644568

95. Hamilton M, Gao C. youthvars: Describe and Validate Youth Mental Health Datasets [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6084467

96. Hamilton M, Gao C. Scorz: Score questionnaire item responses [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6084824

97. Hamilton M, Gao C. specific: Specify Candidate Models for Representing Mental Health Systems [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6116701

98. Gao C, Hamilton M. TTU: Implement Transfer to Utility Mapping Algorithms [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6130155

99. Hamilton MP, Gao CX. Youthu: Transform youth outcomes to health utility predictions [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6210978

100. Hamilton M, Gao C. Complete study program to reproduce all steps from data ingest through to results dissemination for a study to map mental health measures to AQoL-6D health utility [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6212704

101. Hamilton M, Gao C. aqol6dmap_use: Apply AQoL-6D Utility Mapping Models To New Data [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6416330

102. Hamilton MP. aqol6dmap_fakes: Generate fake input data for an AQoL-6D mapping study [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6321821

103. Hamilton M. ttu_mdl_ctlg: Generate a template utility mapping (transfer to utility) model catalogue [Internet]. Zenodo; 2022. doi:10.5281/zenodo.6116385

104. Hamilton MP. ready4-dev/ttu_lng_ss: Create a Draft Scientific Manuscript For A Utility Mapping Study [Internet]. Zenodo; 2022. doi:10.5281/zenodo.5976988

105. Hamilton MP, Gao CX, Filia KM, Menssink JM, Sharmin S, Telford N, et al. Transfer to AQoL-6D Utility Mapping Algorithms [Internet]. Harvard Dataverse; 2021. doi:10.7910/DVN/DKDIB0

106. Orygen. Orygen github organisation [Internet]. 2022. Available: https://github.com/orygen/

A Appendix