Package: koalas (via r-universe)

May 21, 2026

Title A Disease Model for Chlamydia in Koalas

Version 0.5.0-1

Date 2026-03-06

Description The koalas package is intended to be used for exploring chlamydia control programmes. The package is still being developed, and feedback is welcomed!

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

LinkingTo Rcpp

Imports methods, rlang, dplyr, ggplot2, ggh4x, tibble, tidyr, purrr, forcats, stringr, IPDMR, checkmate, Rcpp, R6, pbapply

Suggests knitr, rmarkdown, spelling, testthat (>= 3.0.0)

Additional_repositories `https://ku-awdc.r-universe.dev/`

NeedsCompilation yes

SystemRequirements C++20

Config/testthat/edition 3

VignetteBuilder knitr

Language en-GB

Config/pak/sysreqs libicu-dev

Repository https://ku-awdc.r-universe.dev

Date/Publication 2026-03-10 11:43:56 UTC

RemoteUrl https://github.com/ku-awdc/koalas

RemoteRef v0.5.0

RemoteSha 0a79560da00f3ed3ce5712a3acd77c4a3967757b

Contents

`assess_interventions` *Assess different intervention strategies*

Description

Assess different intervention strategies

Usage

```
assess_interventions(
  koala_model,
  years,
  frequency = c(1, 2, 3, 4),
  prop_active = seq(0, 1, by = 0.05),
  prop_targeted = 0,
  cl = NULL,
  ...
)
```

Arguments

`koala_model`	a valid `KoalasV2` model
`years`	the number of years to update the model (passed to the run method of the `KoalasV2` model)
`frequency`	a vector of annual sampling frequencies to test
`prop_active`	a vector of the proportion of animals to test/treat/vaccinate with active sampling at each intervention
`prop_targeted`	a vector of the proportion of diseased animals to treat with targeted interventions at each intervention
`cl`	passed to `pblapply` - DO NOT CHANGE THIS ON WINDOWS as it won't properly without shared memory forking
`...`	additional arguments (cull proportions) passed to the active_intervention method of the `KoalasV2` model

Value

a data frame of final prevalence and population size values at the end of each simulation for each combination of frequency and proportion

---

`KoalasV1` *R6 representation of the Koala model*

Description

This is a within-group SVIDR(D) model class that can either be run on its own or embedded in a between-group model. Note: D is not a real state (and can be negative), it is just used to ensure that the books are balanced including birth/death

Active bindings

`S` number of susceptible animals

`I` total number of infected animals

`D` number of diseased animals

`R` number of recovered animals

`V` number of vaccinated animals

`Ctp` cumulative number of animals removed as true positives on active/passive sampling

`Cfp` cumulative number of animals removed as false positives on active/passive sampling

`mortality_natural` mortality rate from natural causes

`mortality_disease` mortality rate from disease

`carrying_capacity` number of animals supported by the population (can be Inf or NA)

`birth_rate` birth rate

`relative_fecundity` fecundity of diseased animals relative to other states (S/I/R/V)

`beta` the transmission rate parameter per unit time (must be positive)

`sigma` the progression rate from I to D or S

`acute_recovery_prob` the probability that a progression from I will go to S rather than D

`recovery` the recovery rate parameter per unit time (must be positive)

`waning_natural` the immune waning rate from R (following natural infection)

`waning_vaccine` the immune waning rate from V (following vaccination)

`time` the current time point of the model (read-only)

`N` the total number of animals alive (read-only)

`state` a data frame representing the current state of the model (read-only)

`trans_external` the external transmission parameter

`transmission_type` either frequency or density

Methods

Public methods:

- `KoalasV1$new()`
- `KoalasV1$update()`
- `KoalasV1$run()`
- `KoalasV1$vaccinate()`
- `KoalasV1$active_test()`
- `KoalasV1$passive_test()`
- `KoalasV1$print()`
- `KoalasV1$clone()`

Method `new()`: Create a new within-group model

Usage:

```
KoalasV1$new(
  update_type = c("deterministic", "stochastic"),
  group_name = NA_character_
)
```

Arguments:

`update_type` one of deterministic or stochastic

`group_name` an optional name for the group (will be included in the output state, if provided)

Returns: A new within-group model object

Method `update()`: Update the state of the group for a single time point

Usage:

```
KoalasV1$update(d_time)
```

Arguments:

`d_time` the desired time step (delta time)

Returns: self, invisibly

Method `run()`: Update the state of the group for several time points

Usage:

```
KoalasV1$run(add_time, d_time, thin = 1)
```

Arguments:

`add_time` the additional time to add to the current time of the model

`d_time` the desired time step (delta time)

`thin` thinning parameter (currently ignored)

Returns: a data frame of the model state at each (new) time point

Method `vaccinate()`: Implement a (one-time) vaccination effort at the current time point

Usage:

```
KoalasV1$vaccinate(number, proportion, efficacy)
```

Arguments:

`number` the (maximum) number of animals to vaccinate (ignored if proportion is supplied)

`proportion` the proportion of animals to vaccinate

`efficacy` the efficacy of the vaccine i.e. probability of the animal moving to V (from S, I or D - R and V do not move)

Method `active_test():` Implement a (one-time) active sampling/capture/testing of all animals

Usage:

```
KoalasV1$active_test(number, proportion, sensitivity, specificity)
```

Arguments:

`number` the (maximum) number of animals to test and remove (ignored if proportion is supplied)

`proportion` the proportion of animals to test and remove

`sensitivity` the sensitivity of the test

`specificity` the specificity of the test

Method `passive_test():` Implement a (one-time) passive sampling/capture/testing of diseased animals

Usage:

```
KoalasV1$passive_test(
  number,
  proportion,
  sensitivity,
  specificity,
  prevalence_other
)
```

Arguments:

`number` the (maximum) number of animals to test and remove (ignored if proportion is supplied)

`proportion` the proportion of animals to test and remove

`sensitivity` the sensitivity of the test

`specificity` the specificity of the test

`prevalence_other` the prevalence of other conditions that resemble clinical disease

Method `print():` Print method giving an overview of the current state and parameter values

Usage:

```
KoalasV1$print()
```

Returns: self, invisibly

Method `clone():` The objects of this class are cloneable with this method.

Usage:

```
KoalasV1$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

`KoalasV2` *R6/C++ representation of the Koala model V2*

Description

This is the "V2" model that has substantial differences to the Grogan et al model (`KoalasV1`)

Active bindings

`date` the current date of the model (read-only)

`day` the current day number of the model (read-only)

`N` the total number of animals alive (read-only)

`prevalence` the current prevalence (read-only)

`state` a list representing the current state of the model

`parameters` a list of parameter values

`results_wide` a data frame of results from the model in wide format (read-only)

`results_long` a data frame of results from the model in long format, with aggregated/summarised compartments (read-only)

`interventions` a data frame of intervention time points from the model (read-only)

`run_dates` a date vector of dates when run was called, which might be useful for e.g. adding dashed lines to plots (read-only)

`treatments` a data frame of cumulative treatments and vaccinations from the model (read-only)

Methods

Public methods:

- `KoalasV2$new()`
- `KoalasV2$set_parameters()`
- `KoalasV2$set_state()`
- `KoalasV2$update()`
- `KoalasV2$active_intervention()`
- `KoalasV2$targeted_intervention()`
- `KoalasV2$burnin()`
- `KoalasV2$run()`
- `KoalasV2$print()`
- `KoalasV2$autoplot()`
- `KoalasV2$clone()`

Method `new()`**:** Create a new single-group koala model

Usage:

```
KoalasV2$new(
  num = 3L,
  num_V = num,
  num_I = num,
  num_N = num,
  num_R = num,
  num_A = num,
  parameters = list(),
  state = list(),
  start_date = "2021-01-01"
)
```

Arguments:

`num` number of sub-components for all states (unless overridden) - currently must be either 1 or 3

`num_V` number of sub-components for V (and Vf)

`num_I` number of sub-components for I (and If)

`num_N` number of sub-components for N (and Nf)

`num_R` number of sub-components for R (and Rf)

`num_A` number of sub-components for Af

`parameters` a list of parameter values - see the set_parameters method for the allowed values

`state` an initialisation state list - see the set_state method for the allowed values

`start_date` the date corresponding to day 0 of the simulation (only used for outputs)

Returns: A new within-group model object

Method `set_parameters()`**:** Change one or more current parameter values

Usage:

```
KoalasV2$set_parameters(
  vacc_immune_duration,
  vacc_redshed_duration,
  natural_immune_duration,
  beta,
  subcinical_duration,
  subclinical_recover_proportion,
  diseased_recover_proportion,
  birthrate,
  acute_duration,
  lifespan_natural,
  lifespan_acute,
  lifespan_chronic,
  relative_fecundity,
  sensitivity,
  specificity,
  cure_prob_N,
  cure_prob_I,
  cure_prob_A,
  cure_prob_C,
```

```
  vaccine_efficacy,
  vaccine_booster,
  passive_intervention_rate
)
```

Arguments:

`vacc_immune_duration` #1: Average duration of vaccine-related immunity from infection for susceptibles

`vacc_redshed_duration` #2: Average duration of treatment/vaccine-related reduction in shedding for infecteds, relative to #1

`natural_immune_duration` #3: Average duration of natural immunity following resolved infection , relative to #1

`beta` #4: Infection rate (frequency dependent)

`subcinical_duration` #5: Average duration of subclinical infection before progression to “acute” disease OR spontaneous recovery

`subclinical_recover_proportion` #6: Proportion of animals that will spontaneously recover, rather than progressing to acute disease

`diseased_recover_proportion` #7: Spontaneous recovery rate for diseased animals – ASSUMED DOES NOT HAPPEN SO MUST BE ZERO

`birthrate` #8: Birth rate (assumed not density-dependent, for now)

`acute_duration` #9: Average duration of acute (increased mortality) phase before progressing to chronic (normal mortality) phase

`lifespan_natural` #10: Average lifespan of uninfected koalas (assumed not density-dependent, for now)

`lifespan_acute` #11: Disease-related mortality rate (replacement for #10) - default is calibrated so that 25% die before entering Cf

`lifespan_chronic` #11: Disease-related mortality rate (replacement for #10)

`relative_fecundity` #12: Relative fecundity of diseased animals - NOTE: we ignore males as only females are important for reproduction

`sensitivity` sensitivity of lab test to detect shedding in I(f), Af, Cf

`specificity` specificity of lab test to not detect shedding in other compartments

`cure_prob_N` proportion of N(f) animals with a completed treatment course that are cured of infection (go to R(f))

`cure_prob_I` proportion of I(f) animals with a completed treatment course that are cured of infection (go to R(f))

`cure_prob_A` proportion of Af animals with a completed treatment course that are cured of infection (go to Rf)

`cure_prob_C` proportion of Cf animals with a completed treatment course that are cured of infection (go to Rf)

`vaccine_efficacy` proportion of S(f) animals that have effective vaccination i.e. go to V(f)

`vaccine_booster` proportion of already-vaccinated/immune animals that re-start their time in that category due to the “booster effect” i.e. V(f), N(f), and R(f)

`passive_intervention_rate` rate at which animals are brought in for test/treatment/vaccination passively (can be interpreted as the expected number of times per year each animal is brought in)

Returns: self, invisibly

Method `set_state()`**:** Change one or more current state values

Usage:

```
KoalasV2$set_state(
  S,
  V,
  I,
  N,
  R,
  Af,
  Cf,
  Sf,
  Vf,
  If,
  Nf,
  Rf,
  Day,
  SumTx,
  SumVx,
  SumRx,
  SumMx
)
```

Arguments:

`S` number of S

`V` number of V

`I` number of I

`N` number of N

`R` number of R

`Af` number of Af

`Cf` number of Cf

`Sf` number of Sf

`Vf` number of Vf

`If` number of If

`Nf` number of Nf

`Rf` number of Rf

`Day` current day of the simulation (you probably shouldn't change this)

`SumTx` cumulative total number of animals with a successful treatment course (you probably shouldn't change this)

`SumVx` cumulative total number of animals vaccinated - this excludes the animals also treated (you probably shouldn't change this)

`SumRx` cumulative total number of animals removed due to voluntary culling and failure to cure (you probably shouldn't change this)

`SumMx` cumulative mortality due to disease excluding SumRx (you probably shouldn't change this)

Returns: self, invisibly

Method `update()`: Update the model for one or more day

Usage:

```
KoalasV2$update(n_days = 1L, d_time = 1/24, record = TRUE)
```

Arguments:

`n_days` the number of days to update for

`d_time` the desired time step (delta time)

`record` should the state over time be recorded?

Returns: self, invisibly

Method `active_intervention()`: Implement a (one-time) active sampling/capture/testing of all animals

Usage:

```
KoalasV2$active_intervention(
  proportion,
  cull_positive = 0,
  cull_acute = 0.2,
  cull_chronic = 0.3
)
```

Arguments:

`proportion` the proportion of animals to test/treat/vaccinate

`cull_positive` the proportion of test-positive animals that will be culled

`cull_acute` the proportion of acute diseased animals that will be culled

`cull_chronic` the proportion of chronic diseased animals that will be culled

Method `targeted_intervention()`: Implement a (one-time) targeted capture/treatment of diseased animals

Usage:

```
KoalasV2$targeted_intervention(
  proportion,
  prop_acute = proportion,
  prop_chronic = proportion,
  cull_positive = 0,
  cull_acute = 0.2,
  cull_chronic = 0.3
)
```

Arguments:

`proportion` the proportion of diseased animals to identify and treat (overridden by prop_acute and/or prop_chronic, if set)

`prop_acute` the proportion of acutely diseased animals to identify and treat (overrides proportion)

`prop_chronic` the proportion of chronic diseased animals to identify and treat (overrides proportion)

`cull_positive` this parameter is ignored (it is only provided for compatibility with the run method) `cull_acute` the proportion of acute diseased animals that will be culled

`cull_chronic` the proportion of chronic diseased animals that will be culled

Method `burnin()`**:** Set and burn-in the model for standard scenarios (with current parameters)

Usage:

```
KoalasV2$burnin(scenario = c("worst", "best", "combined"), d_time = 1/24)
```

Arguments:

`scenario` the desired population scenario for calibration (worst case, best case, or combined)

`d_time` the desired time step (delta time)

Returns: self, invisibly

Method `run()`**:** Run the model for 1 or more years with the standard scenarios

Usage:

```
KoalasV2$run(
  years,
  frequency,
  prop_active = 0,
  prop_targeted = 0,
  d_time = 1/24,
  ...
)
```

Arguments:

`years` the number of years to run for

`frequency` the number of sampling events per year

`prop_active` the proportion of animals to test/treat/vaccinate with active sampling at each intervention

`prop_targeted` the proportion of diseased animals to treat with targeted interventions at each intervention

`d_time` the desired time step (delta time)

`...` additional arguments (cull proportions) passed to the active_intervention method

Returns: self, invisibly

Method `print()`**:** Print method giving an overview of the current state and parameter values

Usage:

```
KoalasV2$print()
```

Returns: self, invisibly

Method `autoplot()`**:** autoplot method for a default plot

Usage:

```
KoalasV2$autoplot(
  show_treatments = TRUE,
  prev_line = 5,
  number_line = NULL,
  ymax = c(Treatment = NA_real_, Number = NA_real_, Prevalence = 100),
```

```
  alphas = c(Treatment = 0.25, Number = 0.5, Prevalence = 0.25),
  colours = c(Diseased = "#F8766D", Infectious = "#F9C945", Healthy = "#619CFF",
    Treatment = "forestgreen", Prevalence = "black"),
  dot_col = "grey50"
)
```

Arguments:

`show_treatments` option to show/hide treatments

`prev_line` an optional dotted line for target prevalence

`number_line` an optional dotted line for a target stable (or starting) population size - NULL means to use the starting population size

`ymax` a named numeric vector of maximum values for the y axis

`alphas` a named numeric vector of alpha values for each subplot

`colours` a named numeric vector of colour values for each subplot/compartment

`dot_col` the colour to use for the dotted prevalence and number lines

Returns: a ggplot2 object

Method `clone()`**:** The objects of this class are cloneable with this method.

Usage:

```
KoalasV2$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

Index