

# assertHE: an R package to improve quality assurance of HTA models

Robert A Smith<sup>\*1,2</sup>, Yevgeniy Samyshkin<sup>3</sup>, Wael Mohammed<sup>1,2</sup>, Tom Ward<sup>3</sup>, Calum Lewis<sup>1,2</sup>, ...<sup>\*</sup>, and Baris Deniz<sup>3</sup>

<sup>1</sup>Dark Peak Analytics, Sheffield, UK

<sup>2</sup>SCHARR, Regents Court, University of Sheffield, Sheffield, UK, S1 4DA

<sup>3</sup>GSK, GSK House, 980 Great West Road, Brentford, Middlesex, TW8 9GS, United Kingdom.

Corresponding author: Robert A Smith  
(rsmith@darkpeakanalytics.com).

---

## Abstract

**Background:** Health economic models are increasingly used to inform decisions about the allocation of healthcare resources. Ensuring the robustness and reliability of these models is critical. Currently, quality assurance is conducted by both technical and non-technical experts assessing different components of the model manually. This is resource intensive. Understanding how the different components of the model fit together is time consuming, and testing every part of the model is sometimes not feasible in the time available. To aid in this, we propose the `assertHE` R package.

**Methods:** The open source `assertHE` package provides testing functionality for those building and reviewing health economic models. It provides a series of checks which can be integrated into the model development workflow to reduce the probability of common errors. It also provides a suite of functions which allow users to better understand the network of R functions contained in a model, where they are defined, if (and where) they are tested, and the test coverage of those that have.

**Results:** We applied the `assertHE` package to three open source health economic models built in R, showing how to include check functions within the model code and how to visualise the network of functions, see the test coverage, and obtain a Generative Pretrained Transformer Large Language Model (LLM) generated summary of any function in the code-base. We have worked with collaborators from industry, regulators and academia to develop the package to be applicable to the widest possible range of models, making adaptations to the source code based upon feedback.

**Conclusions:** `assertHE` offers a toolkit for health economists building and reviewing models, facilitating a more robust and efficient quality assurance process.

---

## Keywords

HTA, Health Economics, Unit Testing, Model Validation, Model Verification

## Introduction

Cost-effectiveness models play a crucial role in health economic evaluations, aiding the allocation of limited health-care resources. These models aim to simulate the outcomes of alternative courses of action relating to a specific decision problem.

The computation of health economic models has traditionally been conducted in spreadsheet software like MS Excel. While MS Excel is widely accessible and familiar to many users, it has proven to be error-prone in many applications, including health economics [1, 2, 3]. Checking the robustness of these models often involves a laborious manual review process, in which cells of the spreadsheet have to be checked individually for errors. This manual process is not only time-consuming but also subject to human error, making it an inefficient method for ensuring the quality of health economic models.

There is a growing shift towards script-based programming languages. In particular, the R programming language is increasingly used for developing decision analytical models [4]. This transition is occurring because of the improved capabilities, reproducibility, ease of collaboration, and reduced model run-time of R compared to spreadsheet software [5]. Models developed using best-practice methods in R generally consist of a network of functions, each of which are defined in a single location, either within the model code-base or from external software packages [6, 7].

This move toward the development of models as software allows for automated testing and quality assurance techniques to reduce the probability of human error, which are standard practice in mission critical applications in the financial sector, aerospace, and other industries, but are not yet widely adopted in health economic evaluation [6, 7]. The ISPOR-SMDM Modeling Good Research Practices Task Force-7 recommendation V11-4 states that “*VII-4 Models should be subjected to rigorous verification*” and that “*Coding accuracy should be checked using state-of-the-art quality assurance and control methods for software engineering*”, including “*verifying separate parts of a model one by one*” [8] (p. 846). Alarid-Escudero et al. [7] provide a summary of the benefits of testing for health economic models built in R, proposing that these are built into the model development process, with modellers “*writing tests alongside the development of any new function or process or whenever a bug is found. This practice results in a high level of test coverage of the analysis code, reducing the likelihood that unintended interactions or incompatibilities between functions and/or processes will go undetected*” (p. 1335).

Previous work has highlighted the need for verification of models. The AdViSHE tool provides a validation assessment tool to allow modellers to report the validation undertaken [9]. This tool can be used by modellers and reviewers to assess the extent to which a model has been validated. The tool has 13 parts. This paper relates to the 4 contained in part C - Validation of the computerised model (sometimes called verification, internal validity, internal

consistency, technical validity, and debugging). C1 advocates for External review, which the package aims to facilitate, C2 for Extreme Value testing, C3 for Testing of traces and C4 for Unit testing, all of which are the focus of the R package described in this paper. The PACBOARD tool provides software and a user-interface which enables users to explore and validate the inputs and outputs of the probabilistic analysis outputs of health economic model [10]. Additionally, CADTH’s model validation framework serves as a comprehensive resource for ensuring the robustness and credibility of economic models in health technology assessments, offering detailed guidelines and methodologies that can be applied across different stages of model development [11].

This paper introduces the `assertR` R package, which aims to make it easier to build and review robust health economic models in R. The benefits of the package differ by user. From the perspective of the modeller, the package provides a suite of functions which can be used to reduce the probability of human error occurring in iterative model development processes [12, 13]; and can help to map out the relationship between different parts of the model, to plan new additions to the model code and identify redundant code. From the perspective of the reviewer, the package provides a visual summary of the network of functions that make up the model - identifying where each function is defined and tested, and the test coverage of those tests. It also allows the user to obtain a short summary of any function in the network. This summary is generated using a GPT LLM, often referred to as Artificial Intelligence (AI) prompted with the function arguments, body, documentation and other information about the function generated programmatically.

The overall effect of this is to make it quicker and easier to develop and/or review a health economic model developed in R. The intended result is to shift the burden of proof of the quality of the model onto the modeller. In this context, the assumption is ‘guilty unless proven innocent’. The software provided makes it easier for modellers to test their own code, and providing a visual representation of the function network with accompanying testing coverage makes this immediately clear for all parties and can be used to guide discussions. This may be especially useful in instances in which the full source code cannot be made available, since 3rd parties can still get some indication of how the model has been constructed, that tests have been incorporated, and what the model dependencies are.

This is not the first paper to provide more specific verification tools based on AdViSHE. Dasbach and Elbasha [14] provide the justification for several common tests including tests on Life Expectancy, Quality-Adjusted Life Expectancy, Intervention Costs and Cohort Size. The authors include an example set of tests for MS Excel. We adapt and extend these tests, coding them into an R package which can be downloaded (currently from GitHub) using a single line of R code. We provide instructions for how potential collaborators can request to add their own tests

to the package, suggest improvements to existing tests, or help us to build software that performs tests already identified by others as useful. Also of note is the `darthtools` R package [15], which already includes some testing functionality, for example functions to check transition probabilities and to check whether progression free survival exceeds overall survival in extrapolated survival curves. This package aims to build upon these checks, focusing exclusively for model review. To the authors knowledge there is no other existing software which produces a visual representation of a model structure, or provides AI generated summaries of R functions contained in a project.

The remainder of this paper broadly describes the functionality of the `assertHE` package (methods), showcases two case studies previously described in Krijkamp et al. [16] and Alarid-Escudero et al. [17] and applied in previous tutorial papers in a script based format [18, 19] (results) and describes the implications for development and review of models in R more broadly (description). Finally, in the discussion and conclusion sections we outline the implications of this software, its strengths and limitations and potential future research which could build upon this work.

## Methods

### Implementation

The `assertHE` package is designed to be used in two main ways. Firstly, it can be used by a modeller to aid the creation of unit tests, the isolation and examination of individual components (called units) within the software, to ensure the code is performing as intended independently of the rest of the software. These tests are designed to be repeatable, quick to execute, and independent of each other, allowing for early detection of issues in the development process. Since we have pre-created many of the common tests, experienced modellers may wish to write the tests before the model code, fostering a proactive approach to bug detection and promoting software quality through early identification and resolution of issues early on - this is referred to by software engineers as Test-Driven Development (TDD) [20]. Secondly, `assertHE` is designed to be used by reviewers who can visualise the network of functions used in the model, and the coverage of these tests. The reviewer can then use this visualisation and equivalent tabulated output as a checklist when reviewing the model as a starting point before adding their own tests as required. The aim in both instances is to make models more transparent and to reduce the burden of verification, thereby reducing duplication of effort by modeller and reviewer [21].

Detailed documentation for each function can be found in the software package repository. This provides information on each function, what it does and its inputs and outputs. However, in brief the checking functions are as follows:

- `check_trans_prob_array()`

Checks transition probability array for common errors ensuring the same number of rows and columns, only numerical values between 0 and 1, with rows summing to 1. The function provides confirmation or warnings/error messages as appropriate.

- `check_markov_trace()`

Validates Markov trace for feasibility, confirming numerical values, equal row sums, and optionally monotonic increase in dead state proportions, generating confirmation or warnings/error messages.

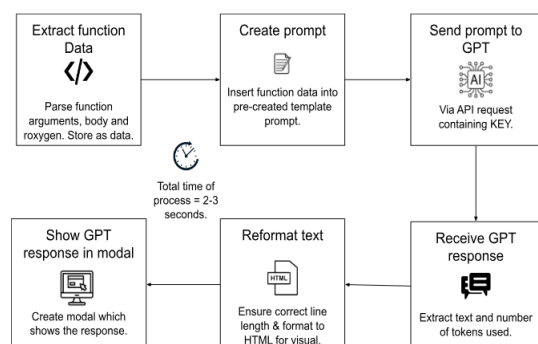
- `check_init()`

Checks initial health state proportions vector for validity, ensuring values within valid probability range, no missing values, sum equals 1, and distinct, non-duplicated names, generating messages for inconsistencies.

- `plot_PSA_stability()`

Generates informative plots to inspect stability of results across model iterations, calculating metrics like INMB, ICER, incremental costs, or effects, allowing customization for specific needs, enhancing visual clarity.

The model overview functions provide a visual and tabulated summary of any R model that has a set of functions in one folder (commonly R/). If the model has a set of tests in another folder (commonly tests/testthat), this can be passed to the overview functions to include data on the location, and coverage of tests for each function in the summary table and network diagram. In the network diagram, created using the `visualise_project` function, tests with coverage below 20% (or no test) are shown in red, tests with coverage from 20-80% are shown in amber, and tests with greater than 80% coverage are shown in green. From the interface that is generated by the function, it is possible to see or navigate to the source code for any function, the source code for any corresponding test, or to generate a short layperson text summary of what the function does. This summary is generated by creating a custom 'prompt' which contains the function arguments, code and documentation along with information on testing coverage on dependencies, and sending this to a GPT LLM via an Application Programming Interface (API). To do this, internet connection and an API Key to the GTP LLM is required. With the existing model, each request uses a small amount of tokens, equating to under \$0.01. A summary of this process can be seen in Figure 1.



**Figure 1.** Diagram showing the process of summarising a function for a user from the network diagram using a GPT-LLM accessed via an API request.

The `assertHE` package has been developed using the same framework of unit testing that it proposes. Each function has associated unit tests that are run using GitHub Actions every time a change is made to the codebase [22].

## Operation

To use `assertHE`, users need R version 3.5.0 or higher. The package imports several dependencies, including `assertthat`, `ggplot2`, `scales`, `tidyr`, `stringr`, `dplyr`, `utils`, `data.table`, `visNetwork`, `covr`, `htmltools`, `officer`, `flextable`, `knitr`, `here`, `shiny`, `shinyjs`, `rstudioapi`, `roxygen2`, `methods`, `waiter`, `igraph`, and `httr`.

## Use Cases

The `assertHE` package has been used in a few ways. Firstly, it was trialled on the sick sicker model described in previous publications [6, 19, 18]). This is used as a vignette to showcase the functionality of the package, and is the focus of the majority of this section. Secondly, it was used to visually inspect a CDX2 biomarker expression model (`cdx2cea` model) and the NICE RCC pathways pilot model (RCC model), two of the few open-source health economics models [23, 24]. Finally, it has been used on closed source models by HTAs, consultancies and industry. The model developers provided feedback on the tool, resulting in iterative improvements in its functionality. We acknowledge those contributions at the bottom of the package README.

### The sick sicker model

In this section, we showcase the functionality of the `assertHE` package on the sick-sicker model described in previous publications [6, 19, 18, 16, 17]. This includes inserting the checks into the model functions while developing the model code, allowing for a quick identification of errors or irregularities. An example of this implementation can be seen on Github.

Within the model code, the health economic modeller or reviewer can insert a check into the model code in a single line. For example the `check_trans_prob_mat()` function checks that a transition probability matrix (e.g. `m_P` in the example below) conforms to a set of rules including that values are between 0 and 1, the matrix is square (same number of rows and columns) and that the names (if any) of rows and columns match. This gives some assurance that a transition matrix object is credible, without requiring a large quantity of code to be included directly into the model, which would make it more difficult to adapt and review. This function can be used multiple times throughout the model to review different transition matrices, and an alternative function `check_trans_prob_array()` is available for three dimensional transition probability arrays commonly used where transition probabilities are time dependent.

The code below creates the visual representation of the network of model functions, with each function represented as a circular node. The size of each node relates to the out-degree centrality of the function, the number of functions called by that function, created using the `igraph` R package [25]. Arrows point from the function being called to the function calling (from child to parent). Colors are used to display testing coverage with custom colors and thresholds passed to the function. Information about each function can be seen by hovering over the function. This information includes the location of the function definition, the location of any tests of the function, and the test coverage.

The code below generates the visual representation. It requires the user to provide the path of the project, the location of the functions (normally "R") and the locations of tests. Calculating test coverage is optional, since some projects may be set up in such a way that it is not possible to test coverage, or the tests break (which itself is useful information for reviewers). If the coverage is run and the tests break then the visual will still appear, albeit with a warning message.

```

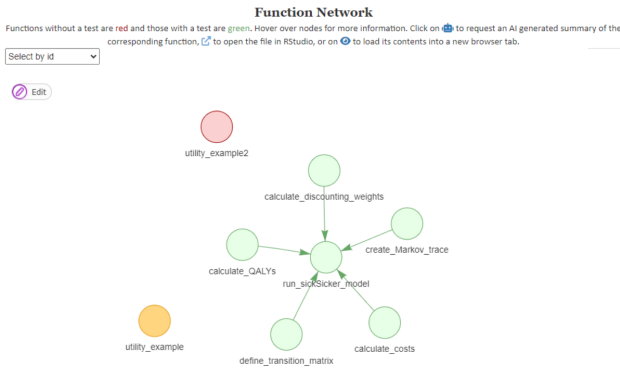
visualise_project(
  project_path = "path_to_project_directory",
  foo_path = "R",
  test_path = "tests/testthat",
  run_coverage = T)

```

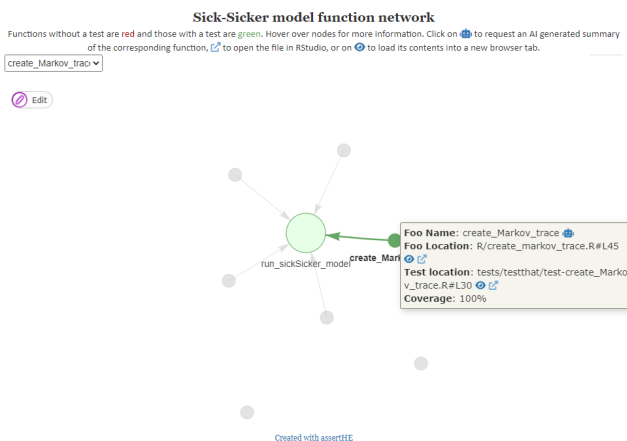
The package also includes an RStudio add-in [26], allowing users of RStudio to click on a button on the RStudio IDE and then use an interface to select the arguments to `visualise_project()` rather than writing code executed in the console. This aims to make `assertHE` easier to use for those who may have access to the model in RStudio but may have limited programming skills.

The resulting visual can be seen in Figure 2. The nodes in green have testing coverage > 80%, in orange testing coverage between 20-80% and in red either no test or coverage < 20%. In Figure 3, hovering over a single node reveals the location of the function definition, the test lo-

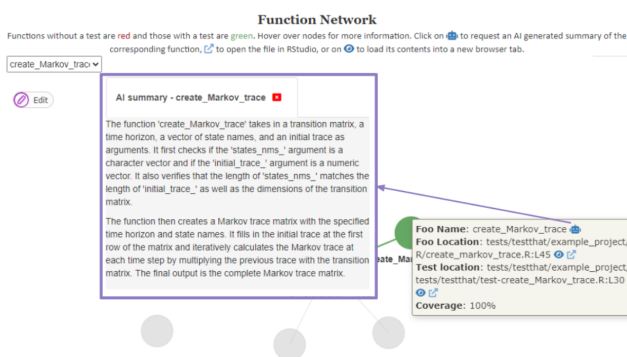
cation and the coverage metric. Users can either click on the eye symbol to see the file in the app, or the navigate symbol to navigate to the file and line number in RStudio. The robot symbol returns the GPT-LLM generated lay-summary of the function as seen in Figure 4.



**Figure 2.** Function network for the sick sicker model developed in R, with the colours indicating testing coverage.



**Figure 3.** Function network for the sick sicker model developed in R, with definitions and locations.



**Figure 4.** GPT-LLM generated lay-summary of a function.

## The cdx2cea model

The cdx2cea model is a cost-effectiveness analysis (CEA) of testing average-risk Stage II colon cancer patients for the absence of CDX2 biomarker expression followed by adjuvant chemotherapy. It is described in detail in Alarid-Escudero et al. [23]. The data and code were made available in R package format with an MIT Licence on GitHub with a version released via Zenodo. It is an example of transparency and good practice in cost-effectiveness modelling. Functions are stored in a folder (“R”) and tests in another folder (“tests”) as is standard in R package.

The cdx2cea model was used to help guide development of assertHE. The function network for local functions (i.e. not those functions called from external packages) can be seen in Figure 5 (at the end of the paper). In this model, only a few key functions, `decision_model`, `calculate_ce_out` and `load_all_params` are tested in the tests folder. Note the interconnectedness of a large number of the functions (top centre), made more transparent to a prospective reviewer with a visual representation, relative to attempting to follow nested functions through multiple R scripts.

## NICE Renal cell carcinoma Pathways Pilot

The NICE Renal cell carcinoma (RCC) Pathways Pilot project developed as part of NICE’s proportionate approach to technology appraisals. The aim and process of the model build is described in Lee et al. [24], and the code is made publicly available at (the somewhat pessimistically named) NICE Model Repo. The codebase is extensive, with approximately 25,000 lines of code. Functions are stored in `3_Functions` and tests in `tests/testthat`. The function network visual, shown in Figure 6 is much larger than the sickicker and cdx2cea models.

## Requirements of a model to be suitable for assertHE

The current functionality of assertHE is primarily tailored for Markov Models utilising transition probability matrices or arrays for constructing Markov traces. To broaden the software’s applicability, we are eager to engage the wider community in suggesting and developing checks for other types of health economic simulation models. These may include partitioned survival models, which partition survival extrapolations into discrete time units to estimate population proportions in various health states over time, and microsimulation models, where individual agents are simulated and results aggregated to estimate impact at the entire population level. We welcome community involvement to expand assertHE’s capabilities and enhance its utility in health economic evaluations.

To generate a visual representation of the model using assertHE, it’s essential to construct the model using functions, which should be organised within a single di-

rectory (e.g., an R folder). To generate information about the existence of tests, tests must be included in a separate folder (e.g., tests). Additional information on code coverage can only be provided if the code complies with the covr package's code coverage requirements, most importantly that tests pass [27]. While these requirements align with standard coding practices in the R programming language [28], it's important to note that the use of R in health economic evaluation is still in its early stages, leading to variations in coding practices among models. By adhering to these practices, `assertHE` not only facilitates the creation of visual representations but also encourages the adoption of good coding practices within the health economic research community, ultimately enhancing collaboration and reproducibility.

## Discussion

Health Economic models are typically conceptualised and described in words and figures and converted into a mathematical model, often with only important or novel subset of the equations provided. However, it is the computational model that generates the results, and the full computational model is often only provided to a small number of reviewers, who have limited time and resources to review. It is crucial for decision-makers, who often do not have the technical skills to verify the model itself, to have confidence in the computational model [29].

The hope is that `assertHE` will make model development and review more efficient and reduce the number of errors in models written in the R programming language, thereby improving the confidence of decision-makers in models. There may also be indirect benefits if it encourages the wider use of formal testing frameworks in health economic models, or mitigates against the reluctance of modellers to make their code open-source [30]. It may also contribute to push the industry toward the use of R for health economic models, resulting in other significant benefits including reduced model run-time, easier collaboration and more efficient model updates [31]. However, it is not a silver bullet. Improving verification of the model code does not validate the model's premise, parameters and structural assumptions. This must be done separately [32]

## Strengths

- This is the first paper we are aware of providing open-source software for verification of health economic models. The R package documentation provides detailed documentation for each function, with examples provided.
- Provided an example, in the form of a case study on the Sick Sicker model to showcase the functionality of the package functions in combination.
- Demonstrated the integration of `assertHE` into continuous integration frameworks, showcasing an au-

tomated testing approach with each model revision.

- Aligns with best practices recommended by ISPOR-SMDM Modeling Good Research Practices Task Force-7 V11-4 for rigorous model verification.
- Open-source collaboration should improve quality, generate consensus and foster transparency.
- Providing standardised testing functions should make the most common checks more efficient, freeing up time for modellers and reviewers to critique models in more innovative ways.
- The authors come from a variety of organisations, providing a diversity of opinion and objectives amongst modelling applications.

## Limitations

- The package `assertHE` is in development phase, and has not yet been submitted to CRAN.
- This paper primarily focuses on R, limiting applicability for health economic models developed in other programming languages. However, functionality does exist to run R code from Python via the software `r2py` [33].
- Most of the functions are developed for State Transition Models, although they can be fairly simply edited for microsimulation models.
- The long-term maintenance and evolution of the `assertHE` package may be subject to challenges, especially if community contributions are limited.
- There is a danger that modellers and reviewers are over-reliant on the checks provided. It should be frequently noted that the functions provided are not exhaustive and further review should be undertaken.
- Outputs returned from pretrained transformer models like OpenAI's GPT models should be treated with caution. In particular there is a risk that models will be biased by the models on which it is trained, and may not be sufficiently intelligent to recognise particularly novel code. It is also likely to be biased by any comments in the code.
- The `assertHE` visualisation functions assume that the model is structured as a network of functions. Models that are scripted without the use of functions (e.g. scripts that source one another and store data for intermediate steps) will not be well described by this function.

## Further research

This paper is only intended to be the beginning of the development of `assertHE`. We invite open collaboration to continuously improve and expand the open-source R package before we submit to the Comprehensive R

Archive Network (CRAN) [34]. It's essential to assess whether and how much the package improves and speeds up the review of health economic decision models in practice. This may involve an examination of its applicability across diverse health economic model types such as microsimulation, decision-tree, and agent-based models, to evaluate its generalizability. It could also incorporate information about external packages used and cross-reference to HTA agency lists of approved packages [35]. HTA agencies and funders could play an important role in improving modelling practices by recommending testing and other good software development practices in submissions.

To enhance the user experience and promote broader adoption, further research should delve into the impact of user proficiency in R programming on `assertHE`'s effectiveness. Developing and evaluating training resources to complement existing publications [36, 6, 19, 18, 37, 38, 39] that cater to users with varying levels of programming experience will be crucial in facilitating a wider adoption of the tool. It is also worth exploring the feasibility of adapting `assertHE` for health economic models developed in languages other than R, particularly Python. This investigation should include an assessment of the challenges and benefits associated with extending `assertHE`'s functionality to accommodate other programming languages that may become commonly used in the health economics domain.

Securing `assertHE`'s long-term future hinges on actively exploring ways to maintain and improve it. If health economists working in R can be encouraged to contribute their tests, the tool will become increasingly useful and therefore will in-turn engage people. Engaging health economists, training them in its use, and in how to contribute, will be essential. Building confidence with HTA agencies, with a view for long-term use as standard practice once a critical mass of use and testing has been achieved, could increase the impact of the software.

The sustainability of `assertHE` requires a commitment to software maintenance and adaptation in the long-term. To achieve this it will be important to incentivise collaboration from the health economics community. Where opportunities exist to integrate `assertHE` with popular R packages commonly employed by health economists, such as `heemod` [40] and `hesim` [5], these could be encouraged since integration could enhance the accessibility and usability of `assertHE` for a broader audience, extending its impact beyond experienced R users.

Future research could also include fine-tuning a large language model on health economics-specific content, such as Health Economics textbooks and open-source R models, to potentially enhance the performance of the model currently used in the prototype described in this paper. This improvement could be evaluated by an independent group of health economists, who would assess both the quality of the current summaries and the impact of fine-tuning. Additionally, integrating a chat-bot into the code would enable users to interact with the model, facilitat-

ing discussions about its outputs. Further research should also explore the broader implications of verification software like `assertHE` on the willingness of modellers to share their code as open-source. Finally, it is crucial to assess the possibilities and risks associated with a future where models and reviews are largely conducted by generative AI.

The checks provided are not intended to be exhaustive. The intention is that others will contribute checks as they use them. This can be done by submitting a pull request to the package on GitHub (with all automated checks shown to pass), or creating an issue describing the checks that would be required, ideally with an example of the code needed to implement them. We would be particularly interested in checks that can be performed outside of model functions, to assess the external validity of models, as described in a recent model validation checklist created by CADTH [41]. More information on how to contribute can be found at the package's Github repo.

## Summary

This paper has introduced `assertHE`, an R package designed to enhance the efficiency of the verification process for health economic models and reduce the likelihood of errors. The case study, using the Sick-Sicker model, illustrates the practical application of `assertHE` on a health economic model, using a continuous integration framework. This automated testing approach with each model revision aims to reduce the risk of introducing bugs during model edits. Meanwhile the visualisation and summary functionality aims to make it easier for external reviewers to understand how the individual algorithms in the model fit together. Used alongside other tools, such as quality checklists and tests of external validity, `assertHE` may help increase confidence in the veracity of health economic models, allowing decision-makers to make better-informed choices regarding the allocation of limited healthcare resources, contributing to improved population health outcomes.

## Data availability

No additional source data are required.

## Software availability

This section will be generated by the Editorial Office before publication. Authors are asked to provide some initial information to assist the Editorial Office, as detailed below.

1. <https://github.com/dark-peak-analytics/assertHE> (AUTHOR TO PROVIDE; optional)
2. <https://github.com/dark-peak-analytics/assertHE> (AUTHOR TO PROVIDE; required)



3. Link to source code as at time of publication (*F1000Research* TO GENERATE)
4. Link to archived source code as at time of publication (*F1000Research* TO GENERATE)
5. Software license (MIT)

## Author contributions

In order to give appropriate credit to each author of an article, the individual contributions of each author to the manuscript should be detailed in this section. We recommend using author initials and then stating briefly how they contributed.

## Competing interests

R.S. is part of the Scientific Committee for R for HTA, an academic consortium whose main objective is to explore the use of R for cost-effectiveness analysis. R.S., P.S. and W.M. work for Dark Peak Analytics Ltd; a company specialising in the application of techniques similar to those discussed in this paper in epidemiology and health economic evaluation. Y.S., A.M, S.J.M, B.D. were employees of GSK at the time of writing.

## Grant information

R.S., P.S. and W.M. are joint funded by the Wellcome Trust Doctoral Training Centre in Public Health Economics and Decision Science [108903] and the University of Sheffield. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

## Acknowledgments

We would like to thank all those at R-HTA 2024 for their useful comments and feedback, as well as those who tested the models internally in particular Niall Davison.

## References

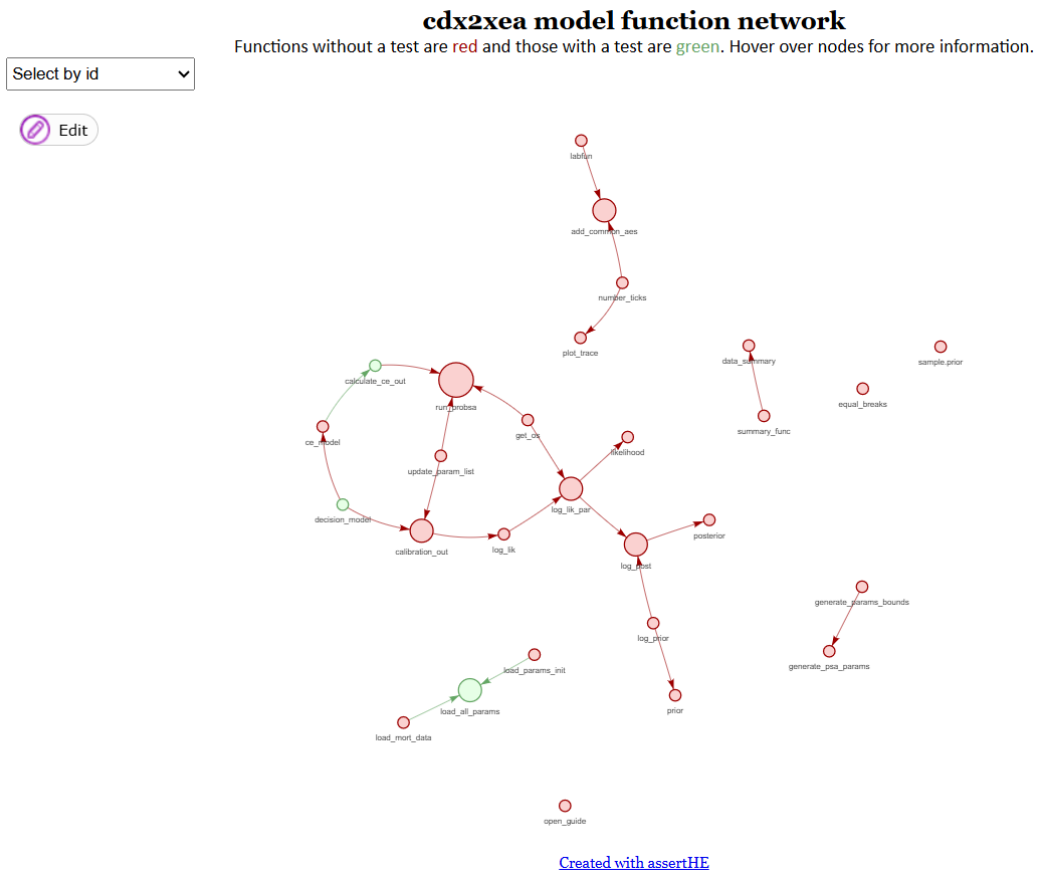
- [1] Demi Radeva, Gareth Hopkin, Elias Mossialos, John Borriell, Leeza Osipenko, and Huseyin Naci. Assessment of technical errors and validation processes in economic models submitted by the company for nice technology appraisals. *International Journal of Technology Assessment in Health Care*, 36(4):311–316, 2020. doi: 10.1017/S0266462320000422.
- [2] National Institute for Health and Care Excellence (NICE). Ribociclib with an aromatase inhibitor for previously untreated, hormone receptor-positive, her2-negative, locally advanced or metastatic breast cancer, 2018. URL <https://www.nice.org.uk/guidance/ta555>. Accessed: 2018-08-06.
- [3] Raymond R. Panko. What we don't know about spreadsheet errors today: The facts, why we don't believe them, and what we need to do. *CoRR*, abs/1602.02601, 2016. URL <http://arxiv.org/abs/1602.02601>.
- [4] Hawre Jalal, Petros Pechlivanoglou, Eline Krijkamp, Fernando Alarid-Escudero, Eva Enns, and M. G. Myriam Hunink. An overview of r in health decision sciences. *Medical Decision Making*, 37(7):735–746, 2017. doi: 10.1177/0272989X16686559. URL <https://doi.org/10.1177/0272989X16686559>.
- [5] Devin Incerti and Jeroen P Jansen. hesim: Health economic simulation modeling and decision analysis. *arXiv preprint arXiv:2102.09437*, 2021.
- [6] Robert Smith, Wael Mohammed, and Paul Schneider. Packaging cost-effectiveness models in r: A tutorial. *Wellcome Open Research*, 8, 2023.
- [7] Fernando Alarid-Escudero, Eline M Krijkamp, Petros Pechlivanoglou, Hawre Jalal, Szu-Yu Zoe Kao, Alan Yang, and Eva A Enns. A need for change! a coding framework for improving transparency in decision modeling. *Pharmacoeconomics*, 37(11):1329–1339, 2019.
- [8] David M. Eddy, William Hollingworth, J. Jaime Caro, Joel Tsevat, Kathryn M. McDonald, and John B. Wong. Model transparency and validation: A report of the ispor-smdm modeling good research practices task force–7. *Medical Decision Making*, 32(5):846, 2012. doi: 10.1177/0272989X12454579. URL <https://doi.org/10.1177/0272989X12454579>. PMID: 22990088.
- [9] P. Vemer, I. Corro Ramos, G. A. K. Voorn, M. J. Al, and T. L. Feenstra. AdViSHE: A Validation-Assessment Tool of Health-Economic Models for Decision Makers and Model Users. *Pharmacoeconomics*, 34(4):349–361, April 2016. doi: 10.1007/s40273-015-0327-2. URL [https://ideas.repec.org/a/spr/pharme/v34y2016i4d10.1007\\_s40273-015-0327-2.html](https://ideas.repec.org/a/spr/pharme/v34y2016i4d10.1007_s40273-015-0327-2.html).
- [10] Xavier GLV Pouwels, Karel Kroeze, Naomi van der Linden, Michelle MA Kip, and Hendrik Koffijberg. Validating health economic models with the probabilistic analysis check dashboard. *Value in Health*, 2024.
- [11] Doug Coyle, Alex Haines, and Karen Lee. Virtual care: Health technology review. *Canadian Journal of Health Technologies*, 1(1):1–28, 2021. URL <https://canjhealthtechnol.ca/index.php/cjht/article/view/MG0026>.
- [12] Hadley Wickham. testthat: Get started with testing. *R J.*, 3(1):5, 2011.
- [13] Nasuh Büyükkaramikli, Maureen Mólken, Hans Severens, and Maiwenn Al. Tech-ver: A verification checklist to reduce errors in models and improve their credibility. *Pharmacoeconomics*, 37, 11 2019. doi: 10.1007/s40273-019-00844-y.
- [14] Erik J Dasbach and Elamin H Elbasha. Verification of decision-analytic models for health economic evaluations: an overview. *Pharmacoeconomics*, 35(7):673–683, 2017.
- [15] Fernando Alarid-Escudero, Gianluca Baio, Eva Enns, Hawre Jalal, and Petros Pechlivanoglou. *darthtools*:



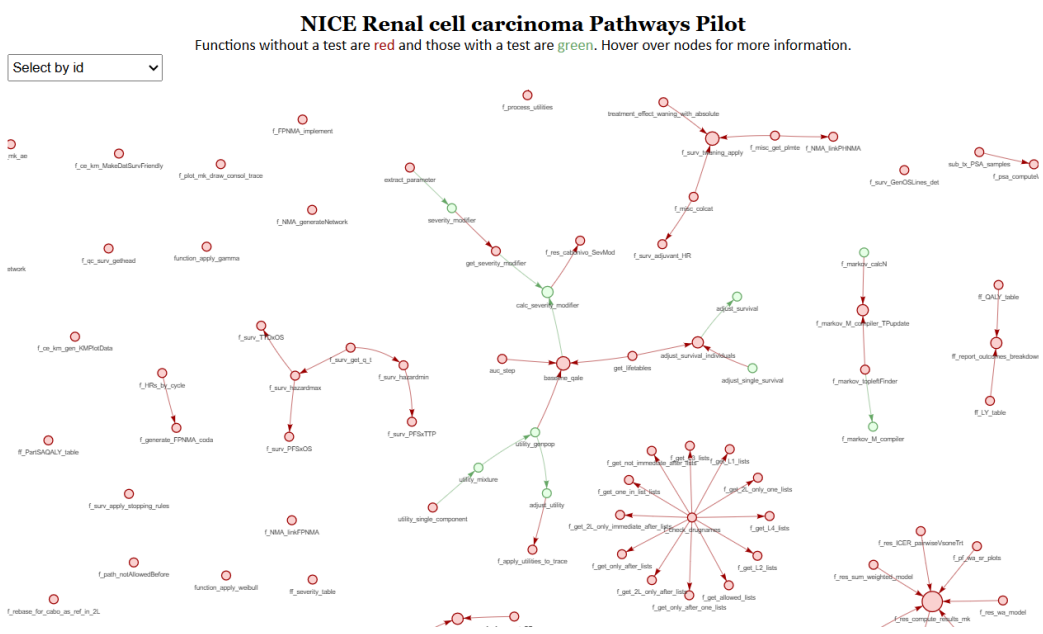
- Decision-Analytic Routines for Transparent Healthcare Models*, 2019. URL <https://github.com/DARTH-git/darthtools>. R package version 1.1.0.
- [16] Eline M Krijkamp, Fernando Alarid-Escudero, Eva A Enns, Hawre J Jalal, MG Myriam Hunink, and Petros Pechlivanoglou. Microsimulation modeling for health decision sciences using r: a tutorial. *Medical Decision Making*, 38(3):400–422, 2018.
- [17] Fernando Alarid-Escudero, Eline Krijkamp, Eva A. Enns, Alan Yang, M. G. Myriam Hunink, Petros Pechlivanoglou, and Hawre Jalal. A tutorial on time-dependent cohort state-transition models in r using a cost-effectiveness analysis example. *Medical Decision Making*, 43(1):21–41, 2023. doi: 10.1177/0272989X221121747. URL <https://doi.org/10.1177/0272989X221121747>. PMID: 36112849.
- [18] Robert Smith and Paul Schneider. Making health economic models shiny: A tutorial. *Wellcome open research*, 5, 2020.
- [19] Robert A Smith, Paul P Schneider, and Wael Mohammed. Living hta: Automating health economic evaluation with r [version 2; peer review: 2 approved]. *Wellcome open research*, 2022.
- [20] K. Beck. *Test-driven Development: By Example*. Addison-Wesley signature series. Addison-Wesley, 2003. ISBN 9780321146533. URL <https://books.google.co.uk/books?id=CULsAAQAQBAJ>.
- [21] Chris J. Sampson and Tom Wrightson. Model registration: A call to action. *PharmacoEconomics - Open*, 1(2):73–77, 2017. doi: 10.1007/s41669-017-0019-2. URL <https://doi.org/10.1007/s41669-017-0019-2>.
- [22] Hadley Wickham. *assertthat: Easy Pre and Post Assertions*, 2019. URL <https://CRAN.R-project.org/package=assertthat>. R package version 0.2.1.
- [23] Fernando Alarid-Escudero, Deborah Schrag, and Karen M Kuntz. Cdx2 biomarker testing and adjuvant therapy for stage ii colon cancer: an exploratory cost-effectiveness analysis. *Value in Health*, 25(3):409–418, 2022.
- [24] Dawn Lee, Darren Burns, and Ed Wilson. Nice’s pathways pilot: Pursuing good decision making in difficult circumstances. *PharmacoEconomics-open*, pages 1–5, 2024.
- [25] Gábor Csárdi, Tamás Nepusz, Vincent Traag, Szabolcs Horvát, Fabio Zanini, Daniel Noom, and Kirill Müller. *igraph: Network Analysis and Visualization in R*, 2024. URL <https://CRAN.R-project.org/package=igraph>. R package version 2.0.3.
- [26] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, PBC., Boston, MA, 2020. URL <http://www.rstudio.com/>.
- [27] Jim Hester. *covr: Test Coverage for Packages*, 2023. URL <https://CRAN.R-project.org/package=covr>. R package version 3.6.4.
- [28] H. Wickham and J. Bryan. *R Packages*. O’Reilly Media, 2023. ISBN 9781098134907. URL <https://books.google.co.uk/books?id=kTHFEAAAQBAJ>.
- [29] Paul Tappenden and James Chilcott. Avoiding and Identifying Errors and Other Threats to the Credibility of Health Economic Models. *PharmacoEconomics*, 32(10):967–979, October 2014. doi: 10.1007/s40273-014-0186-2. URL <https://ideas.repec.org/a/spr/pharme/v32y2014i10p967-979.html>.
- [30] Joanna Emerson, Rachel Bacon, Alma Kent, Peter J. Neumann, and Joshua T. Cohen. Publication of Decision Model Source Code: Attitudes of Health Economics Authors. *PharmacoEconomics*, 37(11):1409–1410, November 2019. doi: 10.1007/s40273-019-00796-. URL [https://ideas.repec.org/a/spr/pharme/v37y2019i11d10.1007\\_s40273-019-00796-3.html](https://ideas.repec.org/a/spr/pharme/v37y2019i11d10.1007_s40273-019-00796-3.html).
- [31] Devin Incerti, Howard Thom, Gianluca Baio, and Jeroen P Jansen. R you still using excel? the advantages of modern software tools for health technology assessment. *Value in Health*, 22(5):575–579, 2019. ISSN 1098-3015. doi: <https://doi.org/10.1016/j.jval.2019.01.003>. URL <https://www.sciencedirect.com/science/article/pii/S1098301519300506>.
- [32] Isaac Ramos, Talitha Feenstra, Salah Ghabri, and Maiwenn Al. Evaluating the validation process: Embracing complexity and transparency in health economic modelling. *PharmacoEconomics*, 42, 03 2024. doi: 10.1007/s40273-024-01364-0.
- [33] Laurent Gautier. Rpy2: A Simple and Efficient Access to R from Python, 2008. URL <http://rpy.sourceforge.net/rpy2.html>. Accessed: 2024-08-12.
- [34] Kurt Hornik. The comprehensive r archive network. *WIREs Computational Statistics*, 4(4):394–398, 2012. doi: <https://doi.org/10.1002/wics.1212>. URL <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.1212>.
- [35] Zorginstituut Nederland. Appendix to guideline for cost-effectiveness models in r, 2022. URL <https://english.zorginstituutnederland.nl/binaries/zinl-eng/documenten/publications/2022/12/15/appendix-to-guideline-for-cost-effectiveness-models-Appendix+to+guideline+for+cost-effectiveness+models+in+R+-+packages+and+versions.pdf>. Accessed: 02 August 2024.
- [36] M.K. Seo and M. Strong. A practical guide to modeling and conducting a cost-effectiveness analysis of companion biomarker tests for targeted therapies using r : tutorial paper. *PharmacoEconomics*, 39(12):1373–1381, December 2021. URL <https://eprints.whiterose.ac.uk/177323/>.
- [37] Paolo Candio, Koen B. Pouwels, David Meads, Andrew J. Hill, Laura Bojke, and Claire Williams. Modelling decay in effectiveness for evaluation of behaviour change interventions: a tutorial for public health economists. *The European Journal of Health Economics*, 23(7):1151–1157, September 2022. doi: 10.1007/s10198-021-01417-. URL [https://ideas.repec.org/a/spr/eujhec/v23y2022i7d10.1007\\_s10198-021-01417-7.html](https://ideas.repec.org/a/spr/eujhec/v23y2022i7d10.1007_s10198-021-01417-7.html).
- [38] Yiqiao Xin, Ewan Gray, Jose Antonio Robles-Zurita, Houra Haghpanahan, Robert Heggie, Ciaran Kohli-Lynch, Andrew Briggs, David A. McAllister, Kenny D.

Lawson, and Jim Lewsey. From Spreadsheets to Script: Experiences From Converting a Scottish Cardiovascular Disease Policy Model into R. *Applied Health Economics and Health Policy*, 20(2):149–158, March 2022. doi: 10.1007/s40258-021-00684-. URL [https://ideas.repec.org/a/spr/aphecp/v20y2022i2d10.1007\\_s40258-021-00684-y.html](https://ideas.repec.org/a/spr/aphecp/v20y2022i2d10.1007_s40258-021-00684-y.html).

- [39] Yi-Shu Lin, James F O'Mahony, and Joost Rosmalen. A Simple Cost-Effectiveness Model of Screening: An Open-Source Teaching and Research Tool Coded in R. *Pharmacoeconomics - Open*, 7(4):507–523, July 2023. doi: 10.1007/s41669-023-00414-. URL [https://ideas.repec.org/a/spr/pharmo/v7y2023i4d10.1007\\_s41669-023-00414-1.html](https://ideas.repec.org/a/spr/pharmo/v7y2023i4d10.1007_s41669-023-00414-1.html).
- [40] A Filipovic-Pierucci, K Zarca, and I Durand-Zaleski. Prm68 - markov models for health economic evaluation modelling in r with the heemod package. *Value in Health*, 19(7):A369, 2016. ISSN 1098-3015. doi: <https://doi.org/10.1016/j.jval.2016.09.133>. URL <https://www.sciencedirect.com/science/article/pii/S1098301516315005>.
- [41] Doug Coyle, Alex Haines, and Karen Lee. The development of a model validation tool to assist in the conduct of economic evaluations. *Canadian Journal of Health Technologies*, 4(3), 2024.



**Figure 5.** Function network for cdx2cea R model.



**Figure 6. Function network for RCC Pathways R model**