PharmacoEconomics - Open (2023) 7:507–523
https://doi.org/10.1007/s41669-023-00414-1

PRACTICAL APPLICATION

A Simple Cost-Effectiveness Model of Screening: An Open-Source Teaching and Research Tool Coded in R

Yi-Shu Lin[1] · James F O'Mahony[1] · Joost van Rosmalen[2,3]

Accepted: 4 April 2023 / Published online: 1 June 2023

Abstract

Applied cost-effectiveness analysis models are an important tool for assessing health and economic effects of healthcare interventions but are not best suited for illustrating methods. Our objective is to provide a simple, open-source model for the simulation of disease-screening cost-effectiveness for teaching and research purposes. We introduce our model and provide an initial application to examine changes to the efficiency frontier as input parameters vary and to demonstrate face validity. We described a vectorised, discrete-event simulation of screening in R with an Excel interface to define parameters and inspect principal results. An R Shiny app permits dynamic interpretation of simulation outputs. An example with 8161 screening strategies illustrates the cost and effectiveness of varying the disease sojourn time, treatment effectiveness, and test performance characteristics and costs on screening policies. Many of our findings are intuitive and straightforward, such as a reduction in screening costs leading to decreased overall costs and improved cost-effectiveness. Others are less obvious and depend on whether we consider gross outcomes or those net to no screening. For instance, enhanced treatment of symptomatic disease increases gross effectiveness, but reduces the net effectiveness and cost-effectiveness of screening. A lengthening of the preclinical sojourn time has ambiguous effects relative to no screening, as cost-effectiveness improves for some strategies but deteriorates for others. Our simple model offers an accessible platform for methods research and teaching. We hope it will serve as a public good and promote an intuitive understanding of the cost-effectiveness of screening.

Key Points for Decision Makers

We provide a simplified screening cost-effectiveness analysis microsimulation for teaching and research purposes.

As an initial application, we present an assessment of 8161 screening strategies and conduct comparative statics to illustrate the influence of parameters on cost-effectiveness.

Our analysis conveys the intuition of the relationship between parameter values and outcomes, including both absolute costs and effects and those relative to no screening, informing the process of model validation.

✉ Yi-Shu Lin
yilin@tcd.ie

[1] Centre for Health Policy and Management, Trinity College Dublin, 2-4 Foster Place, Dublin D02 T253, Ireland

[2] Department of Biostatistics, Erasmus Medical Centre, Rotterdam, The Netherlands

[3] Department of Epidemiology, Erasmus Medical Centre, Rotterdam, The Netherlands

1 Introduction

Cost-effectiveness analysis (CEA) is the standard method for assessing value for money in healthcare [1, 2]. Models have been applied extensively to examine the cost-effectiveness of cancer-screening policies [3–5]. Such models permit appraisal of a broader range of strategies than is feasible to assess in trials [6]. Similarly, simulations can offer decision makers estimates of the long-term effects of screening earlier than can be achieved within trials [3].

Despite the widespread application of CEA models there are recognised methodological shortcomings in applied modelling studies [7, 8]. These include the issues of the failure to conduct incremental analyses and the omission of relevant strategies [8]. Such issues are evident in the cancer-screening CEA literature in particular [9, 10]. Other screening-specific issues include risk stratification, which does not always appear appropriately applied [11–13].

Several factors may contribute to the persistence of such methodological shortcomings. First, many model-based analyses primarily address applied research questions such as the cost-effectiveness of a particular policy

proposal rather than addressing methods concerns or publishing accessible teaching examples. Second, many CEA models are not openly shared by the academic groups that hold them, which inhibits their application by others to examine methods questions and perpetuates "blackbox" opacity concerns. While a few open-source models are available, these are often applied analyses and come with attendant complexity [14, 15]. As such, these models do not offer useful starting points for novice modellers to begin learning the fundamentals of simulation and can require long run times, especially when simulating many strategies. Overall, the limited availability of simple, fast, openly shared models means the screening literature lacks an accessible simulation platform for teaching and methods research.

An important aspect of model development is model validation. This helps avoid errors and ensures the model is fit for purpose [16]. One aspect of validation is establishing face validity, which relies on subjective expert judgement regarding the research question [17]. This is used to assess if a model's outputs are consistent with expectations. Modellers therefore need to be equipped with an understanding of what results appear plausible to identify and avoid errors [16]. Some of the early screening modelling literature used analytical models that employed relatively high levels of abstraction [18–20]. Such simplified models are useful for generating an intuitive understanding of screening, yet these models can be algebraically challenging to solve and become impractically complex once high degrees of abstraction are relaxed. The availability of programmable computers and simulation software means analytical approaches have generally been superseded by simulation. While simulation is suitable for applied optimisation problems, the loss of abstraction can compromise their usefulness in illustrating relationships between parameters and outcomes.

If researchers lack an intuitive understanding of screening cost-effectiveness and how it varies between strategies and across parameter values, they may be ill-equipped to assess elementary face validity. This in turn may compromise prospects for quality improvement in CEA modelling. Previous work in the specific context of CEAs of colorectal cancer screening found that although most authors report having conducted face validation exercises, few studies actually present evidence of the validation [21]. The applied nature of these CEA models does not lend itself to demonstration of face validation in abstract terms, including the explanation for the relationships between the model inputs and their corresponding cost-effectiveness estimates.

Several tutorials on state-transition modelling have been published [22–25], although none are presented in the context of screening interventions. Notable recent examples are the tutorials published by the Decision Analysis in R for Technologies in Health (DARTH) group [22–24]. Most tutorials published to date address discrete-time state-transition models, and we are only aware of one tutorial on discrete-event simulation (DES) [26]. To date, there is no open-source teaching model designed for CEA screening interventions, irrespective of model type.

We believe the lack of an accessible, readily sharable model represents a meaningful research gap in the screening CEA literature. The objective of this study is to introduce an open-source modelling platform for the simulation of cost-effectiveness of disease screening for teaching and research purposes. This simplified model employs DES and is coded in the R programming language. It is deliberately coded largely in base R in order to enhance accessibility and reduce dependence on installed packages. It also employs Microsoft Excel spreadsheets to aid easy definition of parameter values and convenient inspection of results for those less familiar with R. The model is specifically intended to be capable of simulating a large range of screening strategies in order to illustrate the importance of including sufficient screening alternatives among other methods considerations. DES is chosen as it offers an intuitive and highly efficient modelling paradigm within which to simulate screening interventions.

As an initial application of our model, we demonstrate the relationship between parameters and outcomes in order to support the development of intuitive understanding of screening cost-effectiveness. Our analysis aims to illustrate the effects of disease incidence rates, preclinical durations and test performance characteristics on the costs and effects of screening. In particular, we demonstrate how the position of the cost-effective efficiency frontier varies as these parameters change, and the implication for optimal screening policies. We hope that our model will serve as a training tool for those working with screening models and that this will enhance understanding of CEA simulation, which in turn will lead to better evidence, more effective policies and, ultimately, improved health outcomes.

2 A Pedagogical Model

This simplified microsimulation model is coded in R (version 4.2.1) and comprises approximately 730 lines including markup. The complete model code and its specification are available for all to access freely on GitHub (https://github.com/yishu-lin/Pedagogical-CEA-Model-of-Screening.git).

2.1 Model Overview

We first provide a broad outline of the modelling approach before giving a detailed description of selected key elements of the model. The intention is not to give a complete walk through

of the model code within the article. The interested reader can consult the fully marked-up model code on GitHub for a complete description. Rather, we wish to provide an overview of the modelling approach before describing an application used to demonstrate how the efficiency frontier changes as key input parameters change. This model was designed for illustrative purposes and does not represent any specific disease or intervention, though it is broadly conceived in the context of cancer, in which there is a non-communicable preclinical disease that can be screened for at multiple points over an individual's lifetime. Our model is deliberately more abstract than the applied models typically used in applied CEAs.

We use a single-lesion model, meaning that individuals can only develop one instance of disease per lifetime. The disease can be treated, either upon clinical presentation or screen detection. Treatment is not assumed to be perfectly effective as not all patients survive, but those who survive are assumed to have no long-term morbidity. We also assumed no disease recurrence.

This individual-level DES depicts a natural history of disease for a single lesion with five health states (Fig. 1). All individuals start in the perfect health state but are at risk of disease. In the absence of disease, individuals die of other causes, the timing of which is determined by assumed life tables. Individuals in the preclinical state have disease but have not been diagnosed and suffer no symptoms. Their disease is detectable by screening. Individuals can be diagnosed after either symptomatic presentation or screen detection, at which point they enter a clinical state and start treatment. We assume treatment has a higher probability of success if disease is detected in the preclinical state. If individuals are cured, there is no further treatment and survivors are excluded from future screening activity. In the case of treatment failure, individuals die at the same point in time as if no treatment occurred, i.e. there is no longevity benefit. In the case of treatment success, death occurs at the time of other-cause death as determined by the assumed life tables. The model simulates all health outcomes for each individual until death.

2.2 Model Structure

2.2.1 Simulation of Natural History

First, the model simulates the natural history of each individual. The age at entering the preclinical state and age at other-cause death are independently drawn from the probability distributions of disease incidence and life tables. Both distributions are defined by the incidence probability and the survival rate for the user-defined age groups. The sojourn time of the preclinical and clinical stages are assigned from user-defined distributions. Users can choose from uniform, exponential or Weibull distributions. An individual's age exiting a given health state is determined by their age at entry plus the sampled sojourn time in that state. An individual's all-cause death age is the minimum of the cause-specific and other-cause death ages.

The model employs a vectorised approach, meaning that large vectors are used to record the age of entry and exit of specific states corresponding with each element in the vector corresponding to simulated individuals, and the model operations are, wherever possible, applied over these vectors. The vectorised approach aids the efficiency of the model and minimises the iterative use of loops.

2.2.1.1 Pseudo-code A fundamental outcome table in the model is one which records the unique identification of each simulated individual and the age of entry into the three possible disease states, which are preclinical disease, clinical disease and cause-specific death (Box 1). This array also records the other-cause death age and the overall all-cause death age. This array therefore has dimensions $n \times 6$, where n is the number of simulated

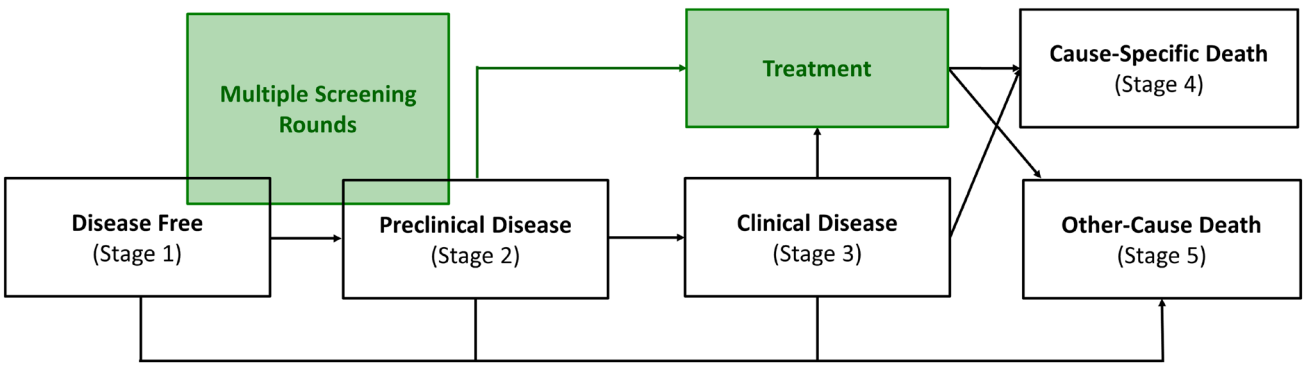

Fig. 1 Model diagram. The white health states represent the natural history of disease. The two interventions, screening and treatment, are labelled in green. The screening intervention can only be implemented before the individual enters clinical health state (stage 3), and treatment can be received when patients are diagnosed from screening (stage 2) or clinically presented (stage 3)

```
# Define number of health stages with the stage arrival
# This model only features five states: (1)Disease Free; (2)Preclinical
Disease; (3)Clinical Disease; (4)Cause-Specific Death; (5)Other-Cause Death
Outcomes <- array(NA, dim = c(SampleSize, 6))  # Create an array of the
length of the sample size
colnames(Outcomes) <- c("PersonNumber",
paste(DefineStages[2:nrow(DefineStages), "Name"]), "AllCauseDeath")  # Set
column names
Outcomes[, "PersonNumber"] <- c(1:SampleSize)  # Set the first column to be
the unique person-number for each individual
```

**DefineStages* is a data frame recording the parameters in the spreadsheet *Stages*, relevant to disease history.

Box 1 Pseudo-code creating an outcome table

```
# The intervening columns correspond to the arrival of the
intermediate disease states
# This model studies the cohort with the same age, so here does not
need to simulate the first health state (healthy)
# Define the generic onset function which applies an age-specific
probability of entering a specific stage
OnsetFunction <- function(x){
  unlist(approx(probability, age, x, ties = max)[2], use.names = F)
# Ties = max is required because of the possibility of multiple zero
probabilities of disease at younger ages
}
```

Box 2 Pseudo-code defining an onset function for disease incidence and other-cause deaths

individuals. There is no explicit recording of membership of the healthy state as all individuals inhabit this state from birth until either the onset of preclinical disease or other-cause death.

To simulate the age of entry into the preclinical disease state, the model uses an onset function that samples the age of entering a specific health state for each individual from an age-specific probability (Box 2). This employs linear interpolation from a piecewise linear function of the cumulative probability of disease onset with age. The same approach is applied to simulate the age of entering the preclinical state and age at other-cause death.

The onset function is then applied over a vector x, generated using a random value between 0 and 1, with a length equal to the simulated population size. Readers seeking detail on how the *probability* and *age* arguments are defined should refer to the complete model code.

Regarding other health states, the age of entering the state is the age of entering the previous state plus the sojourn time of this previous state. This is achieved by applying a loop over the number of stages minus one, to exclude the final stage of other-cause death (Box 3). Inside the loop, we create random numbers to draw from the corresponding sojourn time distributions. Exponential and Weibull distributions can be chosen to sample the sojourn time of each health state, as can a constant duration. Naturally, a constant duration does not require random sampling because it assumes every individual has a fixed duration. Any health states entered after age 100 are then censored as the model assumes all simulated individuals die by age 100.

The age of all-cause death is determined as the minimum of an individual's cause-specific death and other-cause death (Box 4).

2.2.2 Adjustment of Screening Strategies

The model includes a primary screening test, the sensitivity and specificity of which can be adjusted, as can the interval between screens and the start and stop ages. The model assumes a disutility from primary screening to account for the quality-of-life (QoL) loss due to the time and effort associated with undergoing a screening test. The

```
# The model needs a loop here to go through the disease stages
for (Stage in 1:(nrow(DefineStages) - 1)){
  # Apply the sojourn time to the stages
  # Retrieve the distribution type, scale and shape
  DurationType  <- Input[paste("StageType",  Stage, sep = ""),
CurrentRun]
  DurationScale <- Input[paste("StageScale", Stage, sep = ""),
CurrentRun]
  DurationShape <- Input[paste("StageShape", Stage, sep = ""),
CurrentRun]

  if (!(is.na(DurationType))){
    if (DurationType == 1){Duration <- rep(DurationScale,
SampleSize)}  # Set the preclinical distribution to be constant
    if (DurationType == 2){Duration <- -(log(1 - runif(SampleSize)))
* DurationScale}  # Set the preclinical distribution to be
exponentially distributed
    if (DurationType == 3){Duration <- ((-log(1 - runif(SampleSize)))
^ (1 / DurationShape)) * DurationScale}  # Set the preclinical
distribution to be Weibull distribution, and the alternative code:
rweibull(SampleSize, shape = DurationShape, scale = DurationScale)
    Outcomes[, Stage + 1] <- Outcomes[, Stage] + Duration  # Find the
end of the preclinical period by adding the onset to the duration
  }

  # This study assumes all the people died before aged 100
  Outcomes[which(Outcomes[, Stage + 1] > 100), Stage + 1] <- 100
}
```

Box 3 Pseudo-code presenting a loop for sampling the sojourn time of health states

```
# Find the all-cause death
Outcomes[, "AllCauseDeath"] <- pmin(Outcomes[, "CauseSpecificDeath"],
Outcomes[, "OtherCauseDeath"], na.rm = TRUE)
```

Box 4 Pseudo-code adjusting the age of all-cause deaths

model permits the simulation of alternative sets of test performance assumptions, corresponding to alternative primary screen modalities. Which modalities are applied and the screening interval can be varied over the course of an individual's screening programme (Fig. 2). The model does not explicitly simulate a triage test, but does assume all the positives receive triage and only true positives access early treatment. All those false positives from the initial screening test therefore do not receive an intervention and remain eligible for future screening rounds.

The screening schedules are generated from the target age ranges and intervals. In many cases, the schedules are generated as approximations as a given screening age range may not be perfectly divisible by a given screening interval. For example, triennial screening with a start age of 40 and stop age of 60 is approximated by eight screens between the ages of 40 and 61. This approximation is achieved by holding the starting age and interval fixed but choosing the stop age that gives the closest approximation to the target stop age. Where two alternative stop ages can approximate a given target stop age, we specify the higher of the two. For example, if a screening stop age that the user defined is 70, while both 68 and 72 could approximate 70, our model applies 72.

2.2.2.1 Pseudo-code The model can generate screening schedules in which the screening interval length varies, such as is often employed in cervical screening. For the sake of clarity, the application here uses a constant screening inter-

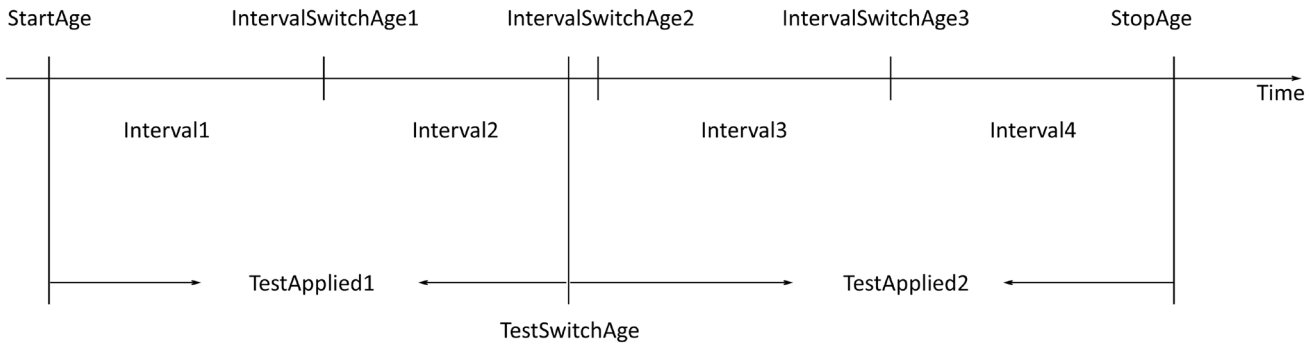

Fig. 2 Screen schedule. The terms used in the figure correspond to the parameter names used in the model code. StartAge is the age of starting the screening programme, and StopAge is the end of the screening. IntervalSwitchAge is the age that changes screening intervals, which can be changed up to three times. TestSwitchAge is the age that changes screening modality from TestApplied1 to TestApplied2

```
NumberOfScreens <- (StopAge - StartAge) / Interval + 1  # Find the
number of screens
NumberOfScreens <- ifelse(NumberOfScreens %% 1 == 0.5,
ceiling(NumberOfScreens), round(NumberOfScreens))  # Rounding the
number of screens
StopAge <- StartAge + (NumberOfScreens - 1) * Interval  # Redefine
the actual StopAge following rounding
Screens <- c(seq(StartAge, StopAge, Interval))
```

Box 5 Pseudo-code generating screening schedules

val, and this is what is presented in the code and pseudo-code (Box 5). A given screening schedule is defined on the basis of the length of the screening interval, the screening start age and the target screening stop age. The total number of screens per schedule is derived from the target age range and the screening interval. In cases in which the age range is not perfectly divisible by the interval and the remainder is 0.5 and above, we round up to generate the applied number of screens as an integer. This number of screens is then used to determine the actual stop age and, in turn, the schedule of screens.

2.2.3 Simulation of Screening Strategies

We make several assumptions to both ensure reproducible results across model iterations and reduce stochastic (first-order) uncertainty. Each simulation starts with a random number seed. The use of common seeds permits holding the natural history of each individual constant across simulations. Other sample seeds are used to ensure a fair comparison among the screening strategies. For instance, we assumed the same random seeds for individuals for common age-specific screening moments in different simulations. This assumes the probability of detecting a true positive at a given screening moment will be the same across separate simulations featuring the same screening moment. Similarly, we also assumed the same random numbers when simulating treatment success from screen-detected and symptomatically presenting disease. This is to ensure any given individual has better outcomes from screen-detected disease and treatment outcomes are comparable across strategies. The same random seeds are used for the probability of cure for symptomatic presentation across iterations.

2.2.3.1 Pseudo-code The simulation of screening employs a loop to iterate through each round of screening (Box 6). In order to eliminate stochastic error, the sample seed can be reset within each loop to permit the analysis to maintain a constant probability of disease detection within each screening moment at a given age over alternative screening schedules. For instance, this can ensure that if disease would be detected within an annual screening programme for an individual aged 30, then we can ensure disease would also be detected for an identical screen also applied at age 30 within a 5-yearly interval. To achieve this, the *set.seed* function refers to tables of random numbers related to specific ages at which screening could be applied. The analysis draws on the same random numbers for every screen at that given age. This process is used to generate random numbers for both test sensitivity and test specificity.

The analysis identifies individuals eligible for screening depending on whether they are both alive and not yet diagnosed with disease. While the earlier descriptions mentioned *Outcomes* as the fundamental output array corresponding to the disease natural history, the array *ScreenedOutcomes* is used here to record those outcomes once we account for how the natural history of disease is modified by screening and treatment for both clinically detected and screen-detected disease. The model identifies those still alive and those not yet diagnosed and finds the intersection of the two to determine all those eligible for screening at the given screening round.

The model then identifies those individuals who are in the preclinical state at the time of screening. It then identifies all those who are in the preclinical phase and eligible for screening at the given screening round and the complementary set of screen-eligible intervals who are negative at that screening moment. The model uses the vectors of all positive and negative cases and combines them with the sensitivity and specificity of the applied screening modality to generate the true and false positives.

The true positive cases detected through screening have their probability of successful treatment applied to determine who is cured of disease following treatment. Again, the model uses a seed that is held constant over individuals, using the *CureSeed* in this instance. A benefit of a consistent random seed is the avoidance of the circumstance where an individual is cured when disease was detected at the clinical stage within one strategy, while treatment was unsuccessful despite the disease being detected at an earlier preclinical phase in another strategy, even though the probability of treatment success is assumed to be higher in preclinical rather than clinical disease.

2.2.4 Cost and Effects Estimates

We considered four types of costs: primary screen, triage, early treatment and late treatment. The primary screen cost is calculated based on the total number of screens conducted regardless of their outcomes. We apply the cost of triage testing to all those primary test positives (including false positives). Early treatment is received by true-positive patients identified by screening. Conversely, late treatment is received by those individuals presenting symptomatically. The treatment costs occur when individuals undergo treatment, which is assumed to occur at a single point in time per patient.

The model includes QoL adjustment to the effectiveness. Both screening and triage incur QoL losses. QoL losses are also applied to treatment for screen-detected and symptomatic disease on a one-off basis. We assumed no QoL decrement for being in the preclinical state, including after screen detection, though we assumed a disutility for the clinical state and we assumed this also applies to screen-detected individuals once their disease progressed to the point that it would have presented symptomatically in the absence of screening. We discounted the costs and effects on a discrete annual basis using a user-defined discount rate and discount year.

2.2.5 The Cost-Effectiveness Outcomes

We record the principal cost-effectiveness outcomes, including the discounted and undiscounted costs, life-years (LYs) gained, quality-adjusted life-years (QALYs) gained, and the set of strategies on the efficiency frontier and the incremental cost-effectiveness ratios (ICERs) between them. We also record intermediate outcomes, including the age entering health states, a disaggregation of costs, and the number of screens, individuals entering the preclinical state in their disease history, true positives, false positives, cancer deaths and clinical cases. The intermediate outcomes also include the over-diagnosed cases, i.e. individuals that were screen detected but in the absence of screening would not have presented symptomatically before death.

2.3 User Interface

To make the framework accessible for those unfamiliar with R, all the parameters can be defined in Excel. An Excel template is prepared for saving parameter inputs and reading main outcomes. The inputs defined in Excel are saved in separate files, which R then imports. R is the main program to execute the model. Similarly, the main model outputs can be accessed by Excel or R, including cost-effectiveness tables with ICERs and cost-effectiveness planes. The additional results are saved in separate output files, which can be read by R.

An R Shiny app offers an intuitive interface with which to adjust model inputs and observe the changes in outputs (Fig. 3). The Shiny app does not conduct model runs itself but rather facilitates the dynamic inspection of previously calculated results. It permits sensitivity analysis for parameter values, adjustment of the cost-effectiveness threshold and quick identification of strategies of policy interest. The Shiny app also plots the intermediate outcomes, which can be useful if the user wishes to interrogate the association between strategy characteristics and cost-effectiveness.

The Shiny app allows the user to change the background settings, including analysis type (base-case vs scenario analysis), effect measurement (QALYs or LYs), results discounted or not, axis orientation and range. The user can choose the parameter they wish to vary, the parameter values (on a three-point scale), a cost-effectiveness threshold value and a specific screening strategy to observe (defined by the screening starting age, stop age and interval). The user can also select the characteristics of a particular strategy of interest, which is shown in the plot with the red marker. The

```
for (ScreenNumber in 1:nrow(ScreenCounts)){
  set.seed(ScreenCounts[ScreenNumber, "ScreenAge"])  # Fixed sample
seed for the same screening age across screening programmes
  ScreenSnSeed <- runif(SampleSize)  # Save random numbers for the
test sensitivity
  ScreenSpSeed <- runif(SampleSize)  # Save random numbers for the
test specificity


  # Define the screen age
  ScreenAge <- ScreenCounts[ScreenNumber, "ScreenAge"]
  # Retrieve the test sensitivity and specificity
  ScreenSn <- Input[paste("TestSensitivity",
ScreenCounts[ScreenNumber, "TestApplied"], sep = ""), CurrentRun]
  ScreenSp <- Input[paste("TestSpecificity",
ScreenCounts[ScreenNumber, "TestApplied"], sep = ""), CurrentRun]


  # Determine the status of individuals at the time of screening
  Alive <- ScreenedOutcomes[, "AllCauseDeath"] >= ScreenAge
  NotDiagnosed <- ScreenedOutcomes[, "Clinical"] >= ScreenAge
  NotDiagnosed[is.na(NotDiagnosed)] <- TRUE  # People who never
develop the disease are also not diagnosed
  ScreenEligible <- Alive * NotDiagnosed  # Only when both conditions
(alive and not diagnosed) are met

  # Identify those in the preclinical stage at the screen age
  Preclinical <- ScreenedOutcomes[, "Preclinical"] <= ScreenAge
  Preclinical[is.na(Preclinical)] <- FALSE  # People who never
develop the disease are not in the preclinical stage
  AllPositives <- which((Preclinical * ScreenEligible) == 1)  # Save
patient numbers that can be screened for positive
  # Identify the negatives as the complement of the positives from
within the ScreenEligible set
  AllNegatives <- which(ScreenEligible == 1)[!(which(ScreenEligible
== 1) %in% AllPositives)]

  # Find the true positives by sampling without replacement over all
positives in proportion to the test sensitivity
  TruePositives  <- AllPositives[ScreenSnSeed[AllPositives] <=
ScreenSn]
  # Find the false positives by sampling without replacement over the
negatives in proportion to the test specificity
  FalsePositives <- AllNegatives[ScreenSpSeed[AllNegatives] >=
ScreenSp]
  # Censor these successfully treated individuals
  ScreenedCured <- TruePositives[which(CureSeed[TruePositives] <=
Input["PreClinicalProbability", CurrentRun])]

  # Now update the screen-adjusted outcomes
  ScreenedOutcomes[TruePositives, "Clinical"] <- ScreenAge  # Update
the age of entering clinical state
  ScreenedOutcomes[ScreenedCured, "AllCauseDeath"] <-
ScreenedOutcomes[ScreenedCured, "OtherCauseDeath"]  # Update all-
cause death
}
```

Box 6 Pseudo-code simulating the screening intervention

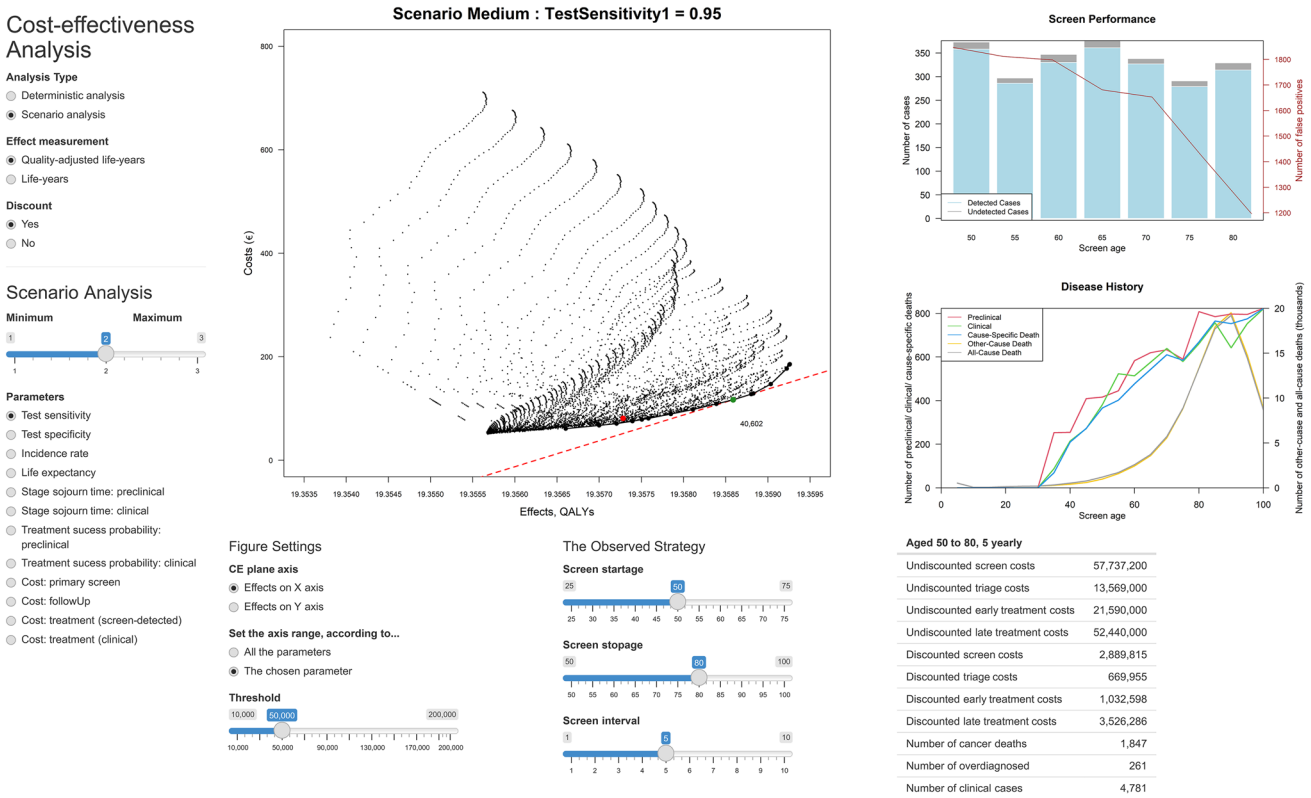

Fig. 3 The screenshot of the interface built in the R Shiny app. The central value within each parameter range corresponds to the base-case scenario in our example. *QALY* quality-adjusted life-year

intermediate outcomes for the observed strategy are also displayed, including the screen performance and the disease history.

3 An Application

To demonstrate the model, we simulate 100,000 individuals with an illustrative parameter set. Assuming the screening interval remains fixed throughout the programme and is an integer in the range of 1–10 years and screening start and stop ages are between 25 and 100, we identified all possible screening strategies. In total, 8161 screening strategies were simulated, including a no-screening strategy and one-off screening. To illustrate the relationship between parameter values and cost-effectiveness, the simulations were repeated over a range of alternative parameter values. Table 1 lists all the parameter values used in this example. The cost-effectiveness threshold in the example is €50,000 per QALY.

We employ comparative statics to show the efficiency frontiers within the cost-effectiveness plane before and after a change in parameter values. The process of comparative statics involves comparing the results of the model with a change in one or more parameters while holding all else equal. It is instructive to view two sets of results: (1) those showing the absolute costs and health effects of all strategies, including no screening and (2) those illustrating costs and health effects relative to the no-screening strategy. We separately describe our observations for the impact on cost and effect estimates.

The execution time is 1.9 min for 250 screening strategies, and 1.45 h for a complete simulation of 8161 strategies on a 3.2 GHz I7-8700 processor with 32 GB RAM.

3.1 Cost-Effectiveness Results

We identified 17 strategies on the efficiency frontier with a broad range of ICERs including a no-screening strategy in the base-case scenario (Table 2; Fig. 4). Compared to no screening, these strategies are more costly and effective. The most intensive strategies are more effective, but not cost-effective. The efficiency frontier's shape reflects diminishing marginal returns of screening intensification. In this example, the optimally cost-effective strategy in the base case is screening every 6 years from ages 35–77, with an ICER of €40,602 per QALY.

Figure 3 shows the Shiny app illustrating an example strategy of 5-yearly screening between ages 50 and 80. The results are plotted on the cost-effectiveness plane, with the strategies forming the efficiency frontier joined by the solid black line, the strategy with the optimal net health benefit shown with the green marker and the cost-effectiveness threshold shown with the dashed red line. This strategy features 4781 clinical cases, 1847 cancer deaths and 261 over-diagnosed cases.

3.2 Comparative Statics

Figures 5 and 6 demonstrate the efficiency frontier over a range of parameter values. Notably, the strategies that comprise the frontier differ as parameters change.

(1) Screen cost

Changes in screening costs result in changes in the vertical plane only as these do not influence screening effectiveness. Lower screening costs makes screening more cost-effective, reducing the ICERs of all efficient strategies. The no-screening strategy, which is located at the origin of the cost-effectiveness plane, is not influenced by changes in the costs of primary screening or those following triage, so the start of the frontier remains static across scenarios. Both relative and absolute cost estimates decrease as screening cost decreases.

(B) Treatment cost

Treatment cost changes only influence overall costs without any change in effectiveness, so strategies simply move vertically in the cost-effectiveness plane. ICERs of screening decrease when the treatment costs decrease for the screen-detected disease or increase for symptomatic disease.

Changes to the cost of screen-detected disease do not influence the position of the no-screening strategy. Consequently, the absolute and relative costs of screening fall identically when the treatment cost for screen-detected disease falls. Conversely, varying treatment costs for symptomatic disease influences costs of both screening and no-screening strategies. While absolute costs increase, they increase the most for no screening and by increasingly less as screening gets more effective. Therefore, a decline in the cost of treating symptomatically detected disease results in a fall of the cost of the no-screening scenario and the relative costs of screening rise and the cost-effectiveness of screening deteriorates.

(C) Treatment effectiveness

Changes in treatment effectiveness only result in changes in the horizontal plane. Changes to the effectiveness of screen-detected disease do not influence the position of the no-screening strategy, so the relative and absolute changes are identical. Conversely, changes to the effectiveness of symptomatically detected disease will influence the position of the no-screening scenario, and the relative and absolute outcomes differ.

An improvement in treatment effectiveness of screen-detected disease shifts the outcomes to the right and all screening strategies become more cost-effective. A reduction in treatment effectiveness of symptomatically detected disease shifts the frontier to the left in terms of the absolute estimates and shifts to the right in terms of estimates relative to no screening, and all strategies become more cost-effective.

If the effectiveness of early treatment falls to parity with that of late treatment, then there is no advantage of screening and no screening becomes the preferred strategy. There is a minimum difference in effectiveness between early and late-stage treatment required for screening to ever be beneficial. This minimal difference is required, in part, to ensure that the advantages of screening at least outweigh the QoL losses imposed by screening itself.

(D) Test performance

Changes in test sensitivity or specificity influence screening effectiveness and costs, resulting in movement in both the horizontal and vertical plane. Improved test sensitivity or specificity improves the accuracy of screening results, and screening then becomes more effective, less costly and more cost-effective. As the no-screening strategy is not influenced by changes in test performance, the absolute and relative outcomes are identical.

(E) Incidence and other-cause mortality

Varying disease incidence and other-cause mortality affect both cost and effectiveness of screening. Both parameters influence the cost and effectiveness estimates of the no-screening strategy, so the absolute and relative outcomes differ. In terms of absolute results, increased disease incidence leads to higher absolute costs and lower absolute effects for all strategies, including no screening. In terms of results relative to no screening, the converse is observed, as the relative costs of screening fall and the relative effects increase, meaning cost-effectiveness increases. Lengthening life expectancy (reducing other-cause mortality) increases both absolute and relative costs and health effects. In this example, the relative outcomes indicate the efficiency frontier moves to the right, indicating screening becomes more cost-effective.

(F) Sojourn time

In absolute terms, a lengthening of preclinical or clinical sojourn time results in increased effectiveness. Absolute

Table 1 The parameter inputs

Parameter	R name	Base case	Low	High
Sojourn time: stage 2				
Distribution	StageType2	3	3	3
Scale	StageScale2	5	3	9
Shape	StageShape2	1	1	1
Sojourn time: stage 3				
Distribution	StageType3	1.0	1.00	1
Scale	StageScale3	0.5	0.25	1
Shape	StageShape3	1.0	1.00	1
Quality of life				
Stage 1	StageUtility1	0.999	0.99	1.00
Stage 2	StageUtility2	0.900	0.70	0.95
Stage 3	StageUtility3	0.500	0.30	0.60
Test sensitivity				
Modality 1	TestSensitivity1	0.95	0.600	1
Modality 2	TestSensitivity2	0.50	0.250	1
Modality 3	TestSensitivity3	0.60	0.300	1
Modality 4	TestSensitivity4	0.65	0.325	1
Modality 5	TestSensitivity5	0.70	0.350	1
Test specificity				
Modality 1	TestSpecificity1	0.98	0.600	1
Modality 2	TestSpecificity2	0.95	0.475	1
Modality 3	TestSpecificity3	0.93	0.465	1
Modality 4	TestSpecificity4	0.90	0.450	1
Modality 5	TestSpecificity5	0.80	0.400	1
Test disutility				
Modality 1	TestDisutility1	0.001	0.0005	0.002
Modality 2	TestDisutility2	0.002	0.0010	0.004
Modality 3	TestDisutility3	0.005	0.0025	0.010
Modality 4	TestDisutility4	0.005	0.0025	0.010
Modality 5	TestDisutility5	0.005	0.0025	0.010
Probability of treatment success				
Early treatment	PreClinicalProbability	0.85	0.425	1
Late treatment	ClinicalProbability	0.40	0.200	1
Other quality-of-life burden				
Triage	DisutilityTriage	0.01	0.005	0.02
Treatment	DisutilityTrt	0.09	0.045	0.18
Discounting factors				
Rate for costs	DiscountRateCost	0.05	0.025	0.1
Rate for effects	DiscountRateEffect	0.05	0.025	0.1
Base-year	DiscountYear	0	0	5
Costs				
Screen	CostPrimaryScreen	100	50	200
Triage	CostFollowUp	1,000	500	2,000
Early treatment	CostTrtScreen	10,000	5,000	20,000
Late treatment	CostTrtClinical	20,000	10,000	40,000
Annual incidence (ages)				
0	Incidence_0	0	0	0
1–5	Incidence_5	0	0	0
6–10	Incidence_10	0	0	0
11–15	Incidence_15	0	0	0

Table 1 (continued)

Parameter	R name	Base case	Low	High
16–20	Incidence_20	0	0	0
21–25	Incidence_25	0	0	0
26–30	Incidence_30	0	0	0
31–35	Incidence_35	0.0005	0.0004167	0.00060
36–40	Incidence_40	0.0005	0.0004167	0.00060
41–45	Incidence_45	0.0008	0.0006667	0.00096
46–50	Incidence_50	0.0008	0.0006667	0.00096
51–55	Incidence_55	0.0008	0.0006667	0.00096
56–60	Incidence_60	0.0012	0.0010000	0.00144
61–65	Incidence_65	0.0012	0.0010000	0.00144
66–70	Incidence_70	0.0012	0.0010000	0.00144
71–75	Incidence_75	0.0012	0.0010000	0.00144
76–80	Incidence_80	0.0016	0.0013333	0.00192
81–85	Incidence_85	0.0016	0.0013333	0.00192
86–90	Incidence_90	0.0016	0.0013333	0.00192
91–95	Incidence_95	0.0016	0.0013333	0.00192
96–100	Incidence_100	0.0016	0.0013333	0.00192
Survival[a]				
0	Survival_0	1	1	1
5	Survival_5	0.9942	0.993968	0.994432
10	Survival_10	0.9937	0.993448	0.993952
15	Survival_15	0.9931	0.992824	0.993376
20	Survival_20	0.9919	0.991576	0.992224
25	Survival_25	0.9902	0.989808	0.990592
30	Survival_30	0.9883	0.987832	0.988768
35	Survival_35	0.9858	0.985232	0.986368
40	Survival_40	0.9821	0.981384	0.982816
45	Survival_45	0.9762	0.975248	0.977152
50	Survival_50	0.9665	0.965160	0.967840
55	Survival_55	0.9508	0.948832	0.952768
60	Survival_60	0.9272	0.924288	0.930112
65	Survival_65	0.8906	0.886224	0.894976
70	Survival_70	0.8347	0.828088	0.841312
75	Survival_75	0.7466	0.736464	0.756736
80	Survival_80	0.6127	0.597208	0.628192
85	Survival_85	0.4344	0.411776	0.457024
90	Survival_90	0.2398	0.209392	0.270208
95	Survival_95	0.0892	0.052768	0.125632
100	Survival_100	0	0	0

[a]The probability of surviving at a particular year of age

costs reduce with an increase in the preclinical sojourn time but remain unchanged with the clinical sojourn time. In relative terms, lengthening the preclinical sojourn time has an ambiguous effect on effectiveness and cost-effectiveness. Some low-intensity strategies become relatively more effective, but higher intensity strategies become relatively less effective. Consequently, the ICERs fall for some strategies but rise for others. The relative outcomes for lengthening the clinical sojourn time are unambiguous as effects fall and costs remain unchanged, meaning the ICERs rise for all strategies as screening becomes less cost-effective.

4 Discussion

We provide a simplified screening CEA microsimulation for teaching and research purposes. As an initial application, we present an assessment over a large range of strategies

Table 2 The cost-effectiveness results for the base-case scenario for strategies that lie on the efficiency frontier

Strategy				Undiscounted outcomes			Discounted outcomes			ICER, €/QALY
Label	Start age	Stop age	Interval	LYs	QALYs	Costs (€)	LYs	QALYs	Costs (€)	
A	No screening			79.801947	79.516654	904.00	19.365480	19.337555	50.498	–
B	46	46	0	79.853977	79.541461	990.91	19.367595	19.338478	61.127	11,515
C	46	56	10	79.886550	79.556584	1073.01	19.368548	19.338875	67.361	15,712
D	46	66	10	79.910897	79.567635	1146.67	19.369057	19.339082	70.908	17,162
E	46	70	8	79.930444	79.576316	1230.96	19.369545	19.339270	75.571	24,743
F	45	66	7	79.934988	79.578538	1239.68	19.369834	19.339381	78.626	27,613
G	45	73	7	79.947866	79.583990	1311.88	19.370050	19.339459	80.960	29,814
H	45	70	5	79.967722	79.592732	1418.12	19.370769	19.339725	89.344	31,578
I	38	73	7	79.973945	79.595844	1415.95	19.371485	19.339986	98.124	33,531
J	35	77	7	79.991348	79.603523	1501.67	19.372273	19.340264	108.922	38,883
K	35	77	6	80.005383	79.609435	1596.73	19.372850	19.340463	116.996	40,602
L	35	75	5	80.021411	79.616258	1707.34	19.373513	19.340678	127.768	50,211
M	35	80	5	80.027628	79.618563	1766.25	19.373595	19.340702	129.106	54,420
N	35	83	4	80.053901	79.628843	2021.69	19.374402	19.340910	147.236	87,398
O	35	83	3	80.080173	79.637804	2409.42	19.375412	19.341099	177.200	158,385
P	34	82	3	80.085183	79.640079	2436.57	19.375637	19.341132	184.887	233,353
Q	34	85	3	80.087298	79.640566	2485.51	19.375661	19.341135	185.699	282,266

ICER incremental cost-effectiveness ratio, *LY* life-year, *QALY* quality-adjusted life-year

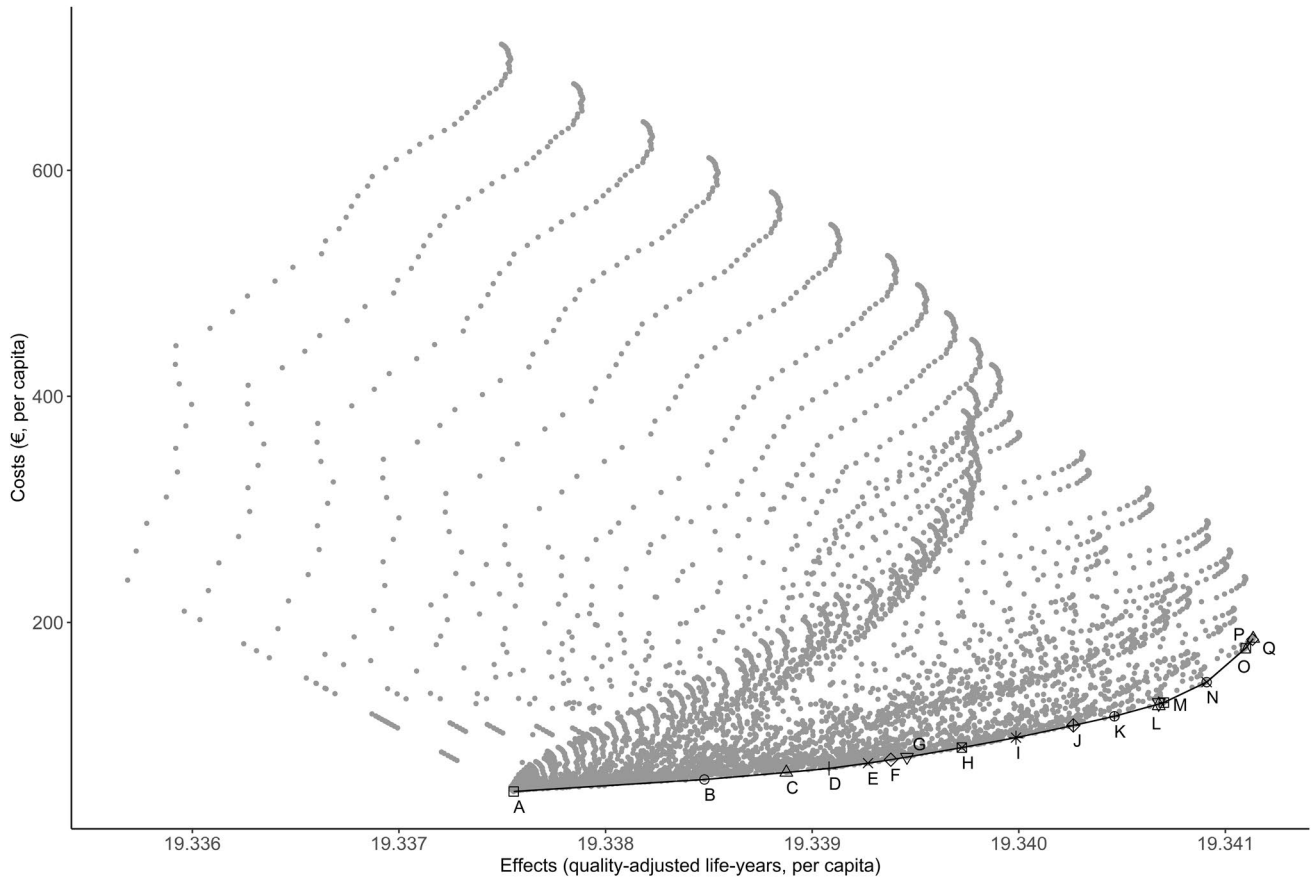

Fig. 4 The cost-effectiveness plane for the base-case scenario with the efficiency frontier shown in black and efficient strategies marked A to Q

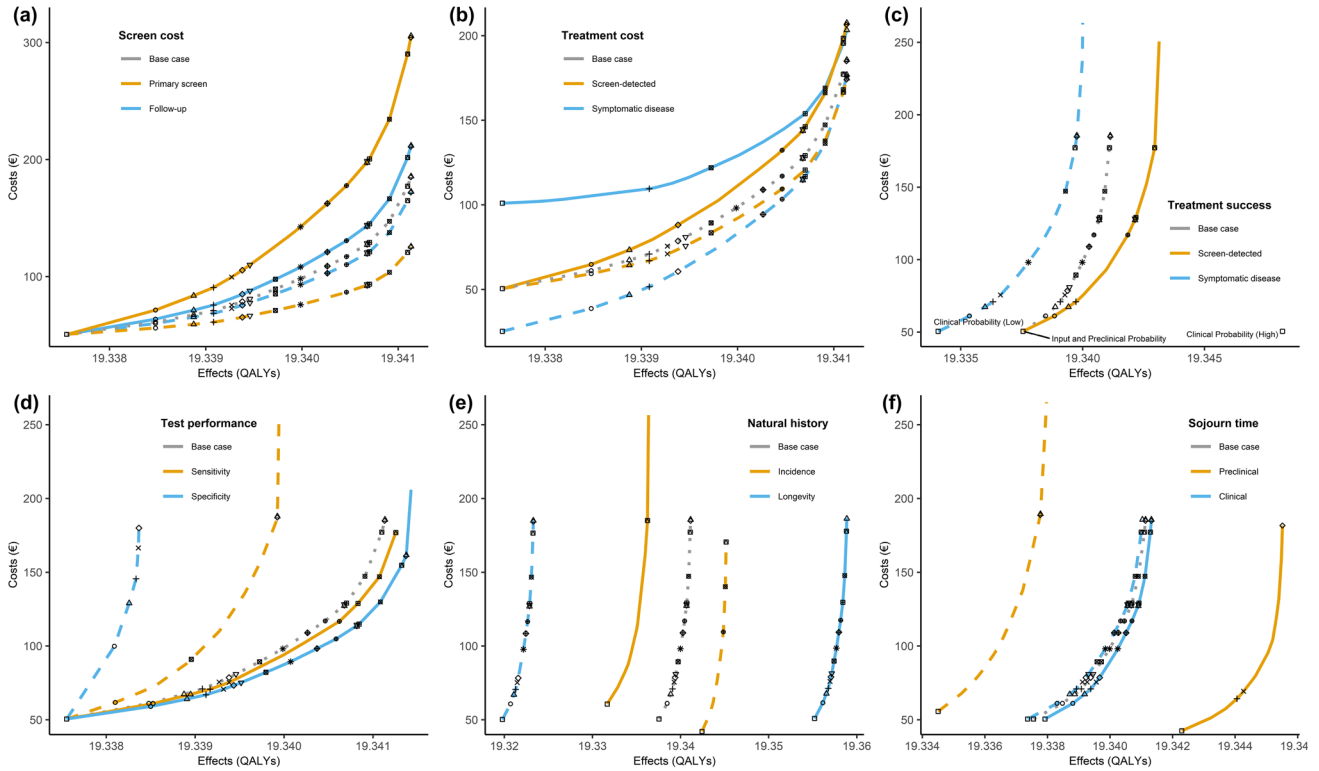

Fig. 5 The cost-effectiveness efficiency frontier as specific input parameters are varied (absolute results). *QALY* quality-adjusted life-year. (i) Solid line represents the higher-value scenario, and dashed lines are the low-value scenarios; the shape of the markers correspond to the strategies in Fig. 4. (ii) In the scenarios of high treatment success for symptomatic disease and low for screen-detected disease, the no-screening strategy becomes the only comparator on the efficiency frontier. (iii) Note that the range of the axes may vary between plots to better illustrate the shape change across the scenarios of each parameter sets

and conduct comparative statics to illustrate the influence of parameters on cost-effectiveness. Our results illustrate the relevance of considering both absolute costs and effects and those relative to no screening. This distinction between absolute and relative outcomes is useful when seeking to demonstrate the intuition behind the observed results. Our analysis conveys the intuition of the relationship between parameter values and outcomes, informing the process of model validation.

To our knowledge, this is the first CEA teaching model published in the specific context of disease screening. While this model does not necessarily correspond to screening for any specific disease, the example presented broadly corresponds to screening for cancer. The framework can, nevertheless, also be applied to other interventions such as periodic dental exams, eye exams, and hepatitis screening.

Our framework is accessible and editable by all as the complete model code and variable inputs are specified and provided online. Users are able to apply and extend the model without concerns of copyright infringement. The full access contributes to research transparency and facilitates sharing knowledge of simulation methodology. As such, our model is intended as a public good, and we hope its dissemination will benefit the field of CEA in disease screening.

An important advantage of our model is its simplicity and speed. Compared to specialised commercial software or spreadsheet applications such as Microsoft Excel, the non-proprietorial nature of R permits an accessible, transparent and adaptable model platform [27, 28]. R is increasingly adopted as the modelling tool, with the support of a large range of open-source materials and well-documented packages and functions [29]. Importantly, models written in R are now accepted by the National Institute for Health and Care Excellence (NICE) [30]. Although there are some published tutorials for modelling in R [16, 17, 28, 31], these models are not applied to screening and do not employ DES. As such, our model offers a novel contribution to the growing R in CEA literature.

As a teaching tool, our model is intended for two groups. First, it can serve as a teaching tool for the students who want to understand the principles of economic evaluation regarding screening interventions. The intuitive interfaces of Excel and Shiny ensure that students do not need to understand R programming as they are able to explore alternative screening policies under different scenarios and threshold

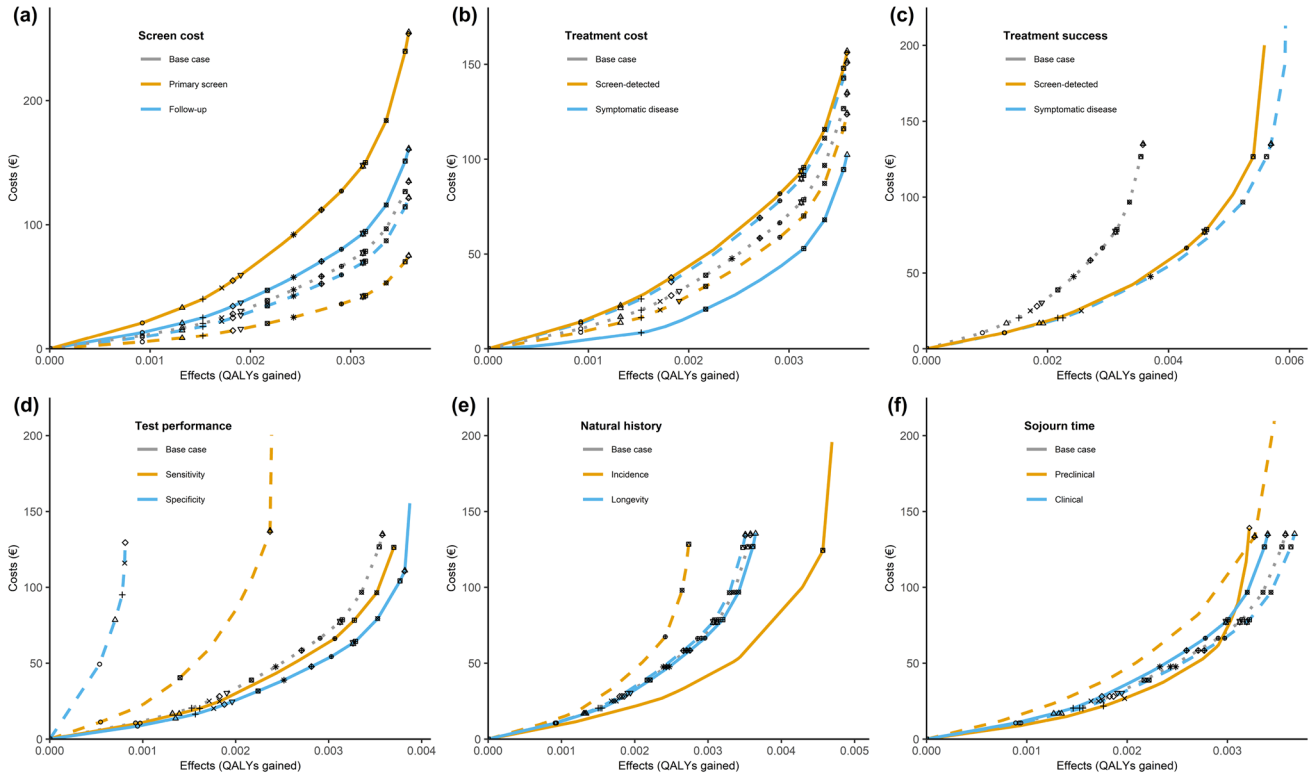

Fig. 6 The cost-effectiveness efficiency frontier as specific input parameters are varied (results relative to no screening). *QALY* quality-adjusted life-year. (i) Solid line represents the higher-value scenario, and dashed lines are the low-value scenarios; the shape of the markers correspond to the strategies in Fig. 4. (ii) In the scenarios of high treatment success for symptomatic disease and low for screen-detected disease, the no-screening strategy becomes the only comparator on the efficiency frontier. (iii) Note that the range of the axes may vary between plots to better illustrate the shape change across the scenarios of each parameter sets

values without having to operate or modify R code. Our Shiny app offers a convenient interface for the examination of changes to parameter values on cost-effectiveness estimates. Second, our model serves as a resource for those intending to learn DES programming in R. Our model provides a starting point for extensions to other implementations, either methodological or applied.

Our simplified model is suitable for the purpose of demonstrating the relationship between key parameter values and cost-effectiveness. Its simplified nature makes demonstrating face validity straightforward. While our simplified model can help modellers develop their understanding of screening, any specific modelling application requires independent, context-specific demonstration of validation. As such, any extension of our model might require a renewed exercise in face validity depending on how extensive the changes are.

Our model deliberately employs a high degree of abstraction to make it accessible and efficient. Although this simulation only has five stages, it is sufficient to demonstrate the fundamentals of screening cost-effectiveness. As an abstracted model it is not suitable for solving applied research questions regarding specific prevention programmes. Rather, it is intended as offering a basis for addressing methods research questions. Potential applications include methods demonstrations of alternative forms of risk stratification, the differences between models of single and multiple birth cohorts and illustrating the consequences of omitting screening strategies.

Our model deliberately eliminates some sources of stochastic error by preserving random seeds for chance events regarding both screening and treatment success. These can help the model yield consistent results across alternative screening strategies with smaller sample sizes. Care must be taken, however, to ensure that the elimination of this stochastic error does not itself cause artefacts in the simulation estimates, especially with smaller simulation sample sizes. An alternative approach is to relax the assumptions around these common random seeds and simply to inflate the simulation size to attenuate the effects of random error, though this can come at the cost of model run time.

Naturally, our model has limitations. At a minimum, users must at least be able to install and run R. They will need to install the Shiny package if they wish to use our Shiny app. Our deliberate avoidance of packages results in minor imperfections in the presentation of overlapping ICERs within the cost-effectiveness plane. A consequence of the degree

of abstraction adopted in the model is that its structure and parameter values are merely notional and do not correspond directly to any specific disease. For example, a one-off treatment cost in our model cannot illustrate the treatment cost correlated with the severity of the disease or the length of hospital stay. The distinction between the treatments for screen-detected and symptomatically detected diseases is a simplistic representation of early and late-stage therapy. The model also does not include palliative care costs and death-related expenses. Furthermore, although our model achieves a fast runtime, it will not retain such speeds when extended to the multiple health states and complex screening and triage algorithms required in applied analyses. Further adaptations might require integration with C++. Another limitation is that this initial demonstration does not explore parameter uncertainty, although the comparative statics framework presented can naturally be used as a template for one-way sensitivity analysis. Adding probabilistic sensitivity analyses or exploring multivariate impacts is an obvious future extension.

5 Conclusion

We present a simple microsimulation model of the cost-effectiveness of screening. Our model is the first open-source DES CEA model of screening coded in R. It is specifically intended to overcome the constraints of the models typically applied in cancer screening, which are both large and not openly shared. In this initial application we simulated thousands of screening strategies as an example to illustrate how the efficiency frontier moves when parameter changes through a series of comparative statics. This permits a demonstration of face validity and is intended to aid modellers' understanding of screening cost-effectiveness. We hope our model will serve as a useful basis for methods research and as a teaching tool.

Acknowledgements We thank Dr. Stefan Lhachimi for his advice on the initial model development.

Declarations

Funding This study was funded by the Health Research Board of Ireland (Grant reference number EIA 2017-054).

Conflict of interest All authors have no conflicts of interest or competing interests that are relevant to this work.

Data availability The datasets generated during and/or analysed during the current study are available in the GitHub repository (https://github.com/yishu-lin/Pedagogical-CEA-Model-of-Screening.git).

Code availability All the code used in this tutorial, the Excel file and the R Shiny template can be found at https://github.com/yishu-lin/Pedagogical-CEA-Model-of-Screening.git. This link also has additional step-by-step supporting material for the R model.

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

Author contributions JOM designed the model and drafted the initial model structure and code. YSL substantially revised the model structure and extended its application with the discussion with all authors. YSL wrote the first draft of the manuscript, and JOM critically revised and proofread the manuscript. All authors approved the final manuscript.

References

1. Weinstein MC, et al. Cost-effectiveness in health and medicine. Oxford: Oxford University Press; 1996.
2. Scheffler RM, Paringer L. A review of the economic evidence on prevention. Med Care. 1980. https://doi.org/10.1097/00005650-198005000-00001.
3. Knudsen AB, McMahon PM, Gazelle GS. Use of modeling to evaluate the cost-effectiveness of cancer screening programs. J Clin Oncol. 2007;25(2):203–8.
4. Rutter CM, Zaslavsky AM, Feuer EJ. Dynamic microsimulation models for health outcomes: a review. Med Decis Mak. 2011;31(1):10–8.
5. Mandelblatt JS, et al. Assessing the effectiveness of health interventions for cost-effectiveness analysis. J Gen Intern Med. 1997;12(9):551–8.
6. Ramsey SD, et al. Simulation modeling of outcomes and cost effectiveness. Hematol Oncol Clin North Am. 2000;14(4):925–38.
7. Byford S, Palmer S. Common errors and controversies in pharmacoeconomic analyses. Pharmacoeconomics. 1998;13(6):659–66.
8. Drummond M, Sculpher M. Common methodological flaws in economic evaluations. Med Care. 2005;43(7 Suppl):5–14.
9. O'Mahony JF, et al. Beware of kinked frontiers: a systematic review of the choice of comparator strategies in cost-effectiveness analyses of human papillomavirus testing in cervical screening. Value Health. 2015;18(8):1138–51.
10. Lin YS, Pokharel R, O'Mahony JF. MS1 cost-effectiveness analysis of colorectal cancer screening programmes: how omissions of relevant comparators CAN lead to inefficient policies. Value Health. 2020;23:S405.
11. O'Mahony JF. Interpreting cost-effectiveness ratios in a cost-effectiveness analysis of risk-tailored prostate screening: a critique of

Callender et al. HRB Open Res. 2020. https://doi.org/10.12688/hrbopenres.13043.2.
12. O'Mahony JF. Risk stratification in cost-effectiveness analyses of cancer screening: intervention eligibility, strategy choice, and optimality. Med Decis Mak. 2022;42(4):513–23.
13. Ten Haaf K, et al. Personalising lung cancer screening: an overview of risk-stratification opportunities and challenges. Int J Cancer. 2021;149(2):250–63.
14. Prakash MK, et al. CMOST: an open-source framework for the microsimulation of colorectal cancer screening strategies. BMC Med Inform Decis Mak. 2017. https://doi.org/10.1186/s12911-017-0458-9.
15. Green C, et al. Assessing cost-effectiveness of early intervention in Alzheimer's disease: an open-source modeling framework. Alzheimers Dement. 2019;15(10):1309–21.
16. Chilcott J, et al. Avoiding and identifying errors in health technology assessment models: qualitative study and methodological review. Health Technol Assess (Winchester, England). 2010;14(25):iii–iv, ix.
17. Eddy DM, et al. Model transparency and validation: a report of the ISPOR-SMDM Modeling Good Research Practices Task Force-7. Med Decis Mak. 2012;32(5):733–43.
18. Kirch RLA, Klein M. Examination schedules for breast cancer. Cancer. 1974;33(5):1444–50.
19. Day NE, Walter SD. Simplified models of screening for chronic disease: estimation procedures from mass screening programmes. Biometrics. 1984;40(1):1–14.
20. Eddy DM. A Mathematical model for timing repeated medical tests. Med Decis Mak. 1983;3(1):45–62.
21. Smith H, et al. Simulation modeling validity and utility in colorectal cancer screening delivery: a systematic review. J Am Med Inform Assoc. 2020;27(6):908–16.
22. Green N, et al. Health economic evaluation using Markov models in R for Microsoft excel users: a tutorial. Pharmacoeconomics. 2022. https://doi.org/10.1007/s40273-022-01199-7.
23. Krijkamp EM, et al. Microsimulation modeling for health decision sciences using R: a tutorial. Med Decis Mak. 2018;38(3):400–22.
24. Krijkamp EM, et al. A Multidimensional array representation of state-transition model dynamics. Med Decis Mak. 2020. https://doi.org/10.1177/0272989x19893973.
25. Alarid-Escudero F, et al. A tutorial on time-dependent cohort state-transition models in R using a cost-effectiveness analysis example. Med Decis Mak. 2022. https://doi.org/10.1177/0272989x221121747.
26. Davis S, Stevenson M, Tappenden P, Wailoo A. NICE decision support unit technical support documents. In: NICE DSU technical support document 15: cost-effectiveness modelling using patient-level simulation. National Institute for Health and Care Excellence (NICE); 2014.
27. Incerti D, et al. R you still using Excel? The advantages of modern software tools for health technology assessment. Value Health. 2019;22(5):575–9.
28. Frederix GW, et al. Development of a framework for cohort simulation in cost-effectiveness analyses using a multistep ordinary differential equation solver algorithm in R. Med Decis Mak. 2013;33(6):780–92.
29. Jalal H, et al. An overview of R in health decision sciences. Med Decis Mak. 2017;37(7):735–46.
30. National Institute for Health and Care Excellence (NICE).Guide to the pro-cesses of technology appraisal. 2014. https://www.nice.org.uk/process/pmg19.
31. Williams C, et al. Cost-effectiveness analysis in R using a multistate modeling survival analysis framework: a tutorial. Med Decis Mak. 2017;37(4):340–52.