

Package: gonovax (via r-universe)

October 12, 2026

Title Deterministic Compartmental Model of Gonorrhoea with Vaccination

Version 0.4.23

Description Model for gonorrhoea vaccination, using odin.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Language en-GB

Imports odin (>= 1.1.0), dde (>= 1.0.1), furrr, progress (>= 1.2.0),
mcstate (>= 0.2.13)

Suggests testthat, knitr, rmarkdown

RoxygenNote 7.3.2

VignetteBuilder knitr

Remotes mrc-ide/odin, mrc-ide/dde, mrc-ide/mcstate

Config/pak/sysreqs cmake make libuv1-dev

Repository <https://mrc-ide.r-universe.dev>

Date/Publication 2025-03-14 12:59:52 UTC

RemoteUrl <https://github.com/mrc-ide/gonovax>

RemoteRef master

RemoteSha 4b02b30855ddd7fbaee03a7ccbf0a3b3f305dd7b

Contents

adjust_baseline	3
aggregate	4
compare	5
compare_baseline	5
compare_baseline_xpvwrh	7
compare_basic	10
create_diagnosis_waning_map	11
create_uptake_map	11

create_uptake_map_xpvwrh	12
create_uptake_map_xpvwrh_trackvt	13
create_uptake_map_xvw	14
create_uptake_map_xvwr	15
create_uptake_map_xvww	16
create_vax_map	17
create_vax_map_branching	18
create_waning_map	18
create_waning_map_branching	19
create_waning_map_trial	20
dbetabinom	20
extract_flows	21
extract_flows_trial	21
extract_flows_xpvwrh	22
gen_erlang_labels	22
gen_erlang_labels_trackvt	23
gen_labels	23
gen_trial_labels	24
gono_params	24
gono_params_trial	25
gonovax_data	25
gonovax_year	26
gonovax_year_as_year	26
initial_params	27
initial_params_trial	27
initial_params_xpvwrh	28
initial_params_xpvwrh_trackvt	29
initial_params_xvw_trial	30
initial_params_xvwrh	30
mcmc	31
mcmc_combine	32
mcmc_thin	32
model	33
model_params	33
model_params_trial	34
model_params_xpvwrh	35
model_params_xpvwrh_trackvt	36
restart_hes	37
restart_params	38
run	38
run_grid	39
run_onevax_xpvwrh	40
run_onevax_xpvwrh_trackvt	43
run_onevax_xvw	45
run_onevax_xvw_trial	47
run_onevax_xvwr	48
run_onevax_xvwrh	50
run_onevax_xvww	51

run_trial	53
run_xpvwrh	53
run_xpvwrh_trackvt	54
set_protection	55
set_protection_trackvt	56
set_strategy	56
stratum_index_xpvwrh	57
stratum_index_xpvwrh_trackvt	58
stratum_index_xvw	58
stratum_index_xvw_trial	59
stratum_index_xvwr	59
stratum_index_xvwrh	60
stratum_index_xvwv	60
transform	61
transform_fixed	61
vax_params_xpvwrh	62
vax_params_xpvwrh_trackvt	64
vax_params_xvw	66
vax_params_xvw_trial	67
vax_params_xvwr	68
vax_params_xvwrh	69
vax_params_xvwv	70
vax_params0	71
Index	73

adjust_baseline	<i>adjust model run in absence of vaccination so diagnoses are spread across vaccine protected and non-vaccine protected strata as if the rate of movement was the same as in an equivalent model run in the presence of vaccination</i>
-----------------	--

Description

adjust model run in absence of vaccination so diagnoses are spread across vaccine protected and non-vaccine protected strata as if the rate of movement was the same as in an equivalent model run in the presence of vaccination

Usage

adjust_baseline(baseline, y)

Arguments

- | | |
|----------|---|
| baseline | A model run in the absence of vaccine uptake |
| y | A model run in the presence of vaccine uptake |

Value

An adjusted baseline in which 'a' and 's' diagnoses in the baseline are divided across vaccine protected and non-vaccine protected strata allowing comparison of model runs with of baselines by vaccine-status

aggregate	<i>aggregate model runs by vaccination group x vaccine strata</i>
-----------	---

Description

aggregate model runs by vaccination group x vaccine strata

Usage

```
aggregate(  
  x,  
  what,  
  as_incid = FALSE,  
  stratum = NULL,  
  f = identity,  
  stochastic = FALSE,  
  ...  
)
```

Arguments

x	a transformed model run output
what	a character string of the cumulative trajectory to transform
as_incid	logical specifying whether to convert to incidence
stratum	an integer denoting the stratum to aggregate over, if NULL aggregation is over all strata
f	function to apply across parameter sets (eg. mean / median)
stochastic	indicates if function is being used on output of stochastic or deterministic version of the trial model, only matters if as_incid = TRUE
...	named arguments to pass to f

Value

a transformed time series / array

compare	<i>Calculate the log likelihood of the data given the parameters diagnoses and attendances lhoods are negative binomial p_symp lhood is betabinomial</i>
---------	--

Description

Calculate the log likelihood of the data given the parameters diagnoses and attendances lhoods are negative binomial p_symp lhood is betabinomial

Usage

```
compare(pars, transform)
```

Arguments

pars	A named vector of parameters
transform	the transform function to use in the comparison

Value

a single log likelihood

compare_baseline	<i>compare model runs with vaccination to a baseline runs</i>
------------------	---

Description

compare model runs with vaccination to a baseline runs

Usage

```
compare_baseline(
  y,
  baseline,
  uptake_first_dose,
  uptake_second_dose,
  cost_params,
  disc_rate
)
```

Arguments

y	list of model runs, e.g. created by 'run_onevax_xvww', each list entry refers to a different parameter set
baseline	list of baseline runs e.g. created by 'run_onevax_xvww'. Should be same length as 'y'
uptake_first_dose	numeric (0-1) either length 1 or same as 'y'
uptake_second_dose	numeric (0-1) either length 1 or same as 'y'
cost_params	list of cost effectiveness parameters, containing entries 'qaly_loss_per_diag_s', 'unit_cost_manage_symptomatic', 'unit_cost_manage_asymptomatic', 'unit_cost_screen_uninfected'. Each entry length 1 or same as 'y'
disc_rate	discount rate for cost-effectiveness output per annum, default = 0

Value

A list of matrices with rows = time point and columns = parameter set list entries are: 'cum_treated' = cumulative number treated in model run 'cum_vaccinated' = cumulative vaccinated treated in model run 'diag_a' = annual number of asymptomatic diagnoses in model run 'diag_s' = annual number of symptomatic diagnoses in model run 'treated' = annual number treated in model run 'screened' = annual number screened in model run 'vaccinated' = annual number vaccinated in model run (includes those getting vbe and boosters) 'vbe' = annual number vaccinated before entry in model run 'revaccinated' = annual number receiving booster in model run 'offered_primary' = annual number offered primary vaccination (does not include hesitant or vbe) 'inc_cum_treated' = cumulative number treated compared to baseline 'inc_cum_vaccinated' = cumulative number vaccinated compared to baseline 'inc_diag_a' = annual number of asymptomatic diagnoses compared to baseline 'inc_diag_s' = annual number of symptomatic diagnoses compared to baseline 'inc_treated' = annual number treated compared to baseline 'inc_screened' = annual number screened compared to baseline 'inc_vaccinated' = annual number vaccinated compared to baseline (includes those getting vbe and boosters) 'inc_vbe' = annual number vaccinated before entry compared to baseline 'inc_revaccinated' = annual number receiving booster compared to baseline 'inc_cum_revaccinated' = cumulative number revaccinated by receiving booster compared to baseline 'inc_offered_primary' = annual number offered primary vaccination compared to baseline (does not include hesitant or vbe) 'inc_primary' = annual number receiving primary vaccination compared to baseline (does not include vbe) 'inc_cum_primary' = cumulative number of individuals receiving primary vaccination compared to baseline (does not include vbe) 'inc_doses' = number of doses compared to baseline (per year). Assumes primary vaccination uses 2 doses, booster uses 1 dose. 'inc_cum_doses' = cumulative number of doses compared to baseline. Assumes primary vaccination uses 2 doses, booster uses 1 dose. 'inc_primary_doses' = annual number of primary doses administered compared to baseline, where 2 doses are needed for full protection. 'inc_cum_primary_doses' = cumulative number of primary doses administered compared to baseline, where 2 doses are needed for full protection. 'inc_booster_doses' = annual number of booster doses administered to waned individuals for revaccination compared to baseline, where 1 dose is needed for full protection. 'inc_cum_booster_doses' = cumulative number of booster doses administered to waned individuals for revaccination compared to baseline, where 1 dose is needed for full protection. 'inc_vbe_doses' = annual number of vaccinations given before entry, where 2

doses give full protection. 'inc_cum_vbe_doses' = cumulative number of vaccinations given before entry, where 2 doses give full protection. 'cases_averted_per_dose' = cumulative number of cases (i.e. diagnoses) averted per dose of vaccine 'cases_averted_per_dose_pv' = present value of cases_averted_per_dose (i.e. sum of annual numbers discounted at rate 'disc_rate' to time 0) 'inc_doses_wasted' annual number of first doses given that are not followed by a second dose, compared to baseline 'inc_cum_doses_wasted' cumulative number of first doses given that are not followed by a second dose, compared to baseline 'pv_inc_doses' = present value of number of doses compared to baseline (i.e. sum of annual numbers discounted at rate 'disc_rate' to time 0) 'pv_red_net_cost' = present value of the reduction in net costs compared to baseline (i.e. sum of annual reduction in costs discounted at rate 'disc_rate' to time 0) 'pv_qaly_gain' = present value of the QALYs gained due to vaccination compared to baseline (i.e. QALYs gained each year discounted at rate 'disc_rate' to time 0, and summed) 'cet_20k' = cost effectiveness threshold (i.e. £ value per dose at which vaccination becomes cost-effective) calculated using £20,000 / QALY. $(pv_red_net_cost + pv_qaly_gain * £20,000) / pv_inc_doses$ 'cet_30k' = cost effectiveness threshold (i.e. £ value per dose at which vaccination becomes cost-effective) calculated using £30,000 / QALY $(pv_red_net_cost + pv_qaly_gain * £30,000) / pv_inc_doses$ 'inc_costs_9' = present value of incremental costs assuming £9 / dose. Incremental costs are calculated as: $pv_inc_doses * £9 - pv_red_net_cost$ 'inc_costs_18' = present value of incremental costs assuming £18 / dose. Incremental costs are calculated as: $pv_inc_doses * £18 - pv_red_net_cost$ 'inc_costs_50' = present value of incremental costs assuming £50 / dose. Incremental costs are calculated as: $pv_inc_doses * £50 - pv_red_net_cost$ 'inc_costs_85' = present value of incremental costs assuming £85 / dose. Incremental costs are calculated as: $pv_inc_doses * £85 - pv_red_net_cost$

compare_baseline_xpvwrh

compare model runs with vaccination to a baseline run for the branching XPVWRH model where both partial and full vaccination have a given level of efficacy

Description

compare model runs with vaccination to a baseline run for the branching XPVWRH model where both partial and full vaccination have a given level of efficacy

Usage

```
compare_baseline_xpvwrh(
  y,
  baseline,
  uptake_first_dose,
  uptake_second_dose,
  cost_params,
  disc_rate,
  vea,
  vea_p
)
```

Arguments

y	list of model runs, e.g. created by 'run_onevax_xpvwrh', each list entry refers to a different parameter set
baseline	list of baseline runs e.g. created by 'run_onevax_xpvwrh'. Should be same length as 'y'
uptake_first_dose	numeric (0-1) either length 1 or same as 'y'
uptake_second_dose	numeric (0-1) either length 1 or same as 'y'
cost_params	list of cost effectiveness parameters, containing entries 'qaly_loss_per_diag_s', 'unit_cost_manage_symptomatic', 'unit_cost_manage_asymptomatic', 'unit_cost_screen_uninfected'. Each entry length 1 or same as 'y'
disc_rate	discount rate for cost-effectiveness output per annum, default = 0
vea	= scalar indicating efficacy of full vaccination against aquisition (between 0-1)
vea_p	scalar indicating efficacy of partial vaccination against acquisition (between 0-1)

Value

A list of matrices with rows = time point and columns = parameter set list entries are: 'cum_treated' = cumulative number treated in model run 'cum_vaccinated' = cumulative vaccinated treated in model run 'diag_a' = annual number of asymptomatic diagnoses in model run 'diag_s' = annual number of symptomatic diagnoses in model run 'treated' = annual number treated in model run 'screened' = annual number screened in model run 'vaccinated' = annual number vaccinated in model run (includes those getting vbe and boosters) 'vbe' = annual number vaccinated before entry in model run 'revaccinated' = annual number receiving booster in model run 'inc_cum_treated' = cumulative number treated compared to baseline 'pv_cases_averted' = cumulative number treated when discounting applied compared to baseline 'inc_cum_vaccinated' = cumulative number vaccinated compared to baseline 'inc_diag_a' = annual number of asymptomatic diagnoses compared to baseline 'inc_diag_s' = annual number of symptomatic diagnoses compared to baseline 'inc_treated' = annual number treated compared to baseline 'inc_screened' = annual number screened compared to baseline 'inc_vaccinated' = annual number vaccinated compared to baseline (includes those getting vbe and boosters) 'inc_vbe' = annual number vaccinated before entry compared to baseline 'inc_revaccinated' = annual number receiving booster compared to baseline 'inc_cum_revaccinated' = cumulative number revaccinated by receiving booster compared to baseline 'inc_primary' = annual number receiving primary vaccination (full or partial) compared to baseline (does not include vbe) 'inc_cum_primary' = cumulative number of individuals receiving primary vaccination (full or partial) compared to baseline (does not include vbe) 'inc_part_to_full' = annual number of individuals who have already received their 1st dose, who receive their 2nd dose compared to baseline 'inc_cum_part_to_full' = cumulative number of individuals who have already received their 1st dose, who receive their 2nd dose compared to baseline 'inc_doses' = number of doses compared to baseline (per year). Assumes primary vaccination uses 2 doses, booster uses 1 dose. 'inc_cum_doses' = cumulative number of doses compared to baseline. Assumes primary vaccination uses 2 doses, booster uses 1 dose. 'inc_primary_part_doses' = annual number of 1st doses administered as part of primary vaccination which were not followed by a 2nd dose compared to baseline. 'inc_cum_primary_part_doses' = cumulative number of 1st doses administered as part of primary vaccination which were not followed by a 2nd dose compared to baseline. 'inc_primary_full_doses' = annual number 1st and 2nd doses administered together as part

of primary vaccination compared to baseline. 'inc_cum_primary_full_doses' = cumulative number of 1st and 2nd doses administered together as part of primary vaccination compared to baseline. 'inc_primary_total_doses' = annual number of primary doses administered compared to baseline. Includes doses contributing to partial and full vaccination 'inc_cum_primary_total_doses' = cumulative number of primary doses administered compared to baseline. Includes doses contributing to partial and full vaccination. 'inc_booster_doses' = annual number of booster doses administered to waned individuals for revaccination compared to baseline, where 1 dose is needed for full protection. 'inc_cum_booster_doses' = cumulative number of booster doses administered to waned individuals for revaccination compared to baseline, where 1 dose is needed for full protection. 'inc_vbe_doses' = annual number of vaccinations given before entry, where 2 doses give full protection. 'inc_cum_vbe_doses' = cumulative number of vaccinations given before entry, where 2 doses give full protection. 'cases_averted_per_dose' = cumulative number of cases (i.e. diagnoses) averted per dose of vaccine 'cases_averted_per_dose_pv' = present value of cases_averted_per_dose (i.e. sum of annual numbers discounted at rate 'disc_rate' to time 0) 'pv_inc_doses' = present value of number of doses compared to baseline (i.e. sum of annual numbers discounted at rate 'disc_rate' to time 0) 'pv_red_net_cost' = present value of the reduction in net costs compared to baseline (i.e. sum of annual reduction in costs discounted at rate 'disc_rate' to time 0) 'pv_qaly_gain' = present value of the QALYs gained due to vaccination compared to baseline (i.e. QALYs gained each year discounted at rate 'disc_rate' to time 0, and summed) 'cet_20k' = cost effectiveness threshold (i.e. £ value per dose at which vaccination becomes cost-effective) calculated using £20,000 / QALY. $(pv_red_net_cost + pv_qaly_gain * £20,000) / pv_inc_doses$ 'cet_30k' = cost effectiveness threshold (i.e. £ value per dose at which vaccination becomes cost-effective) calculated using £30,000 / QALY $(pv_red_net_cost + pv_qaly_gain * £30,000) / pv_inc_doses$ 'inc_costs_9' = present value of incremental costs assuming £9 / dose. Incremental costs are calculated as: $pv_inc_doses * £9 - pv_red_net_cost$ 'inc_costs_18' = present value of incremental costs assuming £18 / dose. Incremental costs are calculated as: $pv_inc_doses * £18 - pv_red_net_cost$ 'inc_costs_35' = present value of incremental costs assuming £35 / dose. Incremental costs are calculated as: $pv_inc_doses * £35 - pv_red_net_cost$ 'inc_costs_50' = present value of incremental costs assuming £50 / dose. Incremental costs are calculated as: $pv_inc_doses * £50 - pv_red_net_cost$ 'inc_costs_70' = present value of incremental costs assuming £70 / dose. Incremental costs are calculated as: $pv_inc_doses * £70 - pv_red_net_cost$ 'inc_costs_85' = present value of incremental costs assuming £85 / dose. Incremental costs are calculated as: $pv_inc_doses * £85 - pv_red_net_cost$ 'vacprotec_full' = number of fully vaccine protected individuals in the population at a given timepoint 'vacprotec_part' = number of partially vaccine protected individuals in the population at a given timepoint 'vacprotec_total' = total number of vaccine protected individuals in the population (both partially and fully vaccinated) at a given timepoint 'vacprotec_full_prop' = proportion of the population experiencing full vaccine protection at a given timepoint 'vacprotec_part_prop' = proportion of the population experiencing partial vaccine protection at a given timepoint 'vacprotec_total_prop' = total proportion of the population experiencing some form of vaccine protection at a given timepoint 'level_vacprotec' = the level of vaccine protection in the population at a given timepoint. Given as the sum of the products of the number of partially and fully vaccinated individuals and the one dose and two dose vaccine efficacies respectively. 'inc_diag_a_xwh' = annual number of asymptomatic diagnoses in non- vaccine protected strata compared to an adjusted baseline 'inc_diag_s_xwh' = annual number of symptomatic diagnoses in non- vaccine protected strata compared to an adjusted baseline 'inc_diag_t_xwh' = annual number of total diagnoses in non-vaccine protected strata compared to an adjusted baseline 'inc_diag_a_pvr' = annual number of asymptomatic diagnoses in vaccine protected strata compared to an adjusted baseline 'inc_diag_s_pvr' = annual number of symptomatic diagnoses in vaccine protected strata compared to an adjusted baseline

'inc_diag_t_pvr' = annual number of total diagnoses in vaccine protected strata compared to an adjusted baseline
 'inc_cum_diag_a_xwh' = cumulative number of asymptomatic diagnoses in non-vaccine protected strata compared to an adjusted baseline
 'inc_cum_diag_s_xwh' = cumulative number of symptomatic diagnoses in non-vaccine protected strata compared to an adjusted baseline
 'inc_cum_diag_t_xwh' = cumulative number of total diagnoses in non-vaccine protected strata compared to an adjusted baseline
 'inc_cum_diag_a_pvr' = cumulative number of asymptomatic diagnoses in vaccine protected strata compared to an adjusted baseline
 'inc_cum_diag_s_pvr' = cumulative number of symptomatic diagnoses in vaccine protected strata compared to an adjusted baseline
 'inc_cum_diag_t_pvr' = cumulative number of total diagnoses in vaccine protected strata compared to an adjusted baseline
 'percentage_cases_avert' = percentage of cases avoided in each time-point compared to a baseline of no vaccination when cases are counted as successful treatment
 'cum_percentage_cases_avert' = percentage of cases avoided cumulatively across all time-points compared to a baseline of no vaccination when cases are counted as successful treatment
 'percentage_diag_avert' = percentage of cases avoided in each time-point compared to a baseline of no vaccination when cases are counted as the sum of all diagnoses
 'cum_percentage_diags_avert' = percentage of cases avoided cumulatively across all timepoints compared to a baseline of no vaccination when cases are counted as the sum of all diagnoses
 'inc_diag_t' = total number of diagnoses in each time point
 'inc_cum_diag_t' = cumulative total number of diagnoses over time
 'inc_cum_diag_a' = cumulative number of asymptomatic diagnoses over time
 'inc_cum_diag_s' = cumulative number of symptomatic diagnoses over time
 'pv_num_vaccinations' = present value cumulative number of vaccinations

 compare_basic

Calculate the log likelihood of the data given the parameters

Description

Calculate the log likelihood of the data given the parameters

Usage

```
compare_basic(pars)
```

Arguments

pars A named vector of parameters

Value

a single log likelihood

```
create_diagnosis_waning_map
```

Create mapping for movement between strata due to diagnosis waning

Description

Create mapping for movement between strata due to diagnosis waning

Usage

```
create_diagnosis_waning_map(n_vax, z, n_diag_rec = 1)
```

Arguments

n_vax	Integer in (0, 5) denoting total number of strata
z	Scalar denoting rate of waning diagnosis
n_diag_rec	integer for the number of diagnosis history substrata

Value

an array of the mapping

create_uptake_map	<i>Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.</i>
-------------------	--

Description

Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.

Usage

```
create_uptake_map(
  n_group,
  n_vax,
  primary_uptake,
  booster_uptake,
  i_eligible,
  i_v,
  screening_or_diagnosis
)
```

Arguments

n_group	scalar indicating number of activity groups
n_vax	scalar indicating number the number of stratum in the model
primary_uptake	proportion of the unvaccinated population who accept primary vaccination
booster_uptake	proportion of the formerly fully vaccinated, waned population who accept a booster vaccination dose
i_eligible	vector of indices of stratum which are eligible for vaccination, of same length as the number of paths from unvaccinated to vaccinated
i_v	vector of indices of stratum which are vaccinated and experience protection
screening_or_diagnosis	string indicating screening or diagnosis

Value

an array of the uptakes with dimensions n_group x n_vax x n_vax

create_uptake_map_xpvwrh

Creates uptake mapping for the branching XPVWRH model where individuals can move from unvaccinated (X) to vaccinated (V) or partially vaccinated (P) as well as revaccinated from waned (W) to (R) and, and partially vaccinated (P) to fully vaccianted (V). The former reflects the specific indices which are chosen for assigning uptakes.

Description

Creates uptake mapping for the branching XPVWRH model where individuals can move from unvaccinated (X) to vaccinated (V) or partially vaccinated (P) as well as revaccinated from waned (W) to (R) and, and partially vaccinated (P) to fully vaccianted (V). The former reflects the specific indices which are chosen for assigning uptakes.

Usage

```
create_uptake_map_xpvwrh(
  array,
  r1,
  r2,
  r2_p,
  booster_uptake,
  idx,
  n_diag_rec = 1,
  screening_or_diagnosis
)
```

Arguments

array	a vaccine map array of dimensions n_group by n_vax by n_vax generated through create_vax_map_branching()
r1	proportion of population offered vaccine only accepting the first dose, becoming partially vaccinated
r2	proportion of the population who accepted the first dose of the vaccine who go on to accept the second dose, becoming fully vaccinated
r2_p	proportion of partially vaccinated individuals who receive a second dose when returning to the clinic due to screening or illness
booster_uptake	proportion of the formerly fully vaccinated, waned population who accept a booster vaccination dose
idx	list containing indices of all X, P, V, W, R & H strata and n_vax through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
screening_or_diagnosis	string indicating screening or diagnosis

Value

an array of the uptakes of same dimensions

create_uptake_map_xpvwrh_trackvt

Creates uptake mapping for the branching XPVWRH model where individuals can move from unvaccinated (X) to vaccinated (V) or partially vaccinated (P) as well as revaccinated from waned (W) to (R) and, and partially vaccinated (P) to fully vaccinated (V). The former reflects the specific indices which are chosen for assigning uptakes.

Description

Creates uptake mapping for the branching XPVWRH model where individuals can move from unvaccinated (X) to vaccinated (V) or partially vaccinated (P) as well as revaccinated from waned (W) to (R) and, and partially vaccinated (P) to fully vaccinated (V). The former reflects the specific indices which are chosen for assigning uptakes.

Usage

```
create_uptake_map_xpvwrh_trackvt(
  array,
  r1,
  r2,
  r1_p,
  r2_p,
```

```

    booster_uptake,
    idx,
    n_diag_rec = 1,
    screening_or_diagnosis
)

```

Arguments

array	a vaccine map array of dimensions n_group by n_vax by n_vax generated through create_vax_map_branching()
r1	proportion of population offered vaccine only accepting the first dose, becoming partially vaccinated
r2	proportion of the population who accepted the first dose of the vaccine who go on to accept the second dose, becoming fully vaccinated
r1_p	proportion of partially vaccinated individuals who receive a ≥ 1 dose when returning to the clinic due to screening or illness
r2_p	proportion of partially vaccinated individuals who receive a second dose when returning to the clinic due to screening or illness
booster_uptake	proportion of the formerly fully vaccinated, waned population who accept a booster vaccination dose
idx	list containing indices of all X, P, V, W, R & H strata and n_vax through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
screening_or_diagnosis	string indicating screening or diagnosis

Value

an array of the uptakes of same dimensions

create_uptake_map_xvw *Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.*

Description

Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.

Usage

```
create_uptake_map_xvw(
  n_group,
  n_vax,
  uptake,
  idx,
  n_diag_rec = 1,
  screening_or_diagnosis
)
```

Arguments

n_group	scalar indicating number of activity groups
n_vax	scalar indicating number the number of stratum in the model
uptake	proportion of the unvaccinated population who accept vaccination
idx	list containing indices of all X, V, W strata and n_vax through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
screening_or_diagnosis	string indicating screening or diagnosis

Value

an array of the uptakes with dimensions n_group x n_vax x n_vax

```
create_uptake_map_xvwr
```

Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.

Description

Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.

Usage

```
create_uptake_map_xvwr(
  n_group,
  n_vax,
  primary_uptake,
  booster_uptake,
  idx,
  n_diag_rec = 1,
  screening_or_diagnosis
)
```

Arguments

n_group	scalar indicating number of activity groups
n_vax	scalar indicating number the number of stratum in the model
primary_uptake	proportion of the unvaccinated population who accept primary vaccination
booster_uptake	proportion of the formerly fully vaccinated, waned population who accept a booster vaccination dose
idx	list containing indices of all X, V, W, R strata and n_vax through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
screening_or_diagnosis	string indicating screening or diagnosis

Value

an array of the uptakes with dimensions n_group x n_vax x n_vax

create_uptake_map_xvwv

Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.

Description

Creates uptake mapping array with dimensions n_group x n_vax x n_vax and assigns the relevant primary uptake and booster uptake values defined by the user.

Usage

```
create_uptake_map_xvwv(
  n_group,
  n_vax,
  primary_uptake,
  booster_uptake,
  idx,
  n_diag_rec = 1,
  screening_or_diagnosis
)
```

Arguments

n_group	scalar indicating number of activity groups
n_vax	scalar indicating number the number of stratum in the model
primary_uptake	proportion of the unvaccinated population who accept primary vaccination

booster_uptake	proportion of the formerly fully vaccinated, waned population who accept a booster vaccination dose
idx	list containing indices of all X, V, W strata and n_vax through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
screening_or_diagnosis	string indicating screening or diagnosis

Value

an array of the uptakes with dimensions n_group x n_vax x n_vax

create_vax_map	<i>Create mapping for movement between strata due to vaccination</i>
----------------	--

Description

Create mapping for movement between strata due to vaccination

Usage

```
create_vax_map(n_vax, v, i_u, i_v)
```

Arguments

n_vax	Integer denoting total number of strata
v	0-1 vector of length two indicating whether activity group should be offered vaccination.
i_u	indices of strata eligible for vaccination
i_v	indices of strata being vaccinated

Value

an array of the mapping

```
create_vax_map_branching
```

Create mapping for movement between strata due to vaccination where vaccination uptake splits off into two types (partial and full) from the naive population (X). Different to create_vax_map as this function specifically maps vbe to V (3) than P(2)

Description

Create mapping for movement between strata due to vaccination where vaccination uptake splits off into two types (partial and full) from the naive population (X). Different to create_vax_map as this function specifically maps vbe to V (3) than P(2)

Usage

```
create_vax_map_branching(n_vax, v, i_e, i_p, set_vbe = FALSE, idx)
```

Arguments

n_vax	Integer denoting total number of strata
v	0-1 vector of length two indicating whether activity group should be offered vaccination.
i_e	indices of strata eligible for vaccination
i_p	indices of strata vaccinated and protected
set_vbe	Boolean which indicates that vaccination is occurring at some level of uptake upon entering the model
idx	list containing indices of all X, P, V, W, R & H strata and n_vax through vaccine-protected strata until that protection has waned

Value

an array of the mapping

```
create_waning_map
```

Create mapping for movement between strata due to vaccine waning

Description

Create mapping for movement between strata due to vaccine waning

Usage

```
create_waning_map(n_vax, i_v, i_w, z, n_diag_rec = 1)
```

Arguments

n_vax	Integer in (0, 5) denoting total number of strata
i_v	indices of strata being vaccinated
i_w	Integer in (0, 5) denoting which stratum receives waned vaccinees
z	Scalar denoting rate of waning
n_diag_rec	integer for the number of diagnosis history substrata

Value

an array of the mapping

create_waning_map_branching

Create mapping for movement between strata due to vaccine waning where waning from the partially vaccinated stratum (P) moves individuals back to a naive unvaccinated state (X), and waning from fully vaccinated stratum (V) moves individuals into a separate waned stratum (W) Note, this structure is specific to xpvwrh

Description

Create mapping for movement between strata due to vaccine waning where waning from the partially vaccinated stratum (P) moves individuals back to a naive unvaccinated state (X), and waning from fully vaccinated stratum (V) moves individuals into a separate waned stratum (W) Note, this structure is specific to xpvwrh

Usage

```
create_waning_map_branching(n_vax, i_v, i_w, z, n_erlang = 1, n_diag_rec = 1)
```

Arguments

n_vax	Integer denoting total number of strata
i_v	indices of strata receiving protection through vaccination
i_w	Scalar in (0, 6) denoting which stratum receives waned vaccinees
z	Scalar denoting rate of waning
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer giving number of diagnosis history strata

Value

an array of the mapping

```
create_waning_map_trial
```

Create mapping for movement between strata due to vaccine waning in a vaccine trial with erlang compartments

Description

Create mapping for movement between strata due to vaccine waning in a vaccine trial with erlang compartments

Usage

```
create_waning_map_trial(n_vax, i_v, i_w, z)
```

Arguments

n_vax	Integer in (0, 5) denoting total number of strata
i_v	indices of strata under vaccination protection
i_w	indicies denoting which stratum receives waned vaccinees
z	Scalar denoting rate of waning

Value

an array of the mapping

```
dbetabinom
```

pdf of a betabinomial parametrised in terms of probability and over-dispersion

Description

pdf of a betabinomial parametrised in terms of probability and over-dispersion

Usage

```
dbetabinom(x, size, prob, rho, log = FALSE)
```

Arguments

x	data
size	integer of sample size
prob	probability of observing a single success
rho	overdispersion parameter, has support [0, 1] with low values being less overdispersion
log	logical indicating whether to return log value

Value

probability of observing x

extract_flows	<i>extract flows used for run_grid</i>
---------------	--

Description

extract flows used for run_grid

Usage

extract_flows(y)

Arguments

y a transformed model run output

Value

cumulative and incident flows

extract_flows_trial	<i>extract flows for the XVW trial model</i>
---------------------	--

Description

extract flows for the XVW trial model

Usage

extract_flows_trial(y)

Arguments

y a transformed model run output

Value

cumulative and incident flows

extract_flows_xpvwrh	<i>extract flows used for run_grid when the branching xpvwrh model has been run</i>
----------------------	---

Description

extract flows used for run_grid when the branching xpvwrh model has been run

Usage

```
extract_flows_xpvwrh(y)
```

Arguments

y	a transformed model run output
---	--------------------------------

Value

cumulative and incident flows

gen_erlang_labels	<i>generates the appropriate strata labels for the number of strata in the model, which depends on the value given to n_erlang</i>
-------------------	--

Description

generates the appropriate strata labels for the number of strata in the model, which depends on the value given to n_erlang

Usage

```
gen_erlang_labels(n_erlang = 1, n_diag_rec = 1)
```

Arguments

n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata

Value

a character vector of length n_vax containing strata labels

gen_erlang_labels_trackvt

generates the appropriate strata labels for the number of strata in the model, which depends on the value given to n_erlang

Description

generates the appropriate strata labels for the number of strata in the model, which depends on the value given to n_erlang

Usage

```
gen_erlang_labels_trackvt(n_erlang = 1, n_diag_rec = 1)
```

Arguments

n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata

Value

a character vector of length n_vax containing strata labels

gen_labels

generates the appropriate strata labels for the number of strata in the model, which depends on the value given to n_erlang and diagnosis history levels desired (n_diag_rec)

Description

generates the appropriate strata labels for the number of strata in the model, which depends on the value given to n_erlang and diagnosis history levels desired (n_diag_rec)

Usage

```
gen_labels(n_erlang = 1, n_diag_rec = 1)
```

Arguments

n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer giving the number of levels of diagnosis history for each X, V(*n_erlang), and W stratum

Value

a character vector of length `n_vax` containing strata labels

<code>gen_trial_labels</code>	<i>generates the appropriate strata labels for the number of strata in the model, which depends on the value given to <code>n_erlang</code> and diagnosis history levels desired (<code>n_diag_rec</code>)</i>
-------------------------------	--

Description

generates the appropriate strata labels for the number of strata in the model, which depends on the value given to `n_erlang` and diagnosis history levels desired (`n_diag_rec`)

Usage

```
gen_trial_labels(n_erlang = 1, n_diag_rec = 1)
```

Arguments

- | | |
|-------------------------|--|
| <code>n_erlang</code> | integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned |
| <code>n_diag_rec</code> | integer giving the number of levels of diagnosis history for each X, V(* <code>n_erlang</code>), and W stratum |

Value

a character vector of length `n_vax` containing strata labels

<code>gono_params</code>	<i>Posterior parameters of gonorrhoea natural history</i>
--------------------------	---

Description

Posterior parameters of gonorrhoea natural history

Usage

```
gono_params(n = NULL)
```

Arguments

- | | |
|----------------|---|
| <code>n</code> | an integer vector (or value) containing the indices of the required parameter sets (1:982). If ' <code>n = NULL</code> ' the full parameter set is returned |
|----------------|---|

Value

A list of parameters

gono_params_trial	<i>Posterior parameters of gonorrhoea natural history</i>
-------------------	---

Description

Posterior parameters of gonorrhoea natural history

Usage

```
gono_params_trial(n = NULL)
```

Arguments

n an integer vector (or value) containing the indices of the required parameter sets (1:982). If ‘n = NULL’ the full parameter set is returned

Value

A list of parameters

gonovax_data	<i>Get gonovax data</i>
--------------	-------------------------

Description

Get annual time series of GUMCAD diagnoses and testing figures, and annual proportion symptomatic from GRASP

Usage

```
gonovax_data()
```

Value

A data.frame containing the gonovax year (i.e. year after 2007), the year, the total diagnoses in MSM, the total tests in MSM and the source of the data, the proportion of symptomatic diagnoses

gonovax_year	<i>Convert a year into the number of years after 2009</i>
--------------	---

Description

Convert a year into the number of years after 2009

Usage

```
gonovax_year(year)
```

Arguments

year	an integer year
------	-----------------

Value

An integer, being the number of years after 2009

Examples

```
gonovax_year(2019)
gonovax_year(c(2018, 2019))
```

gonovax_year_as_year	<i>Convert a gonovax year into calendar years</i>
----------------------	---

Description

Convert a gonovax year into calendar years

Usage

```
gonovax_year_as_year(gonovax_year)
```

Arguments

gonovax_year	an integer
--------------	------------

Value

An integer, being the calendar year

Examples

```
gonovax_year_as_year(3)
```

initial_params	<i>Initial conditions for the model</i>
----------------	---

Description

Create initial conditions for the model

Create initial conditions for the model

Usage

```
initial_params_xvw(pars, coverage = 0)
```

```
initial_params(pars, n_vax = 1, coverage = 1)
```

Arguments

pars	A parameter list containing 'N0', 'q', 'prev_Asl' and 'prev_Ash' elements.
coverage	a vector of length 'n_vax' that sums to 1 denoting the initial proportion in each vaccine stratum
n_vax	an integer indicating the number of vaccine compartments

Value

A list of initial conditions

A list of initial model states

initial_params_trial	<i>Initial conditions for the model trial where the entire cohort is in the high sexual activity group.</i>
----------------------	---

Description

Create initial conditions for the model trial

Usage

```
initial_params_trial(pars, n_vax = 1, p_v = 1, n_diag_rec = n_diag_rec)
```

Arguments

pars	A parameter list containing 'N0', and 'q' elements.
n_vax	an integer indicating the number of vaccine compartments
p_v	a vector of length 'n_vax' that sums to 1 denoting the proportion in each vaccine stratum
n_diag_rec	integer giving the number of each X, V(erlang), and W stratum, allowing tracking of diagnosis history. e.g for a n_diag_rec = 2 and erlang = 1, there will be X.I, X.II, V1.I, V1.II, W.I, W.II strata. Where '.I' corresponds to never-diagnosed individuals and '.II' is for individuals diagnosed at least once.

Value

A list of initial model states

initial_params_xpvwrh *Initial conditions for the model*

Description

Create initial conditions for the model

Usage

```
initial_params_xpvwrh(
  pars,
  coverage_p = 0,
  coverage_v = 0,
  hes = 0,
  t = FALSE,
  n_erlang = 1,
  n_diag_rec = 1
)
```

Arguments

pars	A parameter list containing 'N0', 'q', 'prev_Asl' and 'prev_Ash' elements.
coverage_p	partial (one-dose) vaccine coverage of the population already present (as a proportion)
coverage_v	two-dose vaccine coverage of the population already present (as a proportion)
hes	proportion of population vaccine hesitant
t	number of years, only use when using function outside of run_onevax_xpvwrh() to generate initial conditions for tests
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	number of diagnosis history strata

Value

A list of initial conditions

```
initial_params_xpvwrh_trackvt
```

Initial conditions for the model

Description

Create initial conditions for the model

Usage

```
initial_params_xpvwrh_trackvt(
  pars,
  coverage_p = 0,
  coverage_v = 0,
  hes = 0,
  t = FALSE,
  n_erlang = 1,
  n_diag_rec = 1
)
```

Arguments

<code>pars</code>	A parameter list containing ‘N0’, ‘q’, ‘prev_Asl’ and ‘prev_Ash’ elements.
<code>coverage_p</code>	partial (one-dose) vaccine coverage of the population already present (as a proportion)
<code>coverage_v</code>	two-dose vaccine coverage of the population already present (as a proportion)
<code>hes</code>	proportion of population vaccine hesitant
<code>t</code>	number of years, only use when using function outside of <code>run_onevax_xpvwrh()</code> to generate initial conditions for tests
<code>n_erlang</code>	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
<code>n_diag_rec</code>	number of diagnosis history strata

Value

A list of initial conditions

```
initial_params_xvw_trial
```

Initial conditions for the model in a vaccine trial

Description

Create initial conditions for the model in a vaccine trial

Usage

```
initial_params_xvw_trial(pars, p_v = 0.5, n_erlang = 1, n_diag_rec = 1)
```

Arguments

pars	A parameter list containing 'N0', and 'q' elements.
p_v	scalar giving proportion of the trial cohort vaccinated
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer giving the number of each X, V(erlang), and W stratum, allowing tracking of diagnosis history. e.g for a n_diag_rec = 2 and erlang = 1, there will be X.I, X.II, V1.I, V1.II, W.I, W.II strata. Where '.I' corresponds to never-diagnosed individuals and '.II' is for individuals diagnosed at least once.

Value

A list of initial conditions.

```
initial_params_xvwrh    Initial conditions for the model
```

Description

Create initial conditions for the model

Usage

```
initial_params_xvwrh(pars, coverage = 0, hes = 0, n_diag_rec = 1)
```

Arguments

pars	A parameter list containing 'N0', 'q', 'prev_Asl' and 'prev_Ash' elements.
coverage	a vector of length 'n_vax' that sums to 1 denoting the initial proportion in each vaccine stratum
hes	proportion of population vaccine hesitant
n_diag_rec	integer for the number of diagnosis history substrata

Value

A list of initial conditions

mcmc

Run a mcmc sampler

Description

Run a mcmc sampler

Usage

```
mcmc(pars, n_steps, compare = NULL, progress = FALSE, n_chains = 1)
```

Arguments

pars	A ['mcmc_parameters'] object containing information about parameters (parameter ranges, priors, proposal kernel, observation functions).
n_steps	Number of MCMC steps to run
compare	likelihood function to compare data to epidemic trajectory should return a single value representing the log-likelihood. Default is compare_basic()
progress	Logical, indicating if a progress bar should be displayed, using ['progress::progress_bar'].
n_chains	Optional integer, indicating the number of chains to run. If more than one then we run a series of chains and merge them with [mcmc_combine()]. Chains are run in series, with the same model.

Details

This is a basic Metropolis-Hastings MCMC sampler. The 'model' is run with a set of parameters to evaluate the likelihood. A new set of parameters is proposed, and these likelihoods are compared, jumping with probability equal to their ratio. This is repeated for 'n_steps' proposals.

This function is adapted from the 'pmcmc' in the mcstate package <https://github.com/mrc-ide/mcstate>

Value

A 'gonovax_mcmc' object containing 'pars' (sampled parameters) and 'probabilities' (log prior, log likelihood and log posterior values for these probabilities).

mcmc_combine	<i>Combine mcmc samples</i>
--------------	-----------------------------

Description

all functions in mcmc-tools.R are adapted from pmcmc-tools.R functions in github.com/mrc-ide/mcstate
 Combine multiple [mcmc()] samples into one object

Usage

```
mcmc_combine(..., samples = list(...))
```

Arguments

...	Arguments representing [mcmc()] sample, i.e., 'gonovax_mcmc' objects. Alternatively, pass a list as the argument 'samples'. Names are ignored.
samples	A list of 'gonovax_mcmc' objects. This is often more convenient for programming against than '...'

mcmc_thin	<i>Thin a mcmc chain</i>
-----------	--------------------------

Description

Thin results of running [mcmc()]. 'mcmc_thin' takes every 'thin'th sample, while 'mcmc_sample' randomly selects a total of 'n_sample' samples.

Usage

```
mcmc_thin(object, burnin = NULL, thin = NULL)
```

```
mcmc_sample(object, n_sample, burnin = NULL)
```

Arguments

object	Results of running [mcmc()]
burnin	Optional integer number of iterations to discard as "burn-in". If given then samples '1:burnin' will be excluded from your results. Remember that the first sample represents the starting point of the chain. It is an error if this is not a positive integer or is greater than or equal to the number of samples (i.e., there must be at least one sample remaining after discarding burnin).
thin	Optional integer thinning factor. If given, then every 'thin'th sample is retained (e.g., if 'thin' is 10 then we keep samples 1, 11, 21, ...).
n_sample	The number of samples to draw from 'object' *with replacement*. This means that 'n_sample' can be larger than the total number of samples taken (though it probably should not)

model	<i>Model of gonorrhoea with dual vaccines This is an odin model.</i>
-------	--

Description

Model of gonorrhoea with dual vaccines This is an odin model.

model_params	<i>Parameters for the dualvax model</i>
--------------	---

Description

Parameters for the dualvax model

Usage

```
model_params(  
  gono_params = NULL,  
  demographic_params = NULL,  
  init_params = NULL,  
  vax_params = NULL,  
  n_diag_rec = 1  
)
```

Arguments

gono_params	A dataframe of natural history parameters
demographic_params	A dataframe of demographic parameters
init_params	A list of starting conditions
vax_params	A vector of vaccination params
n_diag_rec	integer for the number of diagnosis history substrata

Value

A list of inputs to the model many of which are fixed and represent data. These correspond largely to ‘user()’ calls within the odin code, though some are also used in processing just before the model is run.

model_params_trial	<i>Parameters for the vaccination trial model</i>
--------------------	---

Description

Parameters for the vaccination trial model

Usage

```
model_params_trial(
  gono_params_trial = NULL,
  demographic_params_trial = NULL,
  initial_params_trial = NULL,
  vax_params = NULL,
  p_v = 0,
  n_erlang = 1,
  N = 6e+05,
  n_diag_rec = 1,
  asymp_recorded = TRUE
)
```

Arguments

gono_params_trial	A dataframe of natural history parameters
demographic_params_trial	A dataframe of demographic parameters
initial_params_trial	A list of starting conditions
vax_params	A vector of vaccination params
p_v	A scalar indicating the percentage of the trial cohort that is vaccinated
n_erlang	scalar giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
N	integer to assign the total number of individuals in the trial (split equally across the two arms)
n_diag_rec	integer giving the number of each X, V(erlang), and W stratum, allowing tracking of diagnosis history. e.g for a n_diag_rec = 2 and erlang = 1, there will be X.I, X.II, V1.I, V1.II, W.I, W.II strata. Where '.I' corresponds to never-diagnosed individuals and '.II' is for individuals diagnosed at least once.
asymp_recorded	logical indicating if the trial screens for and records asymptomatic diagnosis. If FALSE, asymptomatic infected individuals undergoing treatment do not move diagnosis history stratum

Value

A list of inputs to the model many of which are fixed and represent data. These correspond largely to ‘user()’ calls within the odin code, though some are also used in processing just before the model is run.

model_params_xpvwrh	<i>Parameters for the dualvax model</i>
---------------------	---

Description

Parameters for the dualvax model

Usage

```
model_params_xpvwrh(
  gono_params = NULL,
  demographic_params = NULL,
  init_params = NULL,
  vax_params = NULL,
  n_erlang = 1,
  n_diag_rec = 1,
  years_history = 1
)
```

Arguments

gono_params	A dataframe of natural history parameters
demographic_params	A dataframe of demographic parameters
init_params	A list of starting conditions
vax_params	A vector of vaccination params
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
years_history	number of years that diagnosis history is recorded for

Value

A list of inputs to the model many of which are fixed and represent data. These correspond largely to ‘user()’ calls within the odin code, though some are also used in processing just before the model is run.

model_params_xpvwrh_trackvt

Parameters for the xpvwrh model tracking time since vaccination

Description

Parameters for the xpvwrh model tracking time since vaccination

Usage

```
model_params_xpvwrh_trackvt(
  gono_params = NULL,
  demographic_params = NULL,
  init_params = NULL,
  vax_params = NULL,
  n_erlang = 1,
  n_diag_rec = 1,
  years_history = 1
)
```

Arguments

gono_params	A dataframe of natural history parameters
demographic_params	A dataframe of demographic parameters
init_params	A list of starting conditions
vax_params	A vector of vaccination params
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
years_history	number of years that diagnosis history is recorded for

Value

A list of inputs to the model many of which are fixed and represent data. These correspond largely to ‘user()’ calls within the odin code, though some are also used in processing just before the model is run.

restart_hes	<i>uses XPVWRH model run in the absence of vaccination or hesitancy. Saves down the number of individuals in each compartment, and moves a given proportion (hes) of them from the X to the H strata to generate new initial conditions in the presence of hesitancy.</i>
-------------	---

Description

uses XPVWRH model run in the absence of vaccination or hesitancy. Saves down the number of individuals in each compartment, and moves a given proportion (hes) of them from the X to the H strata to generate new initial conditions in the presence of hesitancy.

Usage

```
restart_hes(
  y,
  n_vax = 6,
  hes = 0,
  n_erlang = 1,
  n_diag_rec = 1,
  branching = FALSE
)
```

Arguments

y	a transformed model run output
n_vax	an integer indicating the number of vaccine compartments, consistent with the input
hes	proportion of population vaccine hesitant
n_erlang	integer giving the number of transitions that need to be made
n_diag_rec	integer for the number of diagnosis history substrata
branching	boolean to denote if xpvwrh branching model in use

Value

A list of initial conditions to restart a model with n_vax vaccination levels, and a populated hesitant stratum in the given proportion 'hes'

restart_params	<i>Create initial conditions to start the model from the end of a run</i>
----------------	---

Description

Create initial conditions based on previous model run

Usage

```
restart_params(y, n_vax = NULL)
```

Arguments

y	a transformed model run output
n_vax	an integer indicating the number of vaccine compartments, consistent with the input

Value

A list of initial conditions to restart a model with n_vax vaccination levels

run	<i>Run odin model of gonorrhoea with or without vaccination</i>
-----	---

Description

Run odin model of gonorrhoea with or without vaccination

Usage

```
run(
  tt,
  gono_params,
  demographic_params = NULL,
  init_params = NULL,
  vax_params = NULL,
  n_diag_rec = 1,
  transform = TRUE
)
```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	a data frame of parameters
demographic_params	A dataframe of demographic parameters
init_params	A list of starting conditions
vax_params	A vector of vaccination params
n_diag_rec	integer for the number of diagnosis history substrata
transform	= TRUE

run_grid	<i>run model from equilibrium with single vaccine at the input efficacy / duration grid locations for n parameter sets</i>
----------	--

Description

run model from equilibrium with single vaccine at the input efficacy / duration grid locations for n parameter sets

Usage

```
run_grid(
  gono_params,
  init_params,
  cost_params,
  baseline,
  model,
  eff,
  dur,
  vbe = 0,
  strategy = NULL,
  uptake_total = 0,
  uptake_second_dose = uptake_total,
  t_stop = 99,
  full_output = FALSE,
  disc_rate = 0
)
```

Arguments

gono_params	gono params
init_params	initial state parameters
cost_params	cost effectiveness parameters

baseline	optional input of a baseline to compare to, must be a gonovax_grid object if supplied
model	a gonovax run function, by default 'run_onevax'
eff	numeric vector (between 0-1) of vaccine efficacy against acquisition
dur	numeric vector of duration in years of the vaccine
vbe	single numeric indicating (between 0-1), default = 0
strategy	default is null, no vaccination 'VoD': vaccination on diagnosis, 'VoA': vaccination on attendance, 'VoD(L)+VoA(H)': targeted vaccination (i.e. all diagnosed plus group H on screening)
uptake_total	numeric (0-1) of strategy uptake, default = 0
uptake_second_dose	numeric (0-1) of strategy uptake, default = uptake_total
t_stop	time at which vaccination should stop (years), default = 99
full_output	logical indicating whether full results should be output
disc_rate	discount rate for cost-effectiveness output per annum, default = 0

Value

A 'gonovax_grid' object

run_onevax_xpvwrh	<i>run model with a two-dose vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum, those with waned partial vaccines return to the unvaccinated stratum (X) and considered immunologically naive. A user defined proportion of the population is vaccine hesitant and is never vaccinated. Full acciantion (V) with 2 doses gives maximum protection whereas partial vaccination with 1 dose (P) gives less. Individuals can get 2 doses either by committing in X or upon visiting a clinic in P.</i>
-------------------	---

Description

run model with a two-dose vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum, those with waned partial vaccines return to the unvaccinated stratum (X) and considered immunologically naive. A user defined proportion of the population is vaccine hesitant and is never vaccinated. Full acciantion (V) with 2 doses gives maximum protection whereas partial vaccination with 1 dose (P) gives less. Individuals can get 2 doses either by committing in X or upon visiting a clinic in P.

Usage

```

run_onevax_xpvwrh(
  tt,
  gono_params,
  init_params = NULL,
  dur_v = 1000,
  dur_p = dur_v,
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  dur_revax = dur_v,
  vea_revax = vea,
  vei_revax = vei,
  ved_revax = ved,
  ves_revax = ves,
  vea_p = vea,
  vei_p = vei,
  ved_p = ved,
  ves_p = ves,
  vbe = 0,
  r1 = 0,
  r2 = 0,
  r2_p = 0,
  booster_uptake = (r1 * r2),
  strategy = NULL,
  t_stop = 99,
  hes = 0,
  n_erlang = 1,
  n_diag_rec = 1,
  years_history = 1
)

```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	list of gono params
init_params	A list of starting conditions
dur_v	duration of time spent in V stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_p	duration of time spent in the P stratum, partially vaccinated (accepting only the first dose)
vea	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against acquisition (between 0-1)
vei	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against infectiousness (between 0-1)

ved	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against duration (between 0-1)
ves	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against symptoms (between 0-1)
dur_revax	scalar or numeric vector with same length as 'gono_params' giving duration of protection for revaccination, default to same as primary
vea_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against acquisition, default to same as primary
vei_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against infectiousness, default to same as primary
ved_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against duration of infection, default to same as primary
ves_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against symptoms, default to same as primary
vea_p	scalar indicating efficacy of partial vaccination against acquisition (between 0-1)
vei_p	scalar indicating efficacy of partial vaccination against infectiousness (between 0-1)
ved_p	scalar indicating efficacy of partial vaccination against duration (between 0-1)
ves_p	scalar indicating efficacy of partial vaccination against symptoms (between 0-1)
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
r1	scalar or numeric vector with same length as 'gono_params' giving proportion of population offered vaccine only accepting the first dose, becoming partially vaccinated
r2	scalar or numeric vector with same length as 'gono_params' giving proportion of the population who accepted the first dose of the vaccine who go on to accept the second dose, becoming fully vaccinated
r2_p	scalar or numeric vector with same length as 'gono_params' giving proportion of partially vaccinated individuals who later receive a second dose when returning to the clinic due to screening or illness
booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned. Defaults to supplied value of $r1 * r2$
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
t_stop	time at which vaccination should stop (years)
hes	Proportion of individuals in the population who are vaccine hesitant
n_erlang	integer giving the number of erlang vaccination transitions through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
years_history	number of years that diagnosis history is recorded for

Value

A list of transformed model outputs

run_onevax_xpvwrh_trackvt

run model with a two-dose vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum, those with waned partial vaccines return to the unvaccinated stratum (X) and considered immunologically naive. A user defined proportion of the population is vaccine hesitant and is never vaccinated. Full acciantion (V) with 2 doses gives maximum protection whereas partial vaccination with 1 dose (P) gives less. Individuals can get 2 doses either by committing in X or upon visiting a clinic in P.

Description

run model with a two-dose vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum, those with waned partial vaccines return to the unvaccinated stratum (X) and considered immunologically naive. A user defined proportion of the population is vaccine hesitant and is never vaccinated. Full acciantion (V) with 2 doses gives maximum protection whereas partial vaccination with 1 dose (P) gives less. Individuals can get 2 doses either by committing in X or upon visiting a clinic in P.

Usage

```
run_onevax_xpvwrh_trackvt(
    tt,
    gono_params,
    init_params = NULL,
    dur_va = 1000,
    dur_vb = 1,
    dur_p = dur_va,
    vea = 0,
    vei = 0,
    ved = 0,
    ves = 0,
    dur_revaxa = dur_va,
    dur_revaxb = 1,
    vea_revax = vea,
    vei_revax = vei,
    ved_revax = ved,
    ves_revax = ves,
    vea_p = vea,
    vei_p = vei,
    ved_p = ved,
    ves_p = ves,
    vbe = 0,
    r1 = 0,
```

```

    r2 = 0,
    r1_p = 0,
    r2_p = 0,
    booster_uptake = (r1 * r2),
    strategy = NULL,
    t_stop = 99,
    hes = 0,
    n_erlang = 1,
    n_diag_rec = 1,
    years_history = 1
)

```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	list of gono params
init_params	A list of starting conditions
dur_va	duration of time spent in Va stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_vb	duration of time spent in Vb stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_p	duration of time spent in the P stratum, partially vaccinated (accepting only the first dose)
vea	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against acquisition (between 0-1)
vei	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against duration (between 0-1)
ves	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against symptoms (between 0-1)
dur_revaxa	duration of time spent in Ra stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_revaxb	duration of time spent in Rb stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
vea_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against acquisition, default to same as primary
vei_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against infectiousness, default to same as primary
ved_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against duration of infection, default to same as primary
ves_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against symptoms, default to same as primary
vea_p	scalar indicating efficacy of partial vaccination against acquisition (between 0-1)

vei_p	scalar indicating efficacy of partial vaccination against infectiousness (between 0-1)
ved_p	scalar indicating efficacy of partial vaccination against duration (between 0-1)
ves_p	scalar indicating efficacy of partial vaccination against symptoms (between 0-1)
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
r1	scalar or numeric vector with same length as 'gono_params' giving proportion of population offered vaccine only accepting the first dose, becoming partially vaccinated
r2	scalar or numeric vector with same length as 'gono_params' giving proportion of the population who accepted the first dose of the vaccine who go on to accept the second dose, becoming fully vaccinated
r1_p	proportion of partially vaccinated individuals who receive $\geq$ one dose when returning to the clinic due to screening or illness
r2_p	proportion of partially vaccinated individuals who receive two doses when returning to the clinic due to screening or illness
booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned. Defaults to supplied value of $r1 * r2$
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
t_stop	time at which vaccination should stop (years)
hes	Proportion of individuals in the population who are vaccine hesitant
n_erlang	integer giving the number of erlang vaccination transitions through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
years_history	number of years that diagnosis history is recorded for

Value

A list of transformed model outputs

run_onevax_xvw	<i>Run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are not eligible for revaccination.</i>
----------------	--

Description

Run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are not eligible for revaccination.

Usage

```
run_onevax_xvw(
  tt,
  gono_params,
  init_params = NULL,
  dur = 1000,
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  vbe = coverage,
  n_diag_rec = 1,
  uptake = 0,
  strategy = NULL,
  coverage = 0,
  t_stop = 99
)
```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	list of gono params
init_params	A list of starting conditions
dur	scalar or numeric vector with same length as 'gono_params' giving duration of the vaccine (in years)
vea	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against acquisition (between 0-1)
vei	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against duration (between 0-1)
ves	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against symptoms (between 0-1)
vbe	scalar giving uptake of vaccination before entry into population (i.e. adolescent vaccination) defaults to same as @param n_diag_rec integer for the number of diagnosis history substrata 'coverage'
n_diag_rec	integer for the number of diagnosis history substrata
uptake	scalar or numeric vector with same length as 'gono_params' giving pc of population vaccinated as part of strategy
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
coverage	scalar giving initial coverage of vaccination, default 0.
t_stop	time at which vaccination should stop (years)

run_onevax_xvw_trial *Run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are not eligible for re-vaccination.*

Description

Run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are not eligible for re-vaccination.

Usage

```
run_onevax_xvw_trial(
  tt,
  gono_params,
  initial_params_trial = NULL,
  dur = 1000,
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  p_v = 0.5,
  n_erlang = 1,
  stochastic = FALSE,
  N = 6e+05,
  n_diag_rec = 1,
  asymp_recorded = TRUE
)
```

Arguments

tt	a numeric vector of times at which the model state is output in years
gono_params	list of gono params for a vaccination trial
initial_params_trial	list of initial conditions for model trial. Set default as NULL.
dur	scalar or numeric vector with same length as 'gono_params' giving duration of the vaccine (in years)
vea	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against acquisition (between 0-1)
vei	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against duration (between 0-1)
ves	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against symptoms (between 0-1)

p_v	scalar giving proportion of the trial cohort vaccinated, default is 0.5.
n_erlang	integer giving the number of transitions that need to be made
stochastic	logical indicating if the run should be made with the default deterministic trial model in continuous time or stochastic trial model in discrete time
N	integer to assign the total number of individuals in the trial (split equally across the two arms)
n_diag_rec	integer giving the number of each X, V(erlang), and W stratum, allowing tracking of diagnosis history. e.g for a n_diag_rec = 2 and erlang = 1, there will be X.I, X.II, V1.I, V1.II, W.I, W.II strata. Where '.I' corresponds to never-diagnosed individuals and '.II' is for individuals diagnosed at least once.
asypm_recorded	logical indicating if the trial screens for and records asymptomatic diagnosis. If FALSE, asymptomatic infected individuals undergoing treatment do not move diagnosis history stratum

run_onevax_xvwr	<i>run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum</i>
-----------------	--

Description

run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum

Usage

```
run_onevax_xvwr(
  tt,
  gono_params,
  init_params = NULL,
  dur = 1000,
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  dur_revax = dur,
  vea_revax = vea,
  vei_revax = vei,
  ved_revax = ved,
  ves_revax = ves,
  vbe = 0,
  n_diag_rec = 1,
  primary_uptake = 0,
  booster_uptake = primary_uptake,
  strategy = NULL,
  t_stop = 99
)
```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	list of gono params
init_params	A list of starting conditions
dur	scalar or numeric vector with same length as 'gono_params' giving duration of the vaccine (in years)
vea	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against acquisition (between 0-1)
vei	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against duration (between 0-1)
ves	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against symptoms (between 0-1)
dur_revax	scalar or numeric vector with same length as 'gono_params' giving duration of protection for revaccination, default to same as primary
vea_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against acquisition, default to same as primary
vei_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against infectiousness, default to same as primary
ved_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against duration of infection, default to same as primary
ves_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against symptoms, default to same as primary
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
n_diag_rec	integer for the number of diagnosis history substrata
primary_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking primary vaccination as part of strategy
booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned. Defaults to supplied value of 'primary_uptake'. @param n_diag_rec integer for the number of diagnosis history substrata
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
t_stop	time at which vaccination should stop (years)

Value

A list of transformed model outputs

run_onevax_xvwrh	<i>run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum</i>
------------------	--

Description

run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination (R), and return to the R stratum

Usage

```
run_onevax_xvwrh(
  tt,
  gono_params,
  init_params = NULL,
  dur = 1000,
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  dur_revax = dur,
  vea_revax = vea,
  vei_revax = vei,
  ved_revax = ved,
  ves_revax = ves,
  vbe = 0,
  primary_uptake = 0,
  booster_uptake = primary_uptake,
  strategy = NULL,
  t_stop = 99,
  hes = 0,
  n_diag_rec = 1
)
```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	list of gono params
init_params	A list of starting conditions
dur	scalar or numeric vector with same length as 'gono_params' giving duration of the vaccine (in years)
vea	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against acquisition (between 0-1)
vei	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against infectiousness (between 0-1)

ved	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against duration (between 0-1)
ves	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against symptoms (between 0-1)
dur_revax	scalar or numeric vector with same length as 'gono_params' giving duration of protection for revaccination, default to same as primary
vea_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against acquisition, default to same as primary
vei_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against infectiousness, default to same as primary
ved_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against duration of infection, default to same as primary
ves_revax	scalar or numeric vector with same length as 'gono_params' giving efficacy of revaccination against symptoms, default to same as primary
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
primary_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking primary vaccination as part of strategy
booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned. Defaults to supplied value of 'primary_uptake'.
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
t_stop	time at which vaccination should stop (years)
hes	Proportion of individuals in the population who are vaccine hesitant
n_diag_rec	integer for the number of diagnosis history substrata

Value

A list of transformed model outputs

run_onevax_xvww	<i>run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination, and return to the V compartment</i>
-----------------	--

Description

run model with single vaccine for input parameter sets, either from initialisation or from equilibrium, those with waned vaccines are eligible for revaccination, and return to the V compartment

Usage

```
run_onevax_xvww(
  tt,
  gono_params,
  init_params = NULL,
  dur = 1000,
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  vbe = 0,
  n_diag_rec = 1,
  uptake = 0,
  strategy = NULL,
  t_stop = 99
)
```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	list of gono params
init_params	A list of starting conditions
dur	scalar or numeric vector with same length as 'gono_params' giving duration of the vaccine (in years)
vea	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against acquisition (between 0-1)
vei	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against duration (between 0-1)
ves	scalar or numeric vector with same length as 'gono_params' giving efficacy of the vaccine against symptoms (between 0-1)
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
n_diag_rec	integer for the number of diagnosis history substrata
uptake	scalar or numeric vector with same length as 'gono_params' giving pc of population vaccinated as part of strategy @param n_diag_rec integer for the number of diagnosis history substrata
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
t_stop	time at which vaccination should stop (years)

run_trial	<i>Run odin model of gonorrhoea vaccine trial with or without vaccination</i>
-----------	---

Description

Run odin model of gonorrhoea vaccine trial with or without vaccination

Usage

```
run_trial(
  tt,
  gono_params,
  init_params = NULL,
  vax_params = NULL,
  transform = TRUE,
  stochastic = FALSE,
  N = 6e+05
)
```

Arguments

tt	a numeric vector of times at which the model state is output in years
gono_params	a data frame of parameters
init_params	= NULL
vax_params	= NULL
transform	= TRUE
stochastic	logical indicating if the run should be made with the default deterministic trial model in continuous time or stochastic trial model in discrete time
N	integer to assign the total number of individuals in the trial (split equally across the two arms)

run_xpvrh	<i>Run odin model of gonorrhoea with or without vaccination</i>
-----------	---

Description

Run odin model of gonorrhoea with or without vaccination

Usage

```
run_xpvwrh(
  tt,
  gono_params,
  demographic_params = NULL,
  init_params = NULL,
  vax_params = NULL,
  n_erlang = 1,
  n_diag_rec = 1,
  years_history = 1,
  transform = TRUE
)
```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	a data frame of parameters
demographic_params	A dataframe of demographic parameters
init_params	A list of starting conditions
vax_params	A vector of vaccination params
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
years_history	number of years that diagnosis history is recorded for
transform	= TRUE

run_xpvwrh_trackvt	<i>Run odin model of gonorrhoea with or without vaccination</i>
--------------------	---

Description

Run odin model of gonorrhoea with or without vaccination

Usage

```
run_xpvwrh_trackvt(
  tt,
  gono_params,
  demographic_params = NULL,
  init_params = NULL,
  vax_params = NULL,
  n_erlang = 1,
  n_diag_rec = 1,
  years_history = 1,
  transform = TRUE
)
```

Arguments

tt	a numeric vector of times at which the model state is output
gono_params	a data frame of parameters
demographic_params	A dataframe of demographic parameters
init_params	A list of starting conditions
vax_params	A vector of vaccination params
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
years_history	number of years that diagnosis history is recorded for
transform	= TRUE

set_protection	<i>generates vector which tells the model which strata are under vaccine protection and what the value of protection for that strata is</i>
----------------	---

Description

generates vector which tells the model which strata are under vaccine protection and what the value of protection for that strata is

Usage

```
set_protection(i_v, idx, n_vax, ve_p, ve, ve_revax)
```

Arguments

i_v	indices of strata receiving protection through vaccination
idx	list containing indices of all X, P, V, W, R & H strata and n_vax through vaccine-protected strata until that protection has waned through vaccine-protected strata until that protection has waned
n_vax	integer denoting total number of strata
ve_p	scalar 0-1 with degree of partial primary protection of the P(N) strata, can take vea_p, vei_p, ves_p, ved_p
ve	scalar 0-1 with degree of full primary protection of the V(N) strata, can take vea, vei, ves, ved
ve_revax	scalar 0-1 with degree of re-vaccinated protection of the R(N) strata, can take vea_revax, vei_revax, ves_revax, ved_revax

Value

vector of length n_vax with zeros corresponding to the indices of strata with no protection, and the supplied degree of partial, full, and boosted protection corresponding to the indices of strata with partial, full and boosted vaccination status

```
set_protection_trackvt
```

generates vector which tells the model which strata are under vaccine protection and what the value of protection for that strata is

Description

generates vector which tells the model which strata are under vaccine protection and what the value of protection for that strata is

Usage

```
set_protection_trackvt(i_v, idx, n_vax, ve_p, ve, ve_revax)
```

Arguments

i_v	indices of strata receiving protection through vaccination
idx	list containing indices of all X, P, V, W, R & H strata and n_vax through vaccine-protected strata until that protection has waned through vaccine-protected strata until that protection has waned
n_vax	integer denoting total number of strata
ve_p	scalar 0-1 with degree of partial primary protection of the P(N) strata, can take vea_p, vei_p, ves_p, ved_p
ve	scalar 0-1 with degree of full primary protection of the V(N) strata, can take vea, vei, ves, ved
ve_revax	scalar 0-1 with degree of re-vaccinated protection of the R(N) strata, can take vea_revax, vei_revax, ves_revax, ved_revax

Value

vector of length n_vax with zeros corresponding to the indices of strata with no protection, and the supplied degree of partial, full, and boosted protection corresponding to the indices of strata with partial, full and boosted vaccination status

set_strategy	<i>Translate each named vaccine strategy into a format interpretable by 'create_vax_map'</i>
--------------	--

Description

Translate each named vaccine strategy into a format interpretable by 'create_vax_map'

Usage

```
set_strategy(strategy = NULL, include_vbe = FALSE)
```

Arguments

strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)", "VoS" describing who is offered vaccination. defaults to no vaccination.
include_vbe	single logical indicating whether vaccination before entry is offered. Defaults to FALSE.

Value

list with entries 'vod', 'vos' and 'vbe' containing 0-1 vectors of length two indicating whether group L and/or H are offered vaccination via each potential vaccination route (i.e. screening, diagnosis, entry)

stratum_index_xpvwrh *Generate the indices of all xpvwrh strata*

Description

Generate the indices of all xpvwrh strata

Usage

```
stratum_index_xpvwrh(n_erlang = 1, n_diag_rec = 1, strategy = NULL)
```

Arguments

n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
strategy	string of vaccination strategy being considered

Value

A list of strata with their indices

stratum_index_xpvwrh_trackvt	<i>Generate the indices of all xpvwrh strata trcaking time since vacc</i>
------------------------------	---

Description

Generate the indices of all xpvwrh strata trcaking time since vacc

Usage

stratum_index_xpvwrh_trackvt(n_erlang = 1, n_diag_rec = 1, strategy = NULL)

Arguments

- | | |
|------------|--|
| n_erlang | integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned |
| n_diag_rec | integer for the number of diagnosis history substrata |
| strategy | string of vaccination strategy being considered |

Value

A list of strata with their indicies

stratum_index_xvw	<i>Generate the indices of all xvwv trial strata</i>
-------------------	--

Description

Generate the indices of all xvwv trial strata

Usage

stratum_index_xvw(n_erlang = 1, n_diag_rec = 1, strategy = NULL)

Arguments

- | | |
|------------|--|
| n_erlang | integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned |
| n_diag_rec | integer for the number of diagnosis history substrata |
| strategy | string of vaccination strategy being considered |

Value

A list of strata with their indicies

stratum_index_xvw_trial

Generate the indices of all xvw trial strata

Description

Generate the indices of all xvw trial strata

Usage

```
stratum_index_xvw_trial(n_erlang, n_diag_rec = 1)
```

Arguments

n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer giving the number of each X, V(erlang), and W stratum, allowing tracking of diagnosis history. e.g for a n_diag_rec = 2 and erlang = 1, there will be X.I, X.II, V1.I, V1.II, W.I, W.II strata. Where '.I' corresponds to never-diagnosed individuals and '.II' is for individuals diagnosed at least once.

Value

A list of strata with their indices

stratum_index_xvwr

Generate the indices of all xvwr trial strata

Description

Generate the indices of all xvwr trial strata

Usage

```
stratum_index_xvwr(n_erlang = 1, n_diag_rec = 1, strategy = NULL)
```

Arguments

n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	integer for the number of diagnosis history substrata
strategy	string of vaccination strategy being considered

Value

A list of strata with their indices

stratum_index_xvwrh	<i>Generate the indices of all xvwrh trial strata</i>
---------------------	---

Description

Generate the indices of all xvwrh trial strata

Usage

```
stratum_index_xvwrh(n_erlang = 1, n_diag_rec = 1, strategy = NULL)
```

Arguments

- | | |
|------------|--|
| n_erlang | integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned |
| n_diag_rec | integer for the number of diagnosis history substrata |
| strategy | string of vaccination strategy being considered |

Value

A list of strata with their indicies

stratum_index_xvww	<i>Generate the indices of all xvww trial strata</i>
--------------------	--

Description

Generate the indices of all xvww trial strata

Usage

```
stratum_index_xvww(n_erlang = 1, n_diag_rec = 1, strategy = NULL)
```

Arguments

- | | |
|------------|--|
| n_erlang | integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned |
| n_diag_rec | integer for the number of diagnosis history substrata |
| strategy | string of vaccination strategy being considered |

Value

A list of strata with their indicies

transform	<i>Transform fitted parameters into gonovax params</i>
-----------	--

Description

Transform fitted parameters into gonovax params

Usage

```
transform(pars, fix_par_t = TRUE)
```

Arguments

pars	list of fitted parameters
fix_par_t	logical indicating whether parameters with inferred trends are held at their 2020 values after that date.

Value

A list of parameters for use in the model

transform_fixed	<i>Transform fitted parameters into non-time-varying gonovax params</i>
-----------------	---

Description

Transform fitted parameters into non-time-varying gonovax params

Usage

```
transform_fixed(pars)
```

Arguments

pars	list of fitted parameters
------	---------------------------

Value

A list of parameters for use in the model

vax_params_xpvwrh	<i>create vaccination parameters for use in onevax_xpvwrh model</i>
-------------------	---

Description

create vaccination parameters for use in onevax_xpvwrh model

Usage

```
vax_params_xpvwrh(
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  vea_revax = vea,
  vei_revax = vei,
  ved_revax = ved,
  ves_revax = ves,
  vea_p = vea,
  vei_p = vei,
  ved_p = ved,
  ves_p = ves,
  dur_v = 1000,
  dur_p = dur_v,
  dur_revax = dur_v,
  r1 = 0,
  r2 = 0,
  r2_p = 0,
  booster_uptake = r1 * r2,
  strategy = NULL,
  vbe = 0,
  t_stop = 99,
  hes = 0,
  n_erlang = 1,
  n_diag_rec = 1,
  years_history = 1
)
```

Arguments

vea	scalar indicating efficacy of the vaccine against acquisition (between 0-1)
vei	scalar indicating efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar indicating efficacy of the vaccine against duration (between 0-1)
ves	scalar indicating efficacy of the vaccine against symptoms (between 0-1)
vea_revax	scalar indicating efficacy of revaccination against acquisition (between 0-1)

vei_revax	scalar indicating efficacy of revaccination against infectiousness (between 0-1)
ved_revax	scalar indicating efficacy of revaccination against duration of infection (between 0-1)
ves_revax	scalar indicating efficacy of revaccination against symptoms (between 0-1)
vea_p	scalar indicating efficacy of partial vaccination against acquisition (between 0-1)
vei_p	scalar indicating efficacy of partial vaccination against infectiousness (between 0-1)
ved_p	scalar indicating efficacy of partial vaccination against duration (between 0-1)
ves_p	scalar indicating efficacy of partial vaccination against symptoms (between 0-1)
dur_v	duration of time spent in V stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_p	duration of time spent in the P stratum, partially vaccinated (accepting only the first dose)
dur_revax	duration of protection for revaccination, default to same as primary
r1	proportion of population offered vaccine only accepting the first dose, becoming partially vaccinated
r2	proportion of the population who accepted the first dose of the vaccine who go on to accept the second dose, becoming fully vaccinated
r2_p	proportion of partially vaccinated individuals who receive a second dose when returning to the clinic due to screening or illness
booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned @param n_diag_rec integer for the number of diagnosis history substrata
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
t_stop	time at which vaccination should stop (years)
hes	proportion of population vaccine hesitant
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	number of diagnosis history strata
years_history	number of years that diagnosis history is recorded for

Value

A list parameters in the model input format

vax_params_xpvwrh_trackvt

create vaccination parameters for use in onevax_xpvwrh model

Description

create vaccination parameters for use in onevax_xpvwrh model

Usage

```
vax_params_xpvwrh_trackvt(
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  vea_revax = vea,
  vei_revax = vei,
  ved_revax = ved,
  ves_revax = ves,
  vea_p = vea,
  vei_p = vei,
  ved_p = ved,
  ves_p = ves,
  dur_va = 1000,
  dur_vb = 1,
  dur_p = dur_va,
  dur_revaxa = dur_va,
  dur_revaxb = 1,
  r1 = 0,
  r2 = 0,
  r1_p = 0,
  r2_p = 0,
  booster_uptake = r1 * r2,
  strategy = NULL,
  vbe = 0,
  t_stop = 99,
  hes = 0,
  n_erlang = 1,
  n_diag_rec = 1,
  years_history = 1
)
```

Arguments

vea	scalar indicating efficacy of the vaccine against acquisition (between 0-1)
vei	scalar indicating efficacy of the vaccine against infectiousness (between 0-1)

ved	scalar indicating efficacy of the vaccine against duration (between 0-1)
ves	scalar indicating efficacy of the vaccine against symptoms (between 0-1)
vea_revax	scalar indicating efficacy of revaccination against acquisition (between 0-1)
vei_revax	scalar indicating efficacy of revaccination against infectiousness (between 0-1)
ved_revax	scalar indicating efficacy of revaccination against duration of infection (between 0-1)
ves_revax	scalar indicating efficacy of revaccination against symptoms (between 0-1)
vea_p	scalar indicating efficacy of partial vaccination against acquisition (between 0-1)
vei_p	scalar indicating efficacy of partial vaccination against infectiousness (between 0-1)
ved_p	scalar indicating efficacy of partial vaccination against duration (between 0-1)
ves_p	scalar indicating efficacy of partial vaccination against symptoms (between 0-1)
dur_va	duration of time spent in Va stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_vb	duration of time spent in Vb stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_p	duration of time spent in the P stratum, partially vaccinated (accepting only the first dose)
dur_revaxa	duration of time spent in Ra stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
dur_revaxb	duration of time spent in Rb stratum after completing a round of primary vaccination (fully vaccinated, accepting first and second dose)
r1	proportion of population offered vaccine only accepting the first dose, becoming partially vaccinated
r2	proportion of the population who accepted the first dose of the vaccine who go on to accept the second dose, becoming fully vaccinated
r1_p	proportion of partially vaccinated individuals who receive $\geq$ one dose when returning to the clinic due to screening or illness
r2_p	proportion of partially vaccinated individuals who receive two doses when returning to the clinic due to screening or illness
booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned @param n_diag_rec integer for the number of diagnosis history substrata
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
t_stop	time at which vaccination should stop (years)
hes	proportion of population vaccine hesitant
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
n_diag_rec	number of diagnosis history strata
years_history	number of years that diagnosis history is recorded for

Value

A list parameters in the model input format

vax_params_xvw	Create vaccination parameters for use in onevax_xvw model
----------------	---

Description

Create vaccination parameters for use in onevax_xvw model

Usage

```
vax_params_xvw(  
  vea = 0,  
  vei = 0,  
  ved = 0,  
  ves = 0,  
  dur = 1000,  
  uptake = 0,  
  strategy = NULL,  
  vbe = 0,  
  t_stop = 99,  
  n_diag_rec = 1  
)
```

Arguments

vea	scalar indicating efficacy of the vaccine against acquisition (between 0-1)
vei	scalar indicating efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar indicating efficacy of the vaccine against duration (between 0-1)
ves	scalar indicating efficacy of the vaccine against symptoms (between 0-1)
dur	scalar indicating duration of the vaccine (in years)
uptake	scalar indicating pc of those offered who accept vaccination
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
t_stop	time at which vaccination should stop (years)
n_diag_rec	integer for the number of diagnosis history substrata

Value

A list parameters in the model input format

vax_params_xvw_trial *Create vaccination parameters for use in onevax_xvw_trial model, assign who experiences vaccine effects, and how waning occurs.*

Description

Create vaccination parameters for use in onevax_xvw_trial model, assign who experiences vaccine effects, and how waning occurs.

Usage

```
vax_params_xvw_trial(
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  dur = 1000,
  n_erlang = 1,
  stochastic = FALSE,
  n_diag_rec = 1,
  asymp_recorded = TRUE
)
```

Arguments

vea	scalar indicating efficacy of the vaccine against acquisition (between 0-1)
vei	scalar indicating efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar indicating efficacy of the vaccine against duration (between 0-1)
ves	scalar indicating efficacy of the vaccine against symptoms (between 0-1)
dur	scalar indicating duration of the vaccine (in years)
n_erlang	integer giving the number of transitions that need to be made through vaccine-protected strata until that protection has waned
stochastic	logical indicating if the parameters are for the default deterministic trial model in continuous time or stochastic trial model in discrete time
n_diag_rec	integer giving the number of each X, V(erlang), and W stratum, allowing tracking of diagnosis history. e.g for a n_diag_rec = 2 and erlang = 1, there will be X.I, X.II, V1.I, V1.II, W.I, W.II strata. Where '.I' corresponds to never-diagnosed individuals and '.II' is for individuals diagnosed at least once.
asymp_recorded	logical indicating if the trial screens for and records asymptomatic diagnosis. If FALSE, asymptomatic infected individuals undergoing treatment do not move diagnosis history stratum

Value

A list of parameters in the model input format

vax_params_xvwr	<i>create vaccination parameters for use in onevax_xvwr model</i>
-----------------	---

Description

create vaccination parameters for use in onevax_xvwr model

Usage

```
vax_params_xvwr(
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  vea_revax = vea,
  vei_revax = vei,
  ved_revax = ved,
  ves_revax = ves,
  dur = 1000,
  dur_revax = dur,
  primary_uptake = 0,
  booster_uptake = primary_uptake,
  strategy = NULL,
  vbe = 0,
  t_stop = 99,
  n_diag_rec = 1
)
```

Arguments

vea	scalar indicating efficacy of the vaccine against acquisition (between 0-1)
vei	scalar indicating efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar indicating efficacy of the vaccine against duration (between 0-1)
ves	scalar indicating efficacy of the vaccine against symptoms (between 0-1)
vea_revax	scalar indicating efficacy of revaccination against acquisition (between 0-1)
vei_revax	scalar indicating efficacy of revaccination against infectiousness (between 0-1)
ved_revax	scalar indicating efficacy of revaccination against duration of infection (between 0-1)
ves_revax	scalar indicating efficacy of revaccination against symptoms (between 0-1)
dur	scalar indicating duration of the vaccine (in years)
dur_revax	duration of protection for revaccination, default to same as primary
primary_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking primary vaccination as part of strategy

booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned @param n_diag_rec integer for the number of diagnosis history substrata
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
t_stop	time at which vaccination should stop (years)
n_diag_rec	integer for the number of diagnosis history substrata

Value

A list parameters in the model input format

vax_params_xvwrh	<i>create vaccination parameters for use in onevax_xvwrh model</i>
------------------	--

Description

create vaccination parameters for use in onevax_xvwrh model

Usage

```
vax_params_xvwrh(
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  vea_revax = vea,
  vei_revax = vei,
  ved_revax = ved,
  ves_revax = ves,
  dur = 1000,
  dur_revax = dur,
  primary_uptake = 0,
  booster_uptake = primary_uptake,
  strategy = NULL,
  vbe = 0,
  t_stop = 99,
  hes = 0,
  n_diag_rec = 1
)
```

Arguments

vea	scalar indicating efficacy of the vaccine against acquisition (between 0-1)
vei	scalar indicating efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar indicating efficacy of the vaccine against duration (between 0-1)
ves	scalar indicating efficacy of the vaccine against symptoms (between 0-1)
vea_revax	scalar indicating efficacy of revaccination against acquisition (between 0-1)
vei_revax	scalar indicating efficacy of revaccination against infectiousness (between 0-1)
ved_revax	scalar indicating efficacy of revaccination against duration of infection (between 0-1)
ves_revax	scalar indicating efficacy of revaccination against symptoms (between 0-1)
dur	scalar indicating duration of the vaccine (in years)
dur_revax	duration of protection for revaccination, default to same as primary
primary_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking primary vaccination as part of strategy
booster_uptake	scalar or numeric vector with same length as 'gono_params' giving proportion of population undertaking booster vaccination after primary vaccination protection has waned @param n_diag_rec integer for the number of diagnosis history substrata
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
t_stop	time at which vaccination should stop (years)
hes	proportion of population vaccine hesitant
n_diag_rec	integer for the number of diagnosis history substrata

Value

A list parameters in the model input format

vax_params_xvww

create vaccination parameters for use in onevax_xvww model

Description

create vaccination parameters for use in onevax_xvww model

Usage

```
vax_params_xvww(
  vea = 0,
  vei = 0,
  ved = 0,
  ves = 0,
  dur = 1000,
  uptake = 0,
  strategy = NULL,
  vbe = 0,
  t_stop = 99,
  n_diag_rec = 1
)
```

Arguments

vea	scalar indicating efficacy of the vaccine against acquisition (between 0-1)
vei	scalar indicating efficacy of the vaccine against infectiousness (between 0-1)
ved	scalar indicating efficacy of the vaccine against duration (between 0-1)
ves	scalar indicating efficacy of the vaccine against symptoms (between 0-1)
dur	scalar indicating duration of the vaccine (in years)
uptake	scalar indicating pc of those offered who accept vaccination
strategy	single character string in "VoD", "VoD(H)", "VoA", "VoA(H)", "VoD(L)+VoA(H)". Defaults to NULL i.e. no vaccination
vbe	scalar indicating pc of population vaccinated before entry (between 0-1)
t_stop	time at which vaccination should stop (years)
n_diag_rec	integer for the number of diagnosis history substrata

Value

A list parameters in the model input format

vax_params0	<i>create vaccination parameters for use in novax model (null)</i>
-------------	--

Description

create vaccination parameters for use in novax model (null)

Usage

```
vax_params0(n_diag_rec = 1, years_history = 1)
```

Arguments

- `n_diag_rec` integer giving the number of each X, V(erlang), and W
- `years_history` number of years that diagnosis history is recorded for stratum, allowing tracking of diagnosis history. e.g for a `n_diag_rec` = 2 and `erlang` = 1, there will be X.I, X.II, V1.I, V1.II, W.I, W.II strata. Where '.I' corresponds to never-diagnosed individuals and '.II' is for individuals diagnosed at least once.

Value

A list parameters in the model input format

Index

adjust_baseline, 3
aggregate, 4

compare, 5
compare_baseline, 5
compare_baseline_xpvwrh, 7
compare_basic, 10
create_diagnosis_waning_map, 11
create_uptake_map, 11
create_uptake_map_xpvwrh, 12
create_uptake_map_xpvwrh_trackvt, 13
create_uptake_map_xvw, 14
create_uptake_map_xvwr, 15
create_uptake_map_xvwv, 16
create_vax_map, 17
create_vax_map_branching, 18
create_waning_map, 18
create_waning_map_branching, 19
create_waning_map_trial, 20

dbetabinom, 20

extract_flows, 21
extract_flows_trial, 21
extract_flows_xpvwrh, 22

gen_erlang_labels, 22
gen_erlang_labels_trackvt, 23
gen_labels, 23
gen_trial_labels, 24
gono_params, 24
gono_params_trial, 25
gonovax_data, 25
gonovax_year, 26
gonovax_year_as_year, 26

initial_params, 27
initial_params_trial, 27
initial_params_xpvwrh, 28
initial_params_xpvwrh_trackvt, 29
initial_params_xvw (initial_params), 27

initial_params_xvw_trial, 30
initial_params_xvwrh, 30

mcmc, 31
mcmc_combine, 32
mcmc_sample (mcmc_thin), 32
mcmc_thin, 32
model, 33
model_params, 33
model_params_trial, 34
model_params_xpvwrh, 35
model_params_xpvwrh_trackvt, 36

restart_hes, 37
restart_params, 38
run, 38
run_grid, 39
run_onevax_xpvwrh, 40
run_onevax_xpvwrh_trackvt, 43
run_onevax_xvw, 45
run_onevax_xvw_trial, 47
run_onevax_xvwr, 48
run_onevax_xvwrh, 50
run_onevax_xvwv, 51
run_trial, 53
run_xpvwrh, 53
run_xpvwrh_trackvt, 54

set_protection, 55
set_protection_trackvt, 56
set_strategy, 56
stratum_index_xpvwrh, 57
stratum_index_xpvwrh_trackvt, 58
stratum_index_xvw, 58
stratum_index_xvw_trial, 59
stratum_index_xvwr, 59
stratum_index_xvwrh, 60
stratum_index_xvwv, 60

transform, 61

`transform_fixed`, [61](#)

`vax_params0`, [71](#)

`vax_params_xpvwrh`, [62](#)

`vax_params_xpvwrh_trackvt`, [64](#)

`vax_params_xvw`, [66](#)

`vax_params_xvw_trial`, [67](#)

`vax_params_xvwr`, [68](#)

`vax_params_xvwrh`, [69](#)

`vax_params_xvvw`, [70](#)