Package 'plot3logit'

July 19, 2020

Type Package

Title Ternary Plots for Trinomial Regression Models

Version 2.2.0

Author Flavio Santi [cre, aut] (<https://orcid.org/0000-0002-2014-1981>), Maria Michela Dickson [aut] (<https://orcid.org/0000-0002-4307-0469>), Giuseppe Espa [aut] (<https://orcid.org/0000-0002-0331-3630>)

Maintainer Flavio Santi <flavio.santi@univr.it>

URL https://www.flaviosanti.it/software/plot3logit

BugReports https://github.com/f-santi/plot3logit

Description An implementation of the ternary plot for interpreting regression coefficients of trinomial regression models, as proposed in Santi, Dickson and Espa (2019) <doi:10.1080/00031305.2018.1442368>. Ternary plots can be drawn using either 'ggtern' package (based on 'ggplot2') or 'Ternary' package (based on standard graphics).

Depends R (>= 3.5), ggtern (>= 3.3.0), Ternary (>= 1.0.1)

Imports dplyr, ellipse, forcats, generics, ggplot2 (>= 3.3.0), graphics, grDevices, lifecycle, magrittr (>= 1.5), purrr, Rdpack, stats, tibble, tidyr, tidyselect, utils

Suggests knitr, MASS (>= 7.3-51), mlogit, nnet, rmarkdown, VGAM

License GPL (>=2)

LazyData true

NeedsCompilation no

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.0

Repository CRAN

Date 2020-07-18

VignetteBuilder knitr

RdMacros Rdpack, lifecycle

R topics documented:

`plot3logit-package` *Ternary Plots for Trinomial Regression Models*

Description

An implementation of the ternary plot for interpreting regression coefficients of trinomial regression models, as proposed in Santi et al. (2019).

Details

The package permits the covariate effects of trinomial regression models to be represented graphically by means of a ternary plot. The aim of the plots is helping the interpretation of regression coefficients in terms of the effects that a change in regressors' values has on the probability distribution of the dependent variable. Such changes may involve either a single regressor, or a group of them (composite changes), and the package permits both cases to be represented in a user-friendly way. Methodological details are illustrated and discussed in Santi et al. (2019).

The package can read the results of **both categorical and ordinal trinomial logit** regression fitted by various functions (see the next section) and creates a `field3logit` object which may be represented by means of functions `gg3logit()` and `stat_field3logit()`.

The `plot3logit` package inherits graphical classes and methods from the package `ggtern` (Hamilton and Ferry 2018) which, in turn, is based on the `ggplot2` package (Wickham 2017).

Graphical representation based on **standard graphics** is made available through the package `Ternary` (Smith 2017) by function `TernaryField()` and in particular by the method `plot` of `field3logit` class.

Since version 2.0.0, `plot3logit` permits one to draw also the confidence regions associated to the covariates effects. See the vignette of the package (type `vignette('plot3logit-overview')`) and the help of function `stat_conf3logit()` for some examples.

Compatibility

Function `field3logit()` can read trinomial regression estimates from the output of the following functions:

- `multinom` of package `nnet` (logit regression);
- `polr` of package `MASS` (ordinal logit regression);
- `mlogit` of package `mlogit` (logit regression);
- `vgam` of package `VGAM` (logit regression).

Moreover, explicit matrix of regression coefficients can be passed to `field3logit()`. See examples and function `field3logit()` for further details.

References

Hamilton NE, Ferry M (2018). "ggtern: Ternary Diagrams Using ggplot2." *Journal of Statistical Software, Code Snippets*, **87**(3), 1-17. doi: 10.18637/jss.v087.c03.

Santi F, Dickson MM, Espa G (2019). "A graphical tool for interpreting regression coefficients of trinomial logit models." *The American Statistician*, **73**(2), 200-207. doi: 10.1080/00031305.2018.1442368.

Smith MR (2017). "Ternary: An R Package for Creating Ternary Plots." *Zenodo*.

Wickham H (2017). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag.

See Also

`field3logit()`, `gg3logit()`, `TernaryField()`.

Examples

```
## Not run:
data(cross_1year)

# Read from "nnet::multinom"
library(nnet)
mod0 <- multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale')
gg3logit(field0) + stat_field3logit()

# Read from "MASS::polr"
library(MASS)
mydata <- cross_1year
mydata$finalgrade <- factor(mydata$finalgrade,
  c('Low', 'Average', 'High'), ordered = TRUE)
mod1 <- polr(finalgrade ~ gender + irregularity, data = mydata)
field1 <- field3logit(mod1, 'genderFemale')
gg3logit(field1) + stat_field3logit()

# Read from "mlogit::mlogit"
library(mlogit)
mydata <- mlogit.data(cross_1year, choice = 'employment_sit', shape = 'wide')
mod2 <- mlogit(employment_sit ~ 0 | gender + finalgrade, data = mydata)
field2 <- field3logit(mod2, 'genderFemale')
gg3logit(field2) + stat_field3logit()
```

```
# Read from matrix
M <- matrix(c(-2.05, 0.46, -2.46, 0.37), nrow = 2)
rownames(M) <- c('(Intercept)', 'genderFemale')
attr(M, 'labs') <- c('Employed', 'Unemployed', 'Trainee')
field3 <- field3logit(M, c(0, 1))
gg3logit(field3) + stat_field3logit()

## End(Not run)
```

`add_confregions` *Computes the confidence regions of covariate effects*

Description

Given the confidence level, it computes the confidence regions of the effects for each arrow of the `field3logit` or `multifield3logit` object given in input. If the `field3logit` or `multifield3logit` object already contains the confidence regions, they will be updated if the value of `conf` is different.

Usage

```
add_confregions(x, conf = 0.95, npoints = 100)
```

Arguments

`x`	an object of class `field3logit` or `multifield3logit`.
`conf`	confidence level of the regions.
`npoints`	number of points of the borders of the regions.

Value

Object of class `field3logit` or `multifield3logit` with updated confidence regions.

Examples

```
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale')
field0
add_confregions(field0)
```

`autoplot.field3logit` *Create a* `gg3logit` *plot with field and confidence regions*

Description

`autoplot()` creates a `gg3logit` plot and adds a field and its confidence regions. `autoplot()` is a wrapper for `gg3logit()` and `stat_3logit()`.

Usage

```
## S3 method for class 'field3logit'
autoplot(
  object,
  ...,
  mapping_field = aes(),
  mapping_conf = aes(),
  data = NULL,
  params_field = list(),
  params_conf = list(),
  show.legend = NA,
  conf = TRUE
)
```

Arguments

`object`	an object of class `field3logit` or `multifield3logit`.
`...`	other arguments passed to specific methods
`mapping_field`	aesthetic mappings passed to argument `mapping` of `stat_field3logit()` and `stat_conf3logit()`.
`mapping_conf`	aesthetic mappings passed to argument `mapping` of `stat_field3logit()` and `stat_conf3logit()`.
`data`	a `field3logit` or a `multifield3logit` object.
`params_field`	graphical parameters passed to argument `mapping` of `stat_field3logit()` and `stat_conf3logit()`.
`params_conf`	graphical parameters passed to argument `mapping` of `stat_field3logit()` and `stat_conf3logit()`.
`show.legend`	logical. Should this layer be included in the legends? `NA`, the default, includes if any aesthetics are mapped. `FALSE` never includes, and `TRUE` always includes. It can also be a named logical vector to finely select the aesthetics to display.
`conf`	if `TRUE` and if confidence regions are available, the layer of `stat_conf3logit()` is added, otherwise only a `gg3logit()` object with the layer of `stat_field3logit()` is returned.

See Also

Other gg functions: `gg3logit()`, `stat_3logit()`, `stat_conf3logit()`, `stat_field3logit()`

Examples

```
## Not run:
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale', conf = 0.95)

autoplot(field0)

## End(Not run)
```

`cross_1year` *Master's students' employment condition*

Description

`data.frame` with 3282 cross-sectional observations of 7 variables about employment condition of master's students one year after graduation. Data are used in Santi et al. (2019) and refer to students graduated at the University of Trento (Italy) between 2009 and 2013.

Format

`data.frame` with 3282 observations of 7 variables:

employment_sit: employment situation, a factor with three levels: *Employed*, *Unemployed*, *Trainee*.

gender: gender, a factor with two levels: *Male*, *Female*.

finalgrade: final grade degree, a factor with three levels: *Low*, *Average*, *High*.

duration: duration of studies, a factor with three levels: *Short*, *Average*, *Long*.

social_class: social class, a factor with five levels: *Working class*, *White-collar workers*, *Lower middle class*, *Upper middle class*, *Unclassified*.

irregularity: irregularity indicator of student's studies, a factor with three levels: *Low*, *Average*, *High*.

hsscore: high school final score, a numeric between 60 and 100.

References

Santi F, Dickson MM, Espa G (2019). "A graphical tool for interpreting regression coefficients of trinomial logit models." *The American Statistician*, **73**(2), 200-207. doi: 10.1080/00031305.2018.1442368.

deprecated-functions *List of deprecated and defunct functions*

Description

The following functions are deprecated and will no longer be updated. They may be removed in a future version of the package.

Deprecated functions

- `plot3logit()` (since version 2.0.0). Instead of `plot3logit()`, generate a `field3logit` object through `field3logit()` and then plot it through the method `plot()` (standard graphics based on package `Ternary`, through `autoplot()`, or through `gg3logit()` plus some stat_*3logit stats (graphics based on package `ggtern`).

field3logit *Computation of the vector field*

Description

`field3logit()` computes the vector field associated to a change in regressior values (which may involve more than one regressor) of a trinomial logit model either fitted by some multinomial regression function or explicitly specified.

The method `plot()` draws the ternary plot using standard graphics methods provided by package `Ternary`. See function `gg3logit()` for plotting through the package `ggtern` based on the grammar of graphics.

Methods `as.data.frame()`, `as_tibble()`, `fortify()` and `tidy()` permits the graphical information of a `field3logit` object to be exported in a standardised format (either a `data.frame` or a `tibble`).

Usage

```
field3logit(
  model,
  delta,
  label = "<empty>",
  p0 = NULL,
  alpha = NULL,
  vcov = NULL,
  nstreams = 8,
  narrows = Inf,
  edge = 0.01,
  conf = NA,
  npoints = 100
)

## S3 method for class 'field3logit'
print(x, ...)
```

```
## S3 method for class 'field3logit'
plot(x, ..., add = FALSE, length = 0.05)

## S3 method for class 'field3logit'
as_tibble(x, ..., wide = TRUE)

## S3 method for class 'field3logit'
as.data.frame(x, ..., wide = TRUE)

## S3 method for class 'field3logit'
fortify(model, data, ..., wide = TRUE)

## S3 method for class 'field3logit'
tidy(x, ..., wide = TRUE)

## S3 method for class 'field3logit'
coef(object, ...)

## S3 method for class 'field3logit'
vcov(object, ...)

## S3 method for class 'field3logit'
labels(object, ...)

## S3 replacement method for class 'field3logit'
labels(x) <- value
```

Arguments

`model`	either a fitted trinomial model or a matrix of regressor coefficients. See section *Compatibility* and examples of `plot3logit-package`.
`delta`	the change in the values of covariates to be represented. This could be either a `numeric` vector, the name of a covariate (passed either as a `character` or an `expression`), or a mathematical expression involving one or more than one covariates (passed either as a `character` or an `expression`). If a list is passed to `delta`, multiple fields are computed according to parameters passed as components of a 2-level list. See details and examples.
`label`	label to be used for identifying the field when multiple fields are plotted. See `multifield3logit()`.
`p0`	`list` of starting points (ternary coordinates) of the curves of the field. If not specified, `field3logit` automatically compute `nstreams` candidate points so that arrows are evenly distributed over the ternary plot area. See Examples.
`alpha`	`numeric` vector of length two where constants $\alpha^{(1)}$ and $\alpha^{(2)}$ are stored (only for ordinal models), as defined in Equation (7) of Santi et al. (2019).
`vcov`	(**only if** the model is read from a matrix, otherwise it will be ignored) variance-covariance matrix of parameter estimates. The elements of the variance-covariance matrix should be ordered according to the matrix of parameter estimates where the categories of the dependent variable are the slow index, whereas the covariates are the fast index.

nstreams	number of stream lines of the field to be computed. In case of ordinal models, this parameter is ineffective, as only one curve can be drawn. The parameter is ineffective also in case that argument p0 is set.
narrows	maximum number of arrows to be drawn per curve.
edge	minimum distance between each arrow (or point) and the edge of the ternary plot.
conf	confidence level of confidence regions to be computed **for each arrow** of the field.
npoints	number of points of the border to be computed **for each confidence region**.
x, object	object of class field3logit.
...	other arguments passed to or from other methods.
add	logical argument which specifies whether the field should be added to an existing plot (add = TRUE) or a new ternary plot should be drawn (add = FALSE).
length	length of the edges of the arrow head (in inches).
wide	it allows to choose whether as.data.frame, as_tibble, fortify and tidy should return a data.frame or a tibble in wide (default) or long form.
data	not used. Argument included only for interface compatibility with the generic fortify.
value	value to be assigned.

Details

Argument delta could be passed in one of the following formats:

- explicitly, as a numeric vector corresponding to the change $\Delta x \in \boldsymbol{R}^k$ in regressors values $x \in \boldsymbol{R}^k$;
- implicitly, as a character of the name of the covariate to be considered. In this case, vector $\Delta x \in \boldsymbol{R}^k$ is computed for a unit change of the specified covariate;
- as a mathematical expression (passed as an expression or a character object) involving one or more than one covariates. This allows one to analyse the effects of composite covariate changes through an easy-to-write and easy-to-read code without having to cope with explicit numerical specification of vector $\Delta x \in \boldsymbol{R}^k$.

See examples for comparing all three methods.

It is also possible to pass a list **to argument** delta**.** In such a case, the function field3logit is run once for every component of delta, and the set of generated field3logit objects is combined into a single object of class multifield3logit. The compoments of the list passed to delta must be named lists whose elements are used as arguments of each call of function field3logit, whereas the arguments specified in the parent call of field3logit are used as default values. It follows that arguments shared by all fields can be specified once in the parent call of field3logit, and only arguments which changes from field to field (such as delta and label) should be set in the lists making up the list passed to delta. See the last example in section Examples and the help of multifield3logit().

Value

S3 object of class field3logit structured as a named list or an object of class multifield3logit if delta is a list.

References

Santi F, Dickson MM, Espa G (2019). "A graphical tool for interpreting regression coefficients of trinomial logit models." *The American Statistician*, **73**(2), 200-207. doi: 10.1080/00031305.2018.1442368.

See Also

multifield3logit(), gg3logit(), autoplot().

Examples

```
data(cross_1year)

## Not run:
# Fitting the model
mod0 <- nnet::multinom(employment_sit ~ finalgrade + irregularity + hsscore,
  cross_1year)
mod0

# Assessing the effect of "finalgradeHigh" (explicit notation)
field0 <- field3logit(mod0, c(0, 0, 1, 0, 0, 0))
gg3logit(field0) + stat_field3logit()

# Assessing the effect of "finalgradeHigh" (implicit notation)
field0 <- field3logit(mod0, 'finalgradeHigh')
gg3logit(field0) + stat_field3logit()

# Assessing the combined effect of "finalgradeHigh" and
# a decrease of "hsscore" by 10
field0 <- field3logit(mod0, 'finalgradeHigh - 10 * hsscore')
gg3logit(field0) + stat_field3logit()

## End(Not run)

# Fitting the model
mod1 <- nnet::multinom(employment_sit ~ ., data = cross_1year)

# List passed to argument "delta" for generating "multifield3logit" objects
refpoint <- list(c(0.7, 0.15, 0.15))
depo <- list(
  list(delta = 'durationShort',  label = 'Short duration'),
  list(delta = 'durationLong',   label = 'Long duration'),
  list(delta = 'finalgradeHigh', label = 'High final grade'),
  list(delta = 'finalgradeLow',  label = 'Low final grade')
)
mfields <- field3logit(mod1, delta = depo, p0 = refpoint, narrows = 1)
mfields
```

---

gg3logit *Create a new gg3logit*

Description

gg3logit initialises a ggplot object through ggtern. If a fortified field3logit or a multifield3logit object is passed to argument data, the mandatory aesthetics of the ternary plot are automatically set.

Usage

```
gg3logit(data = NULL, mapping = aes(), ...)
```

Arguments

data	a field3logit object, a multifield3logit object, or a data.frame structured like a fortified field3logit or a multifield3logit object. If a field3logit or a multifield3logit is passed, none of the aesthetics mappings listed in Section "Aesthetic mappings" below has to be specified.
mapping	list of aesthetic mappings to use for plot. If a field3logit or a multifield3logit is passed to data, none of the aesthetics mappings listed in section *Aesthetic mappings* below has to be specified (if specified, they will be overwritten).
...	additional arguments passed through to ggtern.

Aesthetic mappings

The following aesthetics are required by at least one of the available stats. None of them should be specified if a field3logit or a multifield3logit is passed to the argument data of gg3logit(), stat_field3logit() or stat_conf3logit():

- x, y, z are required by:
  - stat_field3logit() as ternary coordinates of the starting points of the arrows;
  - stat_conf3logit() ternary coordinates of the points on the border of confidence regions;
- xend, yend, zend: required by stat_field3logit() as ternary coordinates of the ending points of the arrows;
- group: identifier of groups of graphical objects (arrows and their confidence regions);
- type: type of graphical object (arrows or confidence regions).

The following variables of a fortified field3logit or a multifield3logit object may be useful for defining other standard aesthetics (such as fill, colour, ...):

- label identifies a field through a label, thus it is useful for distinguishing the fields in a multifield3logit object.
- idarrow identifies each group of graphical objects (arrows and their confidence regions) *within* every field. Unlike variable group, idarrow is not a global identifier of graphical objects.

See Also

Other gg functions: autoplot.field3logit(), stat_3logit(), stat_conf3logit(), stat_field3logit()

Examples

```
## Not run:
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale')

gg3logit(field0) + stat_field3logit()

## End(Not run)
```

---

labels *Set the labels of a* field3logit *or a* multifield3logit *object*

Description

It enables the labels of an existing field3logit or a multifield3logit object to be set.

Usage

```
labels(x) <- value
```

Arguments

x	a field3logit or a multifield3logit object.
value	a character with the new label (or labels in case of a multifield3logit object).

---

multifield3logit *Multiple trilogit fields*

Description

Methods of S3 class multifield3logit handle multiple fields3logit objects simultaneously and permit new multifield3logit objects to be easily created by means of the sum operator "+".

Usage

```
multifield3logit(x, ...)

## S3 method for class 'field3logit'
x + y

## S3 method for class 'multifield3logit'
print(x, maxitems = 10, ...)

## S3 method for class 'multifield3logit'
plot(x, y = NULL, add = FALSE, col = NA, legend = TRUE, ...)

## S3 method for class 'multifield3logit'
```

```
as_tibble(x, ..., wide = TRUE)

## S3 method for class 'multifield3logit'
as.data.frame(x, ..., wide = TRUE)

## S3 method for class 'multifield3logit'
fortify(model, data, ..., wide = TRUE)

## S3 method for class 'multifield3logit'
tidy(x, ..., wide = TRUE)

## S3 method for class 'multifield3logit'
labels(object, ...)

## S3 replacement method for class 'multifield3logit'
labels(x) <- value

## S3 method for class 'multifield3logit'
x[i, drop = TRUE]

## S3 replacement method for class 'multifield3logit'
x[i] <- value
```

Arguments

x, y, model	object of class field3logit or multifield3logit.
...	other arguments passed to or from other methods.
maxitems	maximum number of items to be enumerated when an object of class multifield3logit is printed.
add	logical argument which specifies whether the field should be added to an existing plot (add = TRUE) or a new ternary plot should be drawn (add = FALSE).
col, legend	graphical parameters if Ternary package is used.
wide	it allows to choose whether as.data.frame, as_tibble, fortify and tidy should return a data.frame or a tibble in wide (default) or long form.
data	not used. Argument included only for interface compatibility with the generic fortify.
object	object of class field3logit.
value	value to be assigned.
i	index of the field3logit object to be selected.
drop	if TRUE, a field3logit object is returned if the subsetted multifield3logit object has length one.

Value

S3 object of class multifield3logit structured as a named list.

See Also

field3logit().

Examples

```
## Not run:
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ ., data = cross_1year)
mod0

field_Sdur <- field3logit(mod0, 'durationShort',
  label = 'Short duration')
field_Hfgr <- field3logit(mod0, 'finalgradeHigh',
  label = 'High final grade')

gg3logit(field_Sdur + field_Hfgr) +
  stat_field3logit()
  facet_wrap(~ label)

refpoint <- list(c(0.7, 0.15, 0.15))

field_Sdur <- field3logit(mod0, 'durationShort',
  label = 'Short duration', p0 = refpoint, narrows = 1)
field_Ldur <- field3logit(mod0, 'durationLong',
  label = 'Long duration', p0 = refpoint, narrows = 1)
field_Hfgr <- field3logit(mod0, 'finalgradeHigh',
  label = 'High final grade', p0 = refpoint, narrows = 1)
field_Lfgr <- field3logit(mod0, 'finalgradeLow',
  label = 'Low final grade', p0 = refpoint, narrows = 1)

mfields <- field_Sdur + field_Ldur  + field_Lfgr + field_Hfgr
mfields

gg3logit(mfields) +
  stat_field3logit(aes(colour = label)) +
  theme_zoom_L(0.45)

## End(Not run)
```

---

plot3logit-deprecated *Computation and representation of the vector field*

Description

Deprecated

This function is deprecated and may be soon removed from the package.

`plot3logit()` method draws the ternary plot using standard graphics methods provided by package `Ternary`. Use the method `plot()` of `field3logit` objects instead.

Usage

```
plot3logit(
  model,
  delta,
  label = "<empty>",
```

```
  p0 = NULL,
  alpha = NULL,
  ncurves = 8,
  narrows = Inf,
  edge = 0.01,
  ...
)
```

Arguments

model	either a fitted trinomial model or a matrix of regressor coefficients. See section *Compatibility* and examples of plot3logit-package.
delta	the change in the values of covariates to be represented. This could be either a numeric vector, the name of a covariate (passed either as a character or an expression), or a mathematical expression involving one or more than one covariates (passed either as a character or an expression). If a list is passed to delta, multiple fields are computed according to parameters passed as components of a 2-level list. See details and examples.
label	label to be used for identifying the field when multiple fields are plotted. See multifield3logit().
p0	list of starting points (ternary coordinates) of the curves of the field. If not specified, field3logit automatically compute nstreams candidate points so that arrows are evenly distributed over the ternary plot area. See Examples.
alpha	numeric vector of length two where constants $\alpha^{(1)}$ and $\alpha^{(2)}$ are stored (only for ordinal models), as defined in Equation (7) of Santi et al. (2019).
ncurves	number of curves of the field to be computed. In case of ordinal models, this parameter is ineffective, as only one curve can be drawn. The parameter is ineffective also in case that argument p0 is set.
narrows	maximum number of arrows to be drawn per curve.
edge	minimum distance between each arrow (or point) and the edge of the ternary plot.
...	other arguments passed to or from other methods.

Value

S3 object of class field3logit structured as a named list.

See Also

field3logit().

---

stat_3logit *Add a field and confidence regions to a* gg3logit *plot*

Description

stat_3logit() adds a field and its confidence regions to a gg3logit plot. stat_3logit() is a wrapper for stats stat_field3logit() and stat_conf3logit() which are jointly applied.

Usage

```
stat_3logit(
  mapping_field = aes(),
  mapping_conf = aes(),
  data = NULL,
  params_field = list(),
  params_conf = list(),
  show.legend = NA,
  inherit.aes = TRUE,
  conf = TRUE
)
```

Arguments

mapping_field, mapping_conf	aesthetic mappings passed to argument mapping of stat_field3logit() and stat_conf3logit().
data	a field3logit or a multifield3logit object.
params_field, params_conf	graphical parameters passed to argument mapping of stat_field3logit() and stat_conf3logit().
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes. It can also be a named logical vector to finely select the aesthetics to display.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders().
conf	if TRUE and if confidence regions are available, the layer of stat_conf3logit() is added, otherwise only the layer of stat_field3logit() is returned.

See Also

Other gg functions: autoplot.field3logit(), gg3logit(), stat_conf3logit(), stat_field3logit()

Examples

```
## Not run:
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale', conf = 0.95)

gg3logit(field0) + stat_3logit()
gg3logit(field0) + stat_3logit(conf = TRUE)

## End(Not run)
```

stat_conf3logit *Add the confidence regions of a field to a* gg3logit *plot*

Description

stat_conf3logit() adds a field to a gg3logit plot.

Usage

```
stat_conf3logit(
  mapping = aes(),
  data = NULL,
  geom = "polygon",
  position = "identity",
  show.legend = NA,
  inherit.aes = TRUE,
  ...
)
```

Arguments

mapping	list of aesthetic mappings to be used for plot. Mandatory aesthetics should not be specified if field3loglit or multifield3logit object is passed to data. See secion"Aesthetic mappings" of gg3logit() for details.
data	a field3logit or a multifield3logit object.
geom	The geometric object to use display the data
position	Position adjustment, either as a string, or the result of a call to a position adjustment function.
show.legend	logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes. It can also be a named logical vector to finely select the aesthetics to display.
inherit.aes	If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders().
...	Other arguments passed on to layer(). These are often aesthetics, used to set an aesthetic to a fixed value, like colour = "red" or size = 3. They may also be parameters to the paired geom/stat.

See Also

Other gg functions: autoplot.field3logit(), gg3logit(), stat_3logit(), stat_field3logit()

Examples

```
## Not run:
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale', conf = 0.95)
```

```
gg3logit(field0) + stat_conf3logit()
gg3logit(field0) + stat_field3logit() + stat_conf3logit()

## End(Not run)
```

`stat_field3logit` *Add a field to a* `gg3logit` *plot*

Description

`stat_field3logit()` adds a field to a `gg3logit` plot.

Usage

```
stat_field3logit(
  mapping = aes(),
  data = NULL,
  geom = "segment",
  position = "identity",
  show.legend = NA,
  inherit.aes = TRUE,
  arrow. = arrow(length = unit(0.2, "cm")),
  ...
)
```

Arguments

`mapping`	list of aesthetic mappings to be used for plot. Mandatory aesthetics should not be specified if `field3loglit` or `multifield3logit` object is passed to `data`. See secion"Aesthetic mappings" of `gg3logit()` for details.
`data`	a `field3logit` or a `multifield3logit` object.
`geom`	The geometric object to use display the data
`position`	Position adjustment, either as a string, or the result of a call to a position adjustment function.
`show.legend`	logical. Should this layer be included in the legends? `NA`, the default, includes if any aesthetics are mapped. `FALSE` never includes, and `TRUE` always includes. It can also be a named logical vector to finely select the aesthetics to display.
`inherit.aes`	If `FALSE`, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. `borders()`.
`arrow.`	specification for arrow heads, as created by function `arrow` of package `grid`.
`...`	Other arguments passed on to `layer()`. These are often aesthetics, used to set an aesthetic to a fixed value, like `colour = "red"` or `size = 3`. They may also be parameters to the paired geom/stat.

See Also

Other gg functions: `autoplot.field3logit()`, `gg3logit()`, `stat_3logit()`, `stat_conf3logit()`

Examples

```
## Not run:
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale', conf = 0.95)

gg3logit(field0) + stat_field3logit()
gg3logit(field0) + stat_field3logit() + stat_conf3logit()

## End(Not run)
```

`TernaryField` *Draw a field on an existing ternary plot*

Description

`TernaryField()` adds the vector field returned by `field3logit()` to an existing ternary plot generated by `Ternary::TernaryPlot()`.

Usage

```
TernaryField(
  field,
  ...,
  length = 0.05,
  conf = FALSE,
  npoints = 100,
  conf.args = list()
)
```

Arguments

`field`	object of class `field3logit` as returned by `field3logit()`.
`...`	other arguments passed to or from other methods.
`length`	length of the edges of the arrow head (in inches).
`conf`	if `FALSE` confidence regions are not drawn, even if available; if `TRUE` confidence regions are drawn only if available; if a `numeric` value is passed, confidence regions at the specified confidence level are computed (if not already available) and drawn.
`npoints`	number of points of the border to be computed **for each confidence region**.
`conf.args`	graphical parameters of confidence regions to be passed to `Ternary::TernaryPolygon()`.

Value

An object of class `field3logit` with confidence regions included, if computed within `TernaryField()`.

See Also

`field3logit()`.

Examples

```
library(nnet)
data(cross_1year)

mod0 <- nnet::multinom(employment_sit ~ gender + finalgrade, data = cross_1year)
field0 <- field3logit(mod0, 'genderFemale')

TernaryPlot()
TernaryField(field0)
```

`USvote2016` *Self-reported votes from VOTER Survey in 2016*

Description

Self-reported votes from 2016 VOTER Survey by Democracy Fund Voter Study Group (2017). Object `USvote2016` includes only few variables based on the result of the survey, which are publicly available online. See file `"data-raw/USvote2016_prepare.R"` in the GitHub repository `"f-santi/plot3logit"` (https://github.com/f-santi/plot3logit), where it is documented how the dataset `USvote2016` has been generated.

Format

`tibble` (`data.frame`) with 8000 observations of 7 variables:

idcode: voter identifier (integer).

vote: declared vote, afactor with three levels: "Clinton", "Trump", "Other".

race: race, a factor with six levels: "White", "Black", "Hispanic", "Asian", "Mixed", "Other".

educ: level of education, a factor with six levels: "No high school", "High school grad.", "Some college", "2-year college", "4-year college", "Post-grad".

gender: gender, a factor with four levels: "Male", "Female", "Skipped", "Not Asked".

birthyr: decades when the voter was born, a factor with six levels: "[1920,1940)", "[1940,1950)", "[1950,1960)", "[1960,1970)", "[1970,1980)", "[1980,2000)".

famincome: income (in USD) of voter's family, a factor with five levels: "[0; 30,000)", "[30,000; 60,000)", "[60,000; 100,000)", "[100,000; 150,000)", "[150,000; Inf)".

References

Democracy Fund Voter Study Group (2017). "Views of the electorate research survey, December 2016." https://www.voterstudygroup.org.

Index