

# Package ‘iDOVE’

September 15, 2021

**Type** Package

**Title** Durability of Vaccine Efficacy Against SARS-CoV-2 Infection

**Version** 1.3

**Date** 2021-09-14

**Author** Yu Gu [aut],  
Shannon T. Holloway [aut, cre],  
Dan-Yu Lin [aut]

**Maintainer** Shannon T. Holloway <sthollow@ncsu.edu>

**Description** Implements a nonparametric maximum likelihood method for assessing potentially time-varying vaccine efficacy (VE) against SARS-CoV-2 infection under staggered enrollment and time-varying community transmission, allowing crossover of placebo volunteers to the vaccine arm.

Lin, D. Y., Gu, Y., Zeng, D., Janes, H. E., and Gilbert, P. B. (2021)  
<[doi:10.1093/cid/ciab630](https://doi.org/10.1093/cid/ciab630)>.

**License** GPL-2

**Encoding** UTF-8

**Suggests** rmarkdown

**VignetteBuilder** utils, knitr

**Imports** Rcpp (>= 1.0.4.6), methods, stats, graphics, knitr

**LinkingTo** Rcpp, RcppArmadillo

**RoxygenNote** 7.1.1

**Collate** 'VEplot.R' 'VEcal.R' 'postProcess.R' 'EMmeth.R'  
'RcppExports.R' 'verifyInputs.R' 'idove.R' 'idoveData.R'  
'intCens.R' 'plot.iDOVE.R' 'print.iDOVE.R'

**NeedsCompilation** yes

**Depends** R (>= 3.5.0)

**Repository** CRAN

**Date/Publication** 2021-09-15 09:00:05 UTC

## R topics documented:

idove . . . . .	2
idoveData . . . . .	5
intCens . . . . .	5
plot . . . . .	6
print . . . . .	7

<b>Index</b>	<b>9</b>
--------------	----------

---

idove	<i>Durability of Vaccine Efficacy Against Asymptomatic SARS-CoV-2 Infection</i>
-------	---

---

### Description

Assesses potentially time-varying vaccine efficacy (VE) against SARS-CoV-2 infection under staggered enrollment and time-varying community transmission, allowing crossover of placebo volunteers to the vaccine arm. The infection time data are interval-censored, and the log hazard ratio is assumed to be a piece-wise linear function of time.

### Usage

```
idove(
  formula,
  data,
  constantVE = FALSE,
  plots = TRUE,
  changePts = NULL,
  timePts = NULL,
  tol = 1e-04,
  maxit = 2000
)
```

### Arguments

formula	A formula object, with all of the time variables on the left hand side of a '~' operator and the covariates on the right. The time variables must be specified through the intCens() function. See ?intCens and Details for further information.
data	A data.frame object. The data.frame in which to interpret the variable names in formula. Must contain the entry time, the left interval time, the right interval time, the vaccination time, and any covariates. See Details.
constantVE	A logical object. If FALSE (default), VE is assumed to be potentially waning after the last change point; otherwise it is assumed to be constant after the last change point.
plots	A logical object. If TRUE (default), plots of the estimated VE in reducing attack rate, the estimated VE in reducing the hazard rate, and their 95% confidence intervals will be automatically generated. If FALSE, plots will not be generated.

<code>changePts</code>	An integer vector object or NULL. The potential change points (in days) of the piece-wise log-linear hazard ratio. See Details for further information. If NULL, the Akaike information criterion (AIC) will be used to select one change point from {28, 35, 42, 49, 56} (weeks 4, 5, 6, 7, 8).
<code>timePts</code>	An integer vector object or NULL. The endpoints (in days) of the time periods for which the VE in reducing the attack rate are to be estimated. The estimated VE in reducing the hazard rate at these endpoints are also returned. If NULL, a default sequence $t_1, 2t_1, 3t_1, \dots$ will be used, where $t_1$ is the first change point. The sequence ends at the maximum of the left and right ends of the time intervals from all participants. This input is ignored when <code>constantVE = TRUE</code> .
<code>tol</code>	A numeric scalar object. The convergence threshold for the EM algorithm. The default value is 0.0001.
<code>maxit</code>	A positive integer object. The maximum number of iterations for the EM algorithm. The default value is 2000.

### Details

The information required for an analysis is

**Entry Time:** The time when the participant enters the trial in whole units days.

**Left Interval Time:** The last examination time when the test is negative in whole units days.

**Right Interval Time:** The first examination time when the test is positive in whole units days. If the participant does not test positive during the trial, use NA or Inf.

**Vaccination Time:** The time when vaccination takes place in whole units days. If the participant is not vaccinated during the trial, use NA or Inf.

**Covariates:** Baseline covariates (e.g., priority group, age, ethnicity).

The covariates can include categorical variables, for which all other categories are compared to the first category. A model without covariates is also allowed.

Note that all of the time variables are measured from the start of the clinical trial and are specified in whole units of days. Though they need not be provided as integer, all non-NA and finite values must be able to be cast as integers without loss of information. For each individual, the `entry_time` and `left_time` should satisfy  $\text{entry\_time} \leq \text{left\_time}$ . For each individual that tests positive,  $\text{entry\_time} \leq \text{left\_time} \leq \text{right\_time}$ . For each individual that is vaccinated,  $\text{entry\_time} \leq \text{vaccination\_time}$ .

The general structure of the formula input is

```
intCens(entry_time, left_time, right_time, vaccination_time) ~ covariates
```

The left-hand side contains all of the time information. It must be specified through function `'intCens()'`. Specifically,

```
intCens(entry_time, left_time, right_time, vaccination_time)
```

If  $\text{entry\_time} > \text{left\_time}$ , or  $\text{left\_time} > \text{right\_time}$ , the case will be removed from the analysis and a message will be generated.

The special case of right-censored data is implemented by `dove2()` in the DOVE package available through CRAN.

**Value**

An S3 object of class `iDOVE` containing a list with elements

<code>call</code>	The unevaluated call.
<code>changePts</code>	The <code>changePts</code> of the analysis.
<code>covariates</code>	A matrix containing the estimated (log) hazard ratio of each covariate, together with the estimated standard error, the 95% confidence intervals, and the two-sided p-value for testing no covariates effect. NA if only an intercept is given as the right hand side in input formula.
<code>vaccine</code>	A list containing one or three elements, depending on the value of <code>constantVE</code> . If <code>constantVE = TRUE</code> , the only element is named 'VE' and is a vector containing the estimate of constant VE, its standard error estimate, and the 95% confidence interval. If <code>constantVE = FALSE</code> , three matrices are returned. The first matrix named 'VE_a' contains the estimates of the VE in reducing the attack rate at all time points given in <code>timePts</code> , together with the 95% confidence intervals. The second matrix named 'VE_h' contains the estimates of the VE in reducing the hazard rate at <code>timePts</code> . The third matrix named 'VE_period' contains the estimates of VE in reducing the attack rate over successive time periods according to <code>timePts</code> , together with the 95% confidence intervals.

**References**

Lin, D-Y, Gu, Y., Zeng, D., Janes, H. E., and Gilbert, P. B. (2021). Evaluating Vaccine Efficacy Against SARS-CoV-2 Infection. *Clinical Infectious Diseases*, ciab630, <https://doi.org/10.1093/cid/ciab630>

**Examples**

```
data(idoveData)

set.seed(1234)
smp <- sample(1L:nrow(x = idoveData), size = 250L)

# NOTE: This sample size is chosen for example only -- larger data sets
# should be used.
# See the vignette for a full analysis of the idoveData dataset

# Fit the model with default settings
idove(formula = intCens(entry.time, left.time, right.time, vaccine.time) ~ 1,
      data = idoveData[smp,])

# Specify Week 4 as the change point
# Assume a potentially waning VE after 4 weeks
# Estimate VE_a over 0-4, 4-16, 16-28, 28-40 weeks
idove(formula = intCens(entry.time, left.time, right.time, vaccine.time) ~ 1,
      data = idoveData[smp,],
      changePts = 4*7,
      timePts = c(4, 16, 28, 40)*7)

# Specify multiple change points at Weeks 4 and 8
# Assume a constant VE after 8 weeks
```

```
idove(formula = intCens(entry.time, left.time, right.time, vaccine.time) ~ 1,  
      data = idoveData[smp,],  
      changePts = c(4, 8)*7,  
      constantVE = TRUE)
```

---

idoveData

*Toy Dataset For Illustration*

---

### Description

This data set is provided for the purposes of illustrating the use of the software. It was simulated under a blinded, priority-tier dependent crossover design.

### Usage

```
data(idoveData)
```

### Format

idoveData is a data.frame containing 40,000 participants. The data.frame contains 6 columns,

**entry.time** The entry time in days.

**left.time** The left end of the time interval in days.

**right.time** The right end of the time interval in days.

**vaccine.time** The time of vaccination in days.

**priority** A composite baseline risk score taking values 1-5.

**sex** A binary indicator of sex (male/female).

---

intCens

*Specify Time Variables*

---

### Description

This function is used in the model statement of idove() to specify the entry time, left interval time, right interval time, and vaccination time.

### Usage

```
intCens(entry_time, left_time, right_time, vaccination_time)
```

**Arguments**

entry_time	The variable for the time when the participant enters the trial. Entry times must be integer (or be able to be cast as integer without loss of information), non-negative, and complete.
left_time	The variable for the last examination time when the test is negative. Left interval times must be integer (or be able to be cast as integer without loss of information), non-negative, and complete.
right_time	The variable for the first examination time when the test is positive. Right interval times must be integer (or be able to be cast as integer without loss of information), non-negative, and NA or Inf should be used if the participant never tested positive during the trial.
vaccination_time	The variable for the time when vaccination takes place. Vaccination times must be integer (or be able to be cast as integer without loss of information), non-negative, and NA or Inf should be used if the participant was not vaccinated during the trial.

**Details**

Times must obey the following relationships: (i) For all participants, `entry_time <= left_time`; (ii) For all participants that tested positive during the trial, `entry_time <= left_time <= right_time`; and (iii) For all participants that received vaccination, `entry_time <= vaccination_time`. If a case is found to violate one or more of these relationships, its `entry_time` is set to NA.

**Value**

This function is intended to be used only in the model statement of `idove()`. The result, a matrix, is used internally.

---

plot	<i>Plot Estimated Vaccine Efficacy</i>
------	--

---

**Description**

Generates plots of the estimated vaccine efficacy in reducing attack rate, the estimated vaccine efficacy in reducing the hazard rate, and their 95% confidence intervals.

**Usage**

```
## S3 method for class 'iDOVE'
plot(x, ...)
```

**Arguments**

x	An iDOVE object. The value object returned by <code>idove()</code> .
...	ignored

**Value**

No return value, called to produce graphical elements.

**Examples**

```
data(idoveData)

set.seed(1234)
smp <- sample(1L:nrow(x = idoveData), size = 250L)

# NOTE: This sample size is chosen for example only -- larger data sets
# should be used.
# See the vignette for a full analysis of the idoveData dataset

# Fit the model with default settings
result <- idove(formula = intCens(entry.time, left.time, right.time, vaccine.time) ~ 1,
                data = idoveData[smp,])

plot(x = result)
```

---

print

*Print the Primary Results of an idove() Analysis*

---

**Description**

Print the primary results of an idove() analysis.

**Usage**

```
## S3 method for class 'iDOVE'
print(x, ...)
```

**Arguments**

x	An iDOVE object. The value object returned by a call to idove()
...	ignored

**Value**

No return value, called to display key results.

**Examples**

```
data(idoveData)

set.seed(1234)
smp <- sample(1L:nrow(x = idoveData), size = 250L)

# NOTE: This sample size is chosen for example only -- larger data sets
# should be used.
# See the vignette for a full analysis of the idoveData dataset

# Fit the model with default settings
result <- idove(formula = intCens(entry.time, left.time, right.time, vaccine.time) ~ 1,
                data = idoveData[smp,])

print(x = result)
```



# Index

## \* datasets

idoveData, 5

idove, 2

idoveData, 5

intCens, 5

plot, 6

print, 7