



Zorginstituut Nederland

Richtlijn voor kosteneffectiviteitsmodellen in R

Datum	December 2022
Status	Concept

Colofon

Zaaknummer	2021045180
Volgnummer	2022041097
Contactpersoon	Dhr. dr. M. El Alili MElalili@zinl.nl
Auteur(s)	Dhr. dr. M. El Alili Dhr. drs. J. Enzing Mw. dr. H.A. Geuzinge Dhr. dr. R.A. Vreman (projectleider)
Afdeling Team	Zorg Geneesmiddelen Ontwikkeling, Wetenschap en Internationale Zaken (OWIZ)

Inhoud

	1	Inleiding	5
	2	Richtlijn	7
2.1		Doel van deze richtlijn	7
2.2		Algemene principes voor KE-modellen in R	7
2.3		End-to-endfuncties	7
2.4		Gegevensvereisten	7
2.5		Mapstructuur en readme	7
2.6		Codestructuur en R Markdown	8
2.7		Coderingsconventies en commentariëren	9
2.8		Packages en functies	10
2.9		Testen	10
2.10		Extra functies	11
2.11		Indieningsprocedure	11
2.12		Zaken die niet in deze richtlijn aan de orde komen	11
	3	Referenties	13

1 Inleiding

Microsoft Excel is momenteel de meest gebruikte software om kosteneffectiviteitsmodellen (KE-modellen) te bouwen voor health technology assessments (HTA, beoordelingen van gezondheidstechnologie). Excel biedt veel voordelen omdat het een begrijpelijk, intuïtief programma is met veel functies en waarvoor veel documentatie beschikbaar is. Daarnaast hebben HTA-organisaties uitgebreide ervaring met Excel. In sommige gevallen kan Excel volstaan als software, maar naarmate de complexiteit van KE-modellen toeneemt, komt het steeds vaker voor dat Excel onpraktisch of zelfs niet bruikbaar is voor het bouwen van KE-modellen. De beperkingen van Excel hebben met name betrekking op het toenemende gebruik van kansverdelingen, het gebruik van veel afzonderlijke maar gekoppelde tabbladen, het gebruik van steeds complexere Visual Basic for Applications (VBA) code, en de complexere vereisten voor statistische berekeningen en visualisaties.^[1, 2] Om toekomstbestendig te blijven moeten HTA-organisaties mogelijkheden bieden die verder gaan dan Microsoft Excel.

Een vaak genoemd alternatief voor Excel is R.^[2, 1] R wordt nu al door veel mensen gebruikt voor het bouwen van KE-modellen. Het wordt aanbevolen vanwege de mogelijkheden om statistische analyses, geavanceerde modellen en goede documentatie te integreren. Wanneer de modellen in R goed worden uitgevoerd, hebben deze de potentie om efficiënter, transparanter, beter aanpasbaar en beter herbruikbaar te zijn dan modellen in Excel.

Dit document omschrijft de richtlijn voor het creëren en indienen van KE-modellen in R voor vergoedingsverzoeken aan Zorginstituut Nederland. Deze richtlijn is voornamelijk gebaseerd op gepubliceerd werk van twee consortia op dit gebied: DARTH (Decision Analysis in R for Technologies in Health) en het consortium 'R for HTA'.

2 Richtlijn

2.1 Doel van deze richtlijn

De vrijheid in het bouwen van modellen die inherent is aan het gebruik van R kan voor problemen zorgen wanneer dit niet goed wordt afgebakend. Om ervoor te zorgen dat de KE-modellen in R transparant, betrouwbaar, controleerbaar, makkelijk aan te passen en over het algemeen geschikt zijn voor het beoogde doel, wordt een zekere mate van standaardisatie noodzakelijk geacht. Daarom is het doel van deze richtlijn het opstellen van de voorwaarden waar KE-modellen in R voor HTA-besluitvorming aan moeten voldoen. Budgetimpactanalyses vallen buiten het bereik van deze richtlijn. Deze richtlijn is een aanvulling op de bestaande richtlijn voor het uitvoeren van economische evaluaties in de gezondheidszorg en is niet bedoeld ter vervanging van bestaande richtlijnen. Wanneer aanbevelingen tegenstrijdig zijn, moet de algemene richtlijn voor economische evaluaties in de gezondheidszorg worden gevolgd.^[3]

2.2 Algemene principes voor KE-modellen in R

De algemene principes voor modellen in R zijn in wezen gelijk aan die voor modellen in andere soorten software. Over het algemeen geldt dat het model:

- Op de juiste wijze gestructureerd is;
- Volledig transparant is;
- Voldoende gevalideerd is;
- Volledig uitvoerbaar en aanpasbaar is;
- Duidelijk en uitgebreid gedocumenteerd is.

Er moet worden gecontroleerd of andere gebruikers dan de oorspronkelijke auteur het model zonder fouten kunnen draaien. Beoordelaars van het Zorginstituut moeten gemakkelijk parameters kunnen wijzigen en analyses opnieuw kunnen draaien in een Windows-omgeving.

2.3 End-to-end functionaliteiten

Een voordeel van R is dat het 'end-to-end' functionaliteit mogelijk maakt. Dit betekent dat alle statistische analyses in R kunnen worden gecodeerd die eerder (in de situatie met Excel) apart van het KE-model werden uitgevoerd. Een voorbeeld hiervan zijn extrapolaties van overlevingscurves die niet in Excel kunnen worden uitgevoerd. Het huidige standpunt van het Zorginstituut is dat KE-modellen met 'end-to-end' functionaliteit worden aangemoedigd, maar dit is niet verplicht. KE-modellen in R moeten ten minste alle functionaliteiten bevatten die ook in Excel zouden zijn opgenomen.

2.4 Gegevensvereisten

Alle noodzakelijke gegevens moeten bij de indiening worden bijgevoegd om het Zorginstituut in staat te stellen het model te kunnen draaien. Gegevens kunnen worden geaggregeerd tot een niveau zoals dat ook zou worden samengevoegd voor een model dat is gebouwd in Excel (zo zijn bijvoorbeeld gegevens van individuele patiënten doorgaans niet vereist). In lijn met de opties voor 'end-to-end' functionaliteiten zijn gegevens alleen nodig voor zover dat ook nodig zou zijn voor een model dat in Excel is gebouwd.

2.5 Mappenstructuur en readme

Er moet een 'readme'-bestand of metadatabestand worden aangeleverd als uitgangspunt voor de gebruikers van het model. Deze bestanden moeten ten minste

het volgende uitleggen: de mappenstructuur, het doel van de bestanden in elke map en hoe deze bestanden met elkaar samenhangen, de gebruikte coderingsconventies, de manier waarop fouten worden verwerkt en elk ander punt dat relevant is voor de gebruiker om het model te begrijpen en te kunnen draaien. Het volledige project moet worden opgeslagen als een Rstudio-project. Een Rstudio-project is een zelfstandige werkomgeving. Hierdoor hoeft u niet meer aan te geven waar bestanden zich bevinden wanneer het model op verschillende computers wordt gebruikt. Alle basisparameters (met hun kansverdeling en bereik) moeten worden gedefinieerd in gegevensbestanden (.csv wordt aanbevolen, inclusief informatie over de bronnen van de inputdata) die in het model moeten worden geladen (zie '03: Laden van inputdata' in hoofdstuk 2.6). Het moet mogelijk zijn om inputdatabestanden te controleren zonder het gebruik van R (het is bijvoorbeeld niet toegestaan om inputdata op te nemen als .Rdata-bestand). Bestanden met inputdata moeten het mogelijk maken dat alternatieve base-case parameters en bereiken kunnen worden getest. Voor dit doel moet in de inputdatabestanden gespecificeerd worden wat de base-case parameters zijn met referenties, het bereik rondom deze parameters, en het type verdeling dat wordt toegepast. De verdelingsparameters moeten zelf niet in deze bestanden worden gespecificeerd (bijv. alfa en bèta voor de bètadistributie): Deze moeten zelf moeten binnen het script gegenereerd worden, zodat ze opnieuw kunnen worden berekend op basis van alternatieve base-case parameters en bereiken die door de gebruiker zijn opgegeven.

Het is verplicht om een mappenstructuur te gebruiken die ten minste gegevens, code en resultaten van elkaar scheidt. De aanbevolen mappenstructuur is:

- I. **Readme**
- II. **Data:** In deze map moet alle data worden opgeslagen. Deze map kan ruwe data (in submappen), geaggregeerde data en een bestand met extern afgeleide parameterwaarden bevatten. De data kunnen worden opgeslagen als .csv-bestanden, R-gegevensbestanden (met de extensies .RData, .rds of .rda) of andere bestandsextensies. Inputdata moeten worden opgeslagen met een bestandsextensie die kan worden geopend zonder R (zie hierboven). De gegevensbestanden moeten worden genummerd, op basis van de nummers van de delen van de code waarin ze worden gebruikt (zie hoofdstuk 2.6).
- III. **Scripts:** In deze map moeten de belangrijkste R-scripts worden opgeslagen. Functies en unit-tests moeten in afzonderlijke scripts worden opgeslagen. Scripts moeten worden genummerd in de volgorde waarin ze worden uitgevoerd.
- IV. **Functies:** In deze map moeten .R-bestanden met door de gebruiker gedefinieerde functies worden opgeslagen.
- V. **Output:** In deze map moeten tussentijdse en definitieve outputgegevens worden opgeslagen in submappen. Soms moeten door het model gegenereerde outputgegevens in andere delen van het model opnieuw worden gebruikt. Deze tussentijdse resultaten kunnen hier worden opgeslagen.
- VI. **Tabellen en figuren:** In deze map moeten tabellen en figuren worden opgeslagen die worden gegenereerd voor het uiteindelijke dossier.
- VII. **Tests:** Deze map moet ten minste één R-script bevatten waarin de unittests zijn gecodeerd. In hoofdstuk 2.9 vindt u een beschrijving van de aanbevolen unittests.

2.6 Codestructuur en R Markdown

De R-code moet 'stand-alone' gerund kunnen worden. Alle codes (behalve extra

functies die geen deel uitmaken van bestaande pakketten en de unittests, zie hoofdstuk 2.9) moeten in één bestand staan. R Markdown moet worden gebruikt om de code in codeblokken en verklarende tekst te structureren. Ondanks de optie om rapporten te genereren met R Markdown, moet het officiële rapport nog steeds in de Word-template van het Zorginstituut worden geschreven. De volgende paragrafen moeten worden gedefinieerd in het R Markdown script:

01 Installatie

Bevat alle informatie om het model operationeel te maken. In dit hoofdstuk moeten ook de versies van R en van de benodigde packages worden vermeld die op het moment van indiening zijn gebruikt.

02 Inladen functies

03 Inladen van inputparameters

Deze paragraaf moet in de volgende onderdelen worden verdeeld (indien nodig kunnen er onderdelen worden toegevoegd):

- 03.a Patiëntkenmerken
- 03.b Effectiviteitsparameters (bijv. transitiekansen of extrapolatie van overlevingscurve)
- 03.c Kansen op bijwerkingen
- 03.d Utiliteiten
- 03.e Kosten

04 Modelsettings

Dit deel moet de modelsettings bevatten voor het modeltype, het aantal gezondheidstoestanden, de tijdshorizon, de cycluslengte, het perspectief, enz.

05 Base-case analyse en resultaten

06 Deterministische gevoeligheidsanalyses en resultaten

07 Probabilistische gevoeligheidsanalyse en resultaten

08 Scenarioanalyses en resultaten

09 Value of information (VOI) (optioneel)

10 Validatie

2.7

Coderingsconventies en beschrijvingen van de codes

Beschrijvingen van de codes

Elke functie en/of codeblok moet duidelijk worden toegelicht om het doel van de code te beschrijven en om het gemakkelijk te maken om te controleren of de code naar behoren functioneert. De beschrijvingen moet uitgebreid, gedetailleerd en duidelijk zijn. Voorbeelden van duidelijke R-code en adequate beschrijvingen worden bijvoorbeeld gegeven op de websites van de R for HTA- en DARTH-consortia (<https://r-hta.org/publication/>; <http://darthworkgroup.com/r-code/>).

Coderingsconventies

De tidyverse-stijlgids (verkrijgbaar via <https://style.tidyverse.org>) bevat algemene aanbevelingen voor goede coderingspraktijken die moeten worden nageleefd. De aanbevelingen in de tidyverse-stijlgids kunnen worden aangevuld met specifieke stijlen voor KE-modellering. Het DARTH Coding Framework presenteert conventies voor bestandsnamen en variabelen die sterk worden aanbevolen.^[4] Het gebruik van andere stijlgidsen is toegestaan, maar ze moeten volledig worden uitgelegd in het readme- of het metadatabestand (zie hoofdstuk 2.5). De namen van variabelen moeten ten minste de specificatie van het type data en het type variabele bevatten.

In tabel 1 zijn de namen van variabelen volgens DARTH weergegeven.^[4] Namen van variabelen en functies moeten *underscores* (lage streepjes) bevatten om woorden te scheiden en mogen alleen kleine letters bevatten. Gebruik de volgende naamgevingsstructuur: <x>_<y>_<var_name>, waarbij <x> het type data aangeeft, <y> het type variabele en <var_name> een korte, maar duidelijke naam

die de variabele beschrijft.^[4] Een voorbeeld is: `v_r_mort_by_age` (vector, rate, mortaliteit naar leeftijd).

Tabel 1: Aanbevolen voorvoegsels. Deze lijst is indicatief en is niet noodzakelijkerwijs volledig.

Gegevenstype	Voorvoegsel	Type variabele	Voorvoegsel
Vector	v	Number	n
Matrix	m	Probability	p
Array	a	Rate	r
Data frame	df	Utility	u
Data table	dtb	Cost	c
List	l	Hazard ratio	hr
Scalar	(geen voorvoegsel)	Relative risk	rr
		Life years	ly
		QALYs	q
		Standard error	se

Bron: Alarid-Escudero et al.^[4]

2.8 Packages en functies

Als uitbreiding van de standaardfuncties van R (base R) zijn er verschillende packages beschikbaar. Deze packages bevatten een verzameling functies, gegevens en documentatie. Voordat deze packages kunnen worden gebruikt, moeten ze worden geïnstalleerd. De lijst van het Zorginstituut met toegestane packages om de verschillende soorten analyses uit te voeren in een KE-model, is te vinden in de online bijlage van deze richtlijn (bijlage A). Binnen deze lijst worden de beschikbare packages gecategoriseerd op basis van hun rol in de schatting van de kosteneffectiviteit, d.w.z. ondersteunende packages, voorbereidende packages, kosteneffectiviteitsanalyse (KEA), deterministische gevoeligheidsanalyse (DSA) en probabilistische gevoeligheidsanalyse (PSA), VOI analyse en modelkalibratie. Gewoonlijk bevatten deze packages alle functies voor de vereiste onderdelen van een KEA. Het gebruik van bestaande packages wordt sterk aangeraden. Wanneer er geen bestaande packages worden gebruikt, moet een duidelijke verklaring worden gegeven waarom de beschikbare functies van deze packages niet toereikend zijn. In het geval dat bestaande packages absoluut niet geschikt zijn, kunnen extra functies worden geschreven met behulp van base R.

Het Zorginstituut staat alleen het gebruik toe van packages die beschikbaar zijn via het Comprehensive R Archive Network (CRAN: <https://cran.r-project.org/>). Het Zorginstituut streeft ernaar een uitgebreide lijst van packages te bieden. Daarom zijn packages, die buiten de lijst in de online bijlage (Bijlage A) vallen, in principe niet toegestaan. Packages die in de toekomst worden ontwikkeld, kunnen echter wel worden toegevoegd; vooral in de vroege versies van deze richtlijn is het mogelijk dat er belangrijke packages ontbreken. Als de intentie bestaat om packages te gebruiken die niet in de lijst worden vermeld, moet contact worden opgenomen met het Zorginstituut.

2.9 Testen

Unittests moeten worden uitgevoerd om ervoor te zorgen dat delen van het model naar verwachting werken, en dat hun onderlinge afhankelijkheid valide is. Er zou een reeks tests moeten zijn voor elke additionele functie die wordt geschreven. Daarnaast kunnen tests worden opgenomen die grotere componenten van het model testen. We raden u aan het *testthat* package te gebruiken om unittests uit te voeren (<https://cran.r-project.org/web/packages/testthat/>). Gebruik van het *testthat* package is niet verplicht; model componenten kunnen ook worden

gecontroleerd met andere functies. DARTH heeft uitgewerkte voorbeelden van unittests beschreven binnen de context van KE-modellen.^[4, 5]

2.10 Extra functies

R shiny interfaces zijn niet vereist voor ingediende KE-modellen maar zijn wel toegestaan. Alle onderliggende codes moet wel worden ingediend. Eventuele extra gebruikersfuncties worden niet gecontroleerd door beoordelaars en worden ook niet opgenomen in de definitieve HTA-dossiers opgesteld door het Zorginstituut.

2.11 Indieningsprocedure

De algemene indieningsprocedure is van toepassing. Alle bestanden en gegevens die nodig zijn om het model te draaien moeten in één map worden ingediend (met submappen volgens de mappenstructuur uit hoofdstuk 2.5). De omschrijvingen van de mappen en code hoeven geen deel uit te maken van het ingediende rapport, maar kunnen worden opgenomen in het readme-bestand in de daarvoor bedoelde R-map. Het geschreven rapport zal dus nagenoeg hetzelfde zijn als rapporten die worden ingediend met modellen die in Excel zijn gebouwd, met uitzondering van technische specificaties.

2.12 Zaken die niet in deze richtlijn aan de orde komen

In het geval bepaalde zaken niet worden besproken in deze richtlijn, in het geval van dubbelzinnige uitspraken, of als er een wens bestaat om extra functies te implementeren die hier niet worden beschreven, fungeren de algemene principes die worden beschreven in hoofdstuk 2.2 van deze richtlijn als de basiscriteria waar alle KE-modellen in R volledig aan moeten voldoen. Wees voorzichtig met de introductie van functies die buiten het bereik van deze richtlijn vallen. Eventuele belangrijke wijzigingen (bijv. het gebruik van extra packages) moeten vooraf met het Zorginstituut worden gecommuniceerd om vertragingen in HTA-procedures te voorkomen.

3 Referenties

1. Baio G and Heath A. When simple becomes complicated: why Excel should lose its place at the top table. *Glob Reg Health Technol Assess* 2016; 4: e3-e6.
2. Inceri D, Thom H, Baio G, et al. R You Still Using Excel? The Advantages of Modern Software Tools for Health Technology Assessment. *Value Health* 2019; 22: 575-9.
3. Zorginstituut Nederland. Richtlijn voor het uitvoeren van economische evaluaties in de gezondheidszorg. 2016.
4. Alarid-Escudero F, Krijkamp EM, Pechlivanoglou P, et al. A Need for Change! A Coding Framework for Improving Transparency in Decision Modeling. *Pharmacoeconomics* 2019; 37: 1329-39.
5. Wickha H. *testthat: Get Started with Testing*. 2011. Geraadpleegd op January 9, 2022 via https://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf.