Package 'smidm'

July 23, 2025

Title Statistical Modelling for Infectious Disease Management

Version 1.0

Description Statistical models for specific coronavirus disease 2019 use cases at German local health authorities. All models of Statistical modelling for infectious disease management 'smidm' are part of the decision support toolkit in the 'EsteR' project. More information is published in Sonja Jäckle, Rieke Alpers, Lisa Kühne, Jakob Schumacher, Benjamin Geisler, Max Westphal ``'EsteR' – A Digital Toolkit for COVID-19 Decision Support in Local Health Authorities" (2022) <doi:10.3233/SHTI220799> and Sonja Jäckle, Elias Röger, Volker Dicken, Benjamin Geisler, Jakob Schumacher, Max Westphal ``A Statistical Model to Assess Risk for Supporting COVID-19 Quarantine Decisions" (2021) <doi:10.3390/ijerph18179166>.

License BSD_3_clause + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.1

Imports dplyr, extraDistr, stats, methods

Suggests testthat (>= 3.0.0), covr, xml2, knitr, rmarkdown, tidyr, hdrcde, ggplot2, scales

Config/testthat/edition 3

VignetteBuilder knitr

URL https://gitlab.cc-asp.fraunhofer.de/ester/smidm

BugReports https://gitlab.cc-asp.fraunhofer.de/ester/smidm/-/issues

NeedsCompilation no

Author Max Westphal [aut] (ORCID: <https://orcid.org/0000-0002-8488-758X>),
Stefanie Grimm [aut],
Sonja Jäckle [aut, cre] (ORCID: <https://orcid.org/0000-0002-2908-299X>),
Rieke Alpers [aut] (ORCID: <https://orcid.org/0000-0001-8317-1435>),
Hong Phuc Truong [aut],
Amelie Lucker [ctb],
Fraunhofer MEVIS [cph],
Fraunhofer ITWM [cph]

Maintainer Sonja Jäckle <sonja.jaeckle@mevis.fraunhofer.de>

Repository CRAN

Date/Publication 2022-08-27 08:40:02 UTC

Contents

calculate_likelihood_negative_tests 2
calculate_likelihood_negative_tests_k 3
calculate_posterior_no_infections 4
calculate_prior_infections 6
generate_data_extended 7
get_expected_total_infections 7
get_incubation_day_distribution 8
get_infection_density 9
get_infectiousness_density 10
get_misc_infection_density 11
get_serial_interval_density 12
get_test_sensitivities 13
predict_future_infections 14
p_onePrimaryMore 15

Index **16**

`calculate_likelihood_negative_tests` *Overall likelihood*

Description

Calculates vector of probabilities that zero positive tests are observed given different numbers of infected.

Usage

```
calculate_likelihood_negative_tests(
  test_infos,
  test_types,
  negative_persons,
  subgroup_size,
  info
)
```

Arguments

test_infos	Matrix with column number of test days and a column for each test with the testday relative to event date, the rows are the groups.
test_types	Matrix with test day (columns) of each group (rows) and whe informations about test types.
negative_persons	Number of people without the infectious persons.
subgroup_size	Array with the number of persons per test group.
info	Dataframe, this is a placeholder

Value

Vector of probabilities calculated.

Examples

```
test_infos <- matrix(nrow = 2, ncol = 3)
test_infos[1,] <- c(1, 2, NA)
test_infos[2,] <- c(2, 4, 6)

test_types <- matrix(nrow = 2, ncol = 2)
test_types[1,] <- c("PCR", NA)
test_types[2,] <- c("PCR", "Antigen")

calculate_likelihood_negative_tests(test_infos = test_infos,
                                    test_types = test_types,
                                    negative_persons = 23,
                                    subgroup_size = c(3, 5))
```

---

calculate_likelihood_negative_tests_k

Likelihood K

Description

Calculates the probability that zero positive tests are observed given K of the group are infected.

Usage

```
calculate_likelihood_negative_tests_k(
  infected_group_size,
  information_data,
  test_infos,
  test_types,
  info,
  combination_infected,
```

```
  number_group_peoples,
  number_subgroups
)
```

Arguments

infected_group_size	Number of infected Persons in the group.
information_data	Matrix with columns person ID, tested (T/F), result(F/NA), testNumbers, groupNumber
test_infos	Matrix with column number of test days and a column for each test with the testday relative to event date, the rows are the groups.
test_types	Matrix with test day (columns) of each group (rows) and whe informations about test types.
info	Dataframe with the day specific information about sensitivity and specificity.
combination_infected	Matrix of all possible combinations how K infected are distributed among subgroups.
number_group_peoples	Vector with the number of people per group.
number_subgroups	Number of subgroups including group of untested (if existent).

Value

The probability.

---

calculate_posterior_no_infections

Negative analysis probability

Description

Calculates the probability that nobody is infected given the negative tests.

Usage

```
calculate_posterior_no_infections(
  negative_persons,
  infected_persons,
  event,
  test_infos,
  test_types,
  subgroup_size,
  distribution = NULL,
  info
)
```

Arguments

negative_persons	Number of people without the infectious persons.
infected_persons	Number of infectious persons.
event	Characters, the name of the event, currently: "school" or "day_care_center".
test_infos	Matrix with testing information; each row gives the number of tests (1. column) and each test date (following columns) for each test group
test_types	Matrix with test day (columns) of each group (rows) and the informations about test types.
subgroup_size	Array with the number of persons per test group.
distribution	Vector, this is a placeholder
info	Dataframe, this is a placeholder

Details

The probability is based on Bayes' theorem.

Value

The probability p.

See Also

calculate_prior_infections, generate_data_extended, get_test_sensitivities and calculate_likelihood_neg

Examples

```
test_infos <- matrix(nrow = 2, ncol = 3)
test_infos[1,] <- c(1, 2, NA)
test_infos[2,] <- c(2, 4, 6)

test_types <- matrix(nrow = 2, ncol = 2)
test_types[1,] <- c("PCR", NA)
test_types[2,] <- c("PCR", "Antigen")

calculate_posterior_no_infections(negative_persons = 23,
                                  infected_persons = 2,
                                  event = "school",
                                  test_infos = test_infos,
                                  test_types = test_types,
                                  subgroup_size = c(3, 5))
```

`calculate_prior_infections` *A priori probability of further Infections*

Description

Calculates the a priori probability of how many people are infected in one event.

Usage

```
calculate_prior_infections(
  negative_persons,
  infected_persons,
  event,
  p_one = NULL,
  infect_average = NULL
)
```

Arguments

`negative_persons`	Number of people without the infectious persons.
`infected_persons`	Number of infected people.
`event`	Characters, event type given as characters, currently: "school" or "day_care_center".
`p_one`	Number, this is a placeholder
`infect_average`	Number, this is a placeholder

Details

The probability is beta-binomial distributed. The values for p1 and infection_average for the events "school" and "day_care_center" are from Schoeps et al. (2021).

Value

The a priori probability y.

References

Schoeps A et al. (2021) "Surveillance of SARS-CoV-2 transmission in educational institutions, August to December 2020, Germany". *Epidemiology and Infection* 149, E213: 1-9.

Examples

```
calculate_prior_infections(negative_persons = 23,
                           infected_persons = 2,
                           event = "school")
```

`generate_data_extended` *Generate data extended*

Description

Creates a dataframe suitable as input for `calculate_likelihood_negative_tests`.

Usage

```
generate_data_extended(
  M = 20,
  d = matrix(data = 1, nrow = 1, ncol = 2),
  S = c(12)
)
```

Arguments

`M`	The size of the group without infected, default is twenty.
`d`	A matrix with the test dates, default is matrix(data = 1, nrow = 1, ncol = 2).
`S`	A vector with the sizes of the subgroups, default is c(12).

Value

The dataframe.

`get_expected_total_infections` *Expected number of total symptomatic infections*

Description

Calculates the expected total number of symptomatic infections after a group event, based on the observed infections so far.

Usage

```
get_expected_total_infections(
  group_size,
  last_day_reported_infection,
  total_reported_infections,
  meanlog = 1.69,
  sdlog = 0.55
)
```

Arguments

group_size	integer, size of the group.
last_day_reported_infection	Number of days the last infection was reported after the event (0 = event day).
total_reported_infections	Number of reported symptomatic infections so far.
meanlog	Number, the parameter of mean from the log-normal distribution.
sdlog	Number, the parameter of sd from the log-normal distribution.

Details

meanlog and sdlog are the log-normal distribution parameters derived from the incubation period characteristics described in Xin et al. (2021). Note that the function often clearly overestimates the number of symptomatic infections if last_day_reported_infections is less than 3.

Value

The total number of expected symptomatic infections.

Examples

```
get_expected_total_infections(25, 5, 4)
```

---

get_incubation_day_distribution

Vector of day-specific probabilities of disease outbreak

Description

Creates a vector containing the probabilities of the disease outbreak for the days 1 to maxi after the infection.

Usage

```
get_incubation_day_distribution(max_days, meanlog = 1.69, sdlog = 0.55)
```

Arguments

max_days	Number, the maximum length of the incubation time, defined as number.
meanlog	Number, the parameter of mean from the log-normal distribution.
sdlog	Number, the parameter of sd from the log-normal distribution.

Details

meanlog and sdlog are the log-normal distribution parameters derived from the incubation period characteristics described in Xin et al. (2021).

Value

Vector of day-specific probabilities of disease outbreak.

References

Xin H, Wong JY, Murphy C et al. (2021) "The Incubation Period Distribution of Coronavirus Disease 2019: A Systematic Review and Meta-Analysis". *Clinical Infectious Diseases*, 73(12): 2344-2352.

Examples

```
get_incubation_day_distribution(10)
get_incubation_day_distribution(10, meanlog = 1.69, sdlog = 0.55)
```

---

get_infection_density *Dataframe with dates and probability of infection*

Description

Creates a dataframe containing probability of infection occurring at a particular date/time, given the symptom_begin_date.

Usage

```
get_infection_density(
  symptom_begin_date,
  max_incubation_days = 14,
  meanlog = 1.69,
  sdlog = 0.55
)
```

Arguments

symptom_begin_date	Date, when the person gets symptoms.
max_incubation_days	Number of incubation days.
meanlog	Number, the parameter of mean from the log-normal distribution.
sdlog	Number, the parameter of sd from the log-normal distribution.

Details

meanlog and sdlog are the log-normal distribution parameters derived from the incubation period characteristics described in Xin et al. (2021).

Value

Dataframe with dates and probability of infection.

References

Xin H, Wong JY, Murphy C et al. (2021) "The Incubation Period Distribution of Coronavirus Disease 2019: A Systematic Review and Meta-Analysis". *Clinical Infectious Diseases*, 73(12): 2344-2352.

Examples

```
get_infection_density(as.Date("2022-03-22"))
get_infection_density(as.Date("2022-03-22"), max_incubation_days = 14, meanlog = 1.69, sdlog = 0.55)
```

---

get_infectiousness_density

Dataframe with dates and infectiousness probability

Description

Creates a dataframe containing infectiousness at a particular date/time, given the symptom_begin_date.

Usage

```
get_infectiousness_density(
  symptom_begin_date,
  infectiousness_shift = 12.272481,
  max_infectious_days = 24,
  shape_infectiousness_gamma = 20.516508,
  rate_infectiousness_gamma = 1.592124
)
```

Arguments

`symptom_begin_date`
: Date, when the person gets symptoms.

`infectiousness_shift`
: Number of days with the largest contagions before the first symptoms.

`max_infectious_days`
: Number of the infectious days.

`shape_infectiousness_gamma`
: Number, the shape parameter for the gamma distribution.

`rate_infectiousness_gamma`
: Number, the rate parameter for the gamma distribution.

Details

infectiousness_shift, shape_infectiousness_gamma and rate_infectiousness_gamma are the distribution parameters for the infectious period from He et al. (2020).

Value

Dataframe with dates and infectiousness probability.

References

He, X et al. (2020) "Temporal dynamics in viral shedding and transmissibility of COVID-19". *Nature Medicine*, 26: 672–675.

Examples

```
get_infectiousness_density(as.Date("2022-03-22"))
get_infectiousness_density(as.Date("2022-03-22"), infectiousness_shift = 12.272481,
                        max_infectious_days = 24, shape_infectiousness_gamma = 20.516508,
                            rate_infectiousness_gamma = 1.592124)
```

---

get_misc_infection_density

Dataframe with dates and probability of infection

Description

Creates a dataframe containing probability of infection occurring at a particular dates/times, given the symptom_begin_dates and number_of_persons per date.

Usage

```
get_misc_infection_density(
  symptom_begin_dates,
  number_of_persons,
  max_incubation_days = 17,
  meanlog = 1.69,
  sdlog = 0.55
)
```

Arguments

`symptom_begin_dates`
: Dates, when the persons get symptoms.

`number_of_persons`
: Number of persons who get symptoms on each date.

`max_incubation_days`
: Number of incubation days.

meanlog	Number, the parameter of mean from the log-normal distribution.
sdlog	Number, the parameter of sd from the log-normal distribution.

Details

meanlog and sdlog are the log-normal distribution parameters derived from the incubation period characteristics described in Xin et al. (2021).

Value

Dataframe with dates and probability of infection.

References

Xin H, Wong JY, Murphy C et al. (2021) "The Incubation Period Distribution of Coronavirus Disease 2019: A Systematic Review and Meta-Analysis". *Clinical Infectious Diseases*, 73(12): 2344-2352.

Examples

```
symptom_begin_dates <- c(as.Date("2022-03-22"), as.Date("2022-03-26"))
number_of_persons <- c(3,1)
get_misc_infection_density(symptom_begin_dates, number_of_persons)
```

---

get_serial_interval_density

Dataframe with dates and contact symptom begin probability

Description

Creates a dataframe containing probability that a contact will start showing symptoms (serial interval) at a particular date/time, given the symptom_begin_date.

Usage

```
get_serial_interval_density(
  symptom_begin_date,
  max_serial_interval_days = 20,
  shape_serial = 2.154631545,
  rate_serial = 0.377343528
)
```

Arguments

`symptom_begin_date`	Date, when the index person got symptoms.
`max_serial_interval_days`	Number of serial interval days.
`shape_serial`	Number, the shape parameter for the gamma distribution.
`rate_serial`	Number, the rate parameter for the gamma distribution.

Details

shape_serial and rate_serial are the parameters of the gamma distribution for the serial interval derived from Najafi et al. (2020).

Value

Dataframe with dates and contact symptom begin probability.

References

Najafi F et al. (2020) "Serial interval and time-varying reproduction number estimation for COVID-19 in western Iran.". *New Microbes and New Infections*, 36: 100715.

Examples

```
get_serial_interval_density(as.Date("2022-03-22"))
get_serial_interval_density(as.Date("2022-03-22"), max_serial_interval_days = 20,
                            shape_serial = 2.15, rate_serial = 0.38)
```

`get_test_sensitivities` *Generate info*

Description

Creates a dataframe with day specific test sensitivity and specificity of PCR and Antigen tests.

Usage

```
get_test_sensitivities(df)
```

Arguments

`df`	Dataframe, this is a placeholder

Value

The dataframe.

Examples

```
get_test_sensitivities()
df <- data.frame(
             "PCR" = c(0, 0, 0, 0.04, 0.34, 0.64, 0.76, 0.79, 0.80, 0.79,
                       0.77, 0.74, 0.71, 0.67, 0.62, 0.58, 0.54, 0.49, 0.44,
                       0.40, 0.37, 0.33),
             "Antigen" = c(0, 0, 0, 0.03, 0.13, 0.40,  0.64, 0.69, 0.70, 0.69,
                           0.62, 0.52, 0.40, 0.29, 0.21, 0.17, 0.13, 0.11,
                           0.08, 0.07, 0.05, 0.04)
                )
get_test_sensitivities(df)
```

---

predict_future_infections

Prediction of future infections per day

Description

Predicts how many people are expected to develop symptoms on each day after the last reported infection after a group event.

Usage

```
predict_future_infections(
  last_day_reported_infection,
  total_reported_infections,
  total_expected_infections,
  meanlog = 1.69,
  sdlog = 0.55
)
```

Arguments

last_day_reported_infection	Number of days the last infection was reported after the event (0 = event day).
total_reported_infections	Number of reported symptomatic infections so far.
total_expected_infections	Number of expected symptomatic infections in total.
meanlog	Number, the parameter of mean from the log-normal distribution.
sdlog	Number, the parameter of sd from the log-normal distribution.

Details

meanlog and sdlog are the log-normal distribution parameters derived from the incubation period characteristics described in Xin et al. (2021).

Value

Vector with expected future infections per day after the event.

References

Xin H, Wong JY, Murphy C et al. (2021) "The Incubation Period Distribution of Coronavirus Disease 2019: A Systematic Review and Meta-Analysis". *Clinical Infectious Diseases*, 73(12): 2344-2352.

Examples

```
predict_future_infections(last_day_reported_infection = 3,
                          total_reported_infections = 5,
                          total_expected_infections = 15)
```

---

p_onePrimaryMore *One more primary a priori probability*

Description

Calculates the a priori probability y for one primary case more by using the current prior distribution and the prior distribution of one single primary case.

Usage

```
p_onePrimaryMore(yCurrent, y1)
```

Arguments

yCurrent	The current prior distribution.
y1	The prior distribution of one single primary case.

Value

The a priori probability y.

Index

calculate_likelihood_negative_tests, 2, *5*, *7*
calculate_likelihood_negative_tests_k, 3
calculate_posterior_no_infections, 4
calculate_prior_infections, *5*, 6

generate_data_extended, *5*, 7
get_expected_total_infections, 7
get_incubation_day_distribution, 8
get_infection_density, 9
get_infectiousness_density, 10
get_misc_infection_density, 11
get_serial_interval_density, 12
get_test_sensitivities, *5*, 13

p_onePrimaryMore, 15
predict_future_infections, 14