

Package ‘callback’

July 13, 2024

Type Package

Title Computes Statistics from Discrimination Experimental Data

Version 0.1.1

Description In discrimination experiments candidates are sent on the same test (e.g. job, house rental) and one examines whether they receive the same outcome. The number of non negative answers are first examined in details looking for outcome differences. Then various answering rates and their exacts confidence intervals are computed. Last, exact and asymptotic discrimination tests are performed. Graphical methods are also available.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Imports stats, ggplot2

Suggests knitr, rmarkdown

BuildVignettes true

VignetteBuilder knitr

NeedsCompilation no

Maintainer Emmanuel Duguet <emmanuel.duguet@u-pec.fr>

Depends R (>= 3.5.0)

Author Emmanuel Duguet [aut, cre] (<<https://orcid.org/0000-0003-2214-4542>>),
David Gray [ctb] (<<https://orcid.org/0000-0003-3812-0350>>),
Loïc du Parquet [dct],
Yannick L'Horty [ctb] (<<https://orcid.org/0000-0001-6258-8565>>),
Rémi Le Gall [dct] (<<https://orcid.org/0000-0002-0557-2935>>),
Noam Léandri [dct],
Pascale Petit [ctb] (<<https://orcid.org/0000-0003-1269-476X>>),
Florent Sari [ctb] (<<https://orcid.org/0000-0002-9396-4030>>)

Repository CRAN

Date/Publication 2024-07-13 10:00:02 UTC

Contents

address1	2
callback	3
gender1	5
gender2	6
gender3	7
gender4	8
graph	9
graph.stat_ecs	9
graph.stat_glob	10
graph.stat_mcr	11
graph.stat_tcs	12
g_difp	13
g_ecs	14
g_prop	15
g_tcs	16
inter1	16
is.calc	17
labour1	18
labour2	19
mobility1	19
origin1	20
origin2	21
print.callback	23
print.stat_count	23
print.stat_glob	24
print.stat_mcr	25
print.stat_tcs	25
stat_colsums	26
stat_count	27
stat_ecs	28
stat_glob	30
stat_mcr	31
stat_signif	33
stat_tcs	34
Index	37

address1

Origin/Gender discrimination and strongly negative mediatic exposure (information technologist)

Description

The data were collected by the TEPP team (FR CNRS 2042) between December 2008 and January 2009. A candidate is defined by the variables ("gender", "origin", "mediaexp"). The variable "reput" creates a sample separation.

- **offer:** add number.
- **date:** date of the application.
- **sentorder:** order in which the application was sent.
- **callback:** TRUE if there was a non negative callback.
- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **origin:** all the candidates are French, the origin is suggested by the name, F = France, M = Morocco.
- **gender:** Woman or Man.
- **city:** candidate location.
- **reput:** reputation of the city, P = privileged, U = Unprivileged.
- **mediaexp:** strong negative mediatic exposure, Yes or No.

Usage

```
data(address1)
```

Format

A data frame with 3684 rows and 11 variables

References

Duguet E., Gray D., L'Horty Y., Du Parquet L, Petit P., 2020. Labor market effects of urban riots:an experimental assessment. Papers in Regional Science, 99:787-806.

callback

Data formatting

Description

Creates the data set used in the callback package.

Usage

```
callback(data, cluster, candid, callback, comp = "ref")
```

Arguments

data	A data frame.
cluster	A variable name, identifying the test (e.g., a job offer number).
candid	A list of factor names defining the candidates (e.g., gender, origin).
callback	A Boolean variable, equal to TRUE for non negative callbacks.
comp	An option, equal to "all" or "ref" (the default). "ref" give the comparisons with the reference candidate, and "all" the pairwise comparisons.

Value

A 'callback' object containing the formatted data set (`fds`), the averaged formatted data set (`afds`, for multiple tests), the list of the paired formatted data sets (`pfds`), the offer-level formatted data set (`ofds`), the names of the candidate variables (`candid`) and the name of the callback variable (`callback`).

`fds` contains the following variables:

- **cluster**: the cluster variable.
- **candid**: the concatenation of the candidate variables.
- **callback**: the callback variable.

`afds` contains the following variables:

- **cluster**: the cluster variable.
- **candid**: the concatenation of the candidate variables.
- **callback**: the aggregated callback dummy.

`pfds` data frames containing the following variables:

- **callback1**: TRUE if candidate 1 had a callback.
- **callback2**: TRUE if candidate 2 had a callback.
- **c00**: TRUE if neither candidate was called back.
- **c10**: TRUE if candidate 1 was the only one called back.
- **c01**: TRUE if candidate 2 was the only one called back.
- **c11**: TRUE if both candidates were called back.
- **callback**: TRUE if either candidate was called back.
- **calldif**: callback difference.

`ofds` contains variables corresponding to the candidates' codes and the following variables:

- **cluster**: the cluster variable.
- **ncall**: the number of callback for the current offer.

Author(s)

Emmanuel Duguet

Examples

```
data(inter1)
m <- callback(data=subset(origin1,reput=="U"),cluster="offer",candid="origin",
  callback="callback",comp = "ref")
str(m)
```

gender1	<i>Gender/Maternity discrimination (commercial and administrative jobs in the financial sector)</i>
---------	---

Description

The data were collected in January-March 2002 by Pascale Petit for her PhD thesis (University of Paris I-Panthéon-Sorbonne, 2004). A candidate is defined by the variables ("gender","age","child").

- **offer:** add number.
- **gender:** Woman or Man.
- **age:** 25 or 37 years old.
- **child:** number of children, 0 or 3.
- **educ:** education, BAC = Baccalauréat = A-level, BTS = 2 years of vocational training after the A-level.
- **qual:** qualification required by the offer, Administrative or Commercial.
- **date:** January 2002, February 2002 or March 2002.
- **cv:** CV template, A or B.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **popp:** promotion opportunity, Yes or No.
- **train:** training included, Yes or No.
- **negow:** negotiable wage, Yes or No.
- **incent:** wage depending on output, Yes or No.
- **callback:** TRUE if there was a non negative callback.

Usage

```
data(gender1)
```

Format

A data frame with 942 rows and 14 variables

References

Duguet E., Petit P., 2005. Hiring discrimination in the French financial sector:an econometric analysis on field experiment data. *Annals of Economics and Statistics*, 78:79-102.

Petit P., 2007. The effects of age and family constraints on gender hiring discrimination:A field experiment in the French financial sector. *Labor Economics*, 14:371-391.

gender2

Gender/Maternity discrimination (electricians)

Description

The data were collected by the TEPP team (FR CNRS 2042) between February and July 2015. A candidate is defined by the variables ("gender", "educ").

- **offer:** add number.
- **gender:** Woman or Man.
- **age:** 23 or 24 years old.
- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **educ:** education, CAP = vocational training certificate before the A level, MAF = CAP + "One of the best French apprentice", OLY = CAP + participation to the Worldskills Competition.
- **zip:** ZIP code.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **recgender:** gender of the recruiter.
- **cv:** CV template, A or B.
- **ansmode:** answering channel, email or ordinary mail ("omail").
- **sentorder:** order in which the application was sent.
- **hours:** weekly work time.
- **callback:** TRUE if there was a non negative callback.
- **date:** between February and July 2015.

Usage

`data(gender2)`

Format

A data frame with 564 rows and 15 variables

References

Duguet E., du Parquet L., Petit P. (2022). Extracting the discrimination components from the callback rates". TEPP Working Paper 2022-15.

Duguet, E., du Parquet, L. & Petit, P. (2022). Révéler les composantes de la discrimination à partir des taux de rappel. *Revue française d'économie*, XXXVII, 233-268.

gender3

Gender/Maternity discrimination (masons)

Description

The data were collected by the TEPP team (FR CNRS 2042) between February and July 2015. A candidate is defined by the variables ("gender", "educ").

- **offer:** add number.
- **gender:** Woman or Man.
- **age:** 23 or 24 years old.
- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **educ:** education, CAP = vocational training certificate before the A level, MAF = CAP + "One of the best French apprentice", OLY = CAP + participation to the Worldskills Competition.
- **zip:** ZIP code.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **recgender:** gender of the recruiter.
- **cv:** CV template, A or B.
- **ansmode:** answering channel, email or ordinary mail ("omail").
- **sentorder:** order in which the application was sent.
- **hours:** weekly work time.
- **callback:** TRUE if there was a non negative callback.
- **date:** between February and July 2015.

Usage

`data(gender3)`

Format

A data frame with 532 rows and 15 variables

References

Duguet E., du Parquet L., Petit P. (2022). Extracting the discrimination components from the call-back rates". TEPP Working Paper 2022-15.

Duguet, E., du Parquet, L. & Petit, P. (2022). Révéler les composantes de la discrimination à partir des taux de rappel. *Revue française d'économie*, XXXVII, 233-268.

gender4

Gender/Maternity discrimination (plumbers)

Description

The data were collected by the TEPP team (FR CNRS 2042) between February and July 2015. A candidate is defined by the variables ("gender", "educ").

- **offer:** add number.
- **gender:** Woman or Man.
- **age:** 23 or 24 years old.
- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **educ:** education, CAP = vocational training certificate before the A level, MAF = CAP + "One of the best French apprentice", OLY = CAP + participation to the Worldskills Competition.
- **zip:** ZIP code.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **recgender:** gender of the recruiter.
- **cv:** CV template, A or B.
- **ansmode:** answering channel, email or ordinary mail ("omail").
- **sentorder:** order in which the application was sent.
- **hours:** weekly work time.
- **callback:** TRUE if there was a non negative callback.
- **date:** between February and July 2015.

Usage

`data(gender4)`

Format

A data frame with 1152 rows and 15 variables

References

Duguet E., du Parquet L., Petit P. (2022). Extracting the discrimination components from the callback rates". TEPP Working Paper 2022-15.

Duguet, E., du Parquet, L. & Petit, P. (2022). Révéler les composantes de la discrimination à partir des taux de rappel. *Revue française d'économie*, XXXVII, 233-268.

graph	<i>Generic plot function</i>
-------	------------------------------

Description

Generic plot function

Usage

```
graph(data, method, col, ...)
```

Arguments

data	A <code>stat_glob</code> , <code>stat_mcr</code> or <code>stat_ecs</code> object.
method	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
col	A list of colors.
...	further arguments passed to or from other methods.

graph.stat_ecs	<i>Exclusive callback shares plots</i>
----------------	--

Description

Exclusive callback shares plots

Usage

```
## S3 method for class 'stat_ecs'
graph(
  data = NULL,
  method = "wilson",
  col = c("#F564E3", "#7CAE00", "#00BFC4"),
  dif = TRUE,
  ...
)
```

Arguments

data	a <code>stat_ecs</code> object.
method	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
col	A list of colors (three colors needed).
dif	TRUE for the difference in proportions (the default), FALSE for a comparison of confidence intervals
...	further arguments passed to or from other methods.

Value

a ggplot2 object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)
m <- callback(labour1,"offer","hist","callback","all")
s <- stat_tcs(m)
graph(data=s,method="student")
```

graph.stat_glob

Global callback rates plot

Description

Global callback rates plot

Usage

```
## S3 method for class 'stat_glob'
graph(
  data = NULL,
  method = "wilson",
  col = c("#F8766D", "#00BA38", "#619CFF"),
  ...
)
```

Arguments

data	A stat_glob object.
method	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
col	A list of colors (one colour needed).
...	further arguments passed to or from other methods.

Value

A ggplot2 object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)
m <- callback(labour1,"offer","hist","callback","all")
s <- stat_glob(m)
graph(s)
```

`graph.stat_mcr`*Matched callback rates plots*

Description

Matched callback rates plots

Usage

```
## S3 method for class 'stat_mcr'
graph(
  data = NULL,
  method = "wilson",
  col = c("#F564E3", "#7CAE00", "#00BFC4"),
  dif = TRUE,
  ...
)
```

Arguments

<code>data</code>	a <code>stat_mcr</code> object.
<code>method</code>	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
<code>col</code>	A list of colors (three colors needed).
<code>dif</code>	TRUE for the difference in proportions (the default), FALSE for a comparison of confidence intervals
<code>...</code>	further arguments passed to or from other methods.

Value

a `ggplot2` object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)
m <- callback(labour1,"offer","hist","callback","all")
s <- stat_mcr(m)
graph(data=s,method="student")
```

graph.stat_tcs

Total callback shares plots

Description

Total callback shares plots

Usage

```
## S3 method for class 'stat_tcs'
graph(
  data = NULL,
  method = "wilson",
  col = c("#F564E3", "#7CAE00", "#00BFC4"),
  dif = TRUE,
  ...
)
```

Arguments

data	a stat_tcs object.
method	the type of confidence interval: "wilson" for Wilson (the default), "student" for Student.
col	A list of colors (three colors needed).
dif	TRUE for the difference in proportions (the default), FALSE for a comparison of confidence intervals
...	further arguments passed to or from other methods.

Value

a ggplot2 object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)
m <- callback(labour1,"offer","hist","callback","all")
s <- stat_tcs(m)
graph(data=s,method="student")
```

g_difp

Difference of proportions plot

Description

Difference of proportions plot

Usage

```
g_difp(
  data = NULL,
  method = "wilson",
  col = c("#619CFF", "#619CFF", "#F564E3"),
  ...
)
```

Arguments

data	A stat_mcr, stat_tcs or stat_ecs object.
method	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
col	A list of colors (two colors needed).
...	further arguments passed to or from other methods.

Value

A ggplot2 object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)
m <- callback(labour1,"offer","hist","callback","all")
s <- stat_mcr(m)
g_difp(data=s,method="student")
```

`g_ecs`*Exclusive callback shares plot*

Description

Exclusive callback shares plot

Usage

```
g_ecs(  
  data = NULL,  
  method = "wilson",  
  col = c("#F564E3", "#7CAE00", "#00BFC4"),  
  ...  
)
```

Arguments

<code>data</code>	a <code>stat_ecs</code> object.
<code>method</code>	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
<code>col</code>	A list of colors (three colors needed).
<code>...</code>	further arguments passed to or from other methods.

Value

A `ggplot2` object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)  
m <- callback(labour1,"offer","hist","callback","all")  
s <- stat_ecs(m)  
g_ecs(s,method="cp")
```

g_prop *Proportions' comparison plot*

Description

Proportions' comparison plot

Usage

```
g_prop(  
  data = NULL,  
  method = "wilson",  
  col = c("#F564E3", "#7CAE00", "#00BFC4"),  
  ...  
)
```

Arguments

data	A stat_mcr object.
method	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
col	A list of colors (two colors needed).
...	further arguments passed to or from other methods.

Value

A ggplot2 object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)  
m <- callback(labour1,"offer","hist","callback","all")  
s <- stat_mcr(m)  
g_prop(data=s,method="student")
```

`g_tcs` *Total callback shares plot*

Description

Total callback shares plot

Usage

```
g_tcs(data = NULL, col = c("gray75", "#F564E3", "#7CAE00"), ...)
```

Arguments

`data` a `stat_tcs` object.
`col` A list of colors (three colors needed).
... further arguments passed to or from other methods.

Value

A `ggplot2` object

Author(s)

Emmanuel Duguet

Examples

```
data(origin1)
m <- callback(labour1, "offer", "hist", "callback", "all")
s <- stat_tcs(m)
g_tcs(s)
```

`inter1` *Gender/Origin discrimination (software developer)*

Description

The data were collected by the TEPP team (FR CNRS 2042) between February and April 2009. A candidate is defined by the variables ("gender", "origin").

- **offer:** add number.
- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **gender:** Woman or Man.

- **origin:** all candidates are French, the origin is suggested by the name. F = French, S = Senegal, M = Morocco, V = Vietnam.
- **date:** date of the application.
- **sentorder:** order in which the application was sent.
- **callback:** TRUE if there was a non negative callback.
- **ansorder:** order in which the answer was received when positive, 9 otherwise.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **paris:** job located inside Paris, Yes or No.

Usage

```
data(inter1)
```

Format

A data frame with 2480 rows and 11 variables

References

Petit P., Duguet E., L'Horty Y., Du Parquet L., Sari F., 2013. Discrimination à l'embauche : les effets du genre et de l'origine se cumulent-ils systématiquement ? *Economie et Statistique*, 464-465-466:141-153.

Duguet E., Du Parquet L., L'Horty Y., Petit P., 2015. New Evidence of Ethnic and Gender discriminations in the French Labor Market using experimental data: A ranking extension of responses from correspondence tests. *Annals of Economics and Statistics*, 117-118:21-39.

is.calc

Computational compatibility

Description

Computational compatibility

Usage

```
is.calc(x)
```

Arguments

x A variable name

Value

A logical variable indicating whether the variable can be used for computations (TRUE) or not (FALSE).

Examples

```
data(origin1)
is.calc(origin1$cartime)
is.calc(origin1$callback)
is.calc(origin1$origin)
```

labour1

Labour market history discrimination (accountants)

Description

The data were collected by the TEPP team (FR CNRS 2042) between February and April 2015. A candidate is defined by the variable "hist".

- **offer:** add number.
- **date:** date of the application.
- **sentorder:** order in which the application was sent.
- **callback:** TRUE if there was a non negative callback.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **paris:** job located inside Paris, Yes or No.
- **hist:** history in the labour market, LTC = Long term contract, LTU = Long term unemployment, STU = Short term unemployment, STC = Short term contract, PTC = Part time contract.

Usage

```
data(labour1)
```

Format

A data frame with 1475 rows and 7 variables

References

Duguet E., Le Gall R., L'Horty Y., Petit P., 2018. How does labour market history influence the access to hiring interviews? *International Journal of Manpower*, 39(4), 519-533.

labour2

Labour market history discrimination (sales assistant)

Description

The data were collected by the TEPP team (FR CNRS 2042) between January and April 2015. A candidate is defined by the variable "hist".

- **offer:** add number.
- **date:** date of the application.
- **sentorder:** order in which the application was sent.
- **callback:** TRUE if there was a non negative callback.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **paris:** job located inside Paris, Yes or No.
- **hist:** history in the labour market, LTC = Long term contract, LTU = Long term unemployment, STU = Short term unemployment, STC = Short term contract, PTC = Part time contract.

Usage

```
data(labour2)
```

Format

A data frame with 1470 rows and 7 variables

References

Duguet E., Le Gall R., L'Horty Y., Petit P., 2018. How does labour market history influence the access to hiring interviews? *International Journal of Manpower*, 39(4), 519-533.

mobility1

Gender discrimination and mobility (management controller)

Description

The data were collected by the TEPP team (FR CNRS 2042) between October 2008 and March 2009. A candidate is defined by the variables ("gender", "licenses").

- **offer:** add number.
- **date:** date of the application.
- **sentorder:** order in which the application was sent.
- **callback:** TRUE if there was a non negative callback.

- **ansorder:** order in which the answer was received when positive, 5 otherwise.
- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **gender:** Woman or Man.
- **licenses:** both moto and car licenses, Yes or No.
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **paris:** job located inside Paris, Yes or No.
- **cv:** CV template, A or B.

Usage

```
data(mobility1)
```

Format

A data frame with 1200 rows and 12 variables

References

Duguet E., du Parquet L., L'Horty Y., Petit P., 2018. Counterproductive hiring discrimination against women:evidence from a French correspondence test. *International Journal of Manpower*, 39(1):37-50.

origin1

Origin discrimination (accountants)

Description

The data were collected by the TEPP team (FR CNRS 2042) between September and November 2006. A candidate is defined by the variables ("nation","lnation", "fnation"). The variables "educ" and "reput" create sample separations.

- **offer:** add number.
- **date:** September 2006, October 2006 or November 2006.
- **callback:** TRUE if there was a non negative callback.
- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **educ:** education, BAC = Baccalauréat = A-level, BTS = 2 years of vocational training after the A-level.
- **cartime:** commuting time by car (minutes).
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).

- **paris:** job located inside Paris, Yes or No.
- **nation:** nationality, M = Moroccan, F = French.
- **fnation:** first name sounding, M = Moroccan, F = French.
- **lnation:** last name sounding, M = Moroccan, F = French.
- **origin:** summary variable made from nation, lnation and fnation. Example:FMM = French nationality, Moroccan family name and French first name.
- **city:** candidate location.
- **reput:** reputation of the city, P = privileged, U = Unprivileged.
- **cv:** CV template, A or B.
- **natemp:** add obtained from the national employment agency (ANPE at the time of the test, France Travail today).
- **subsid:** the firm is a subsidiary of a large corporation, Yes or No.
- **ansmode:** answering channel, email or ordinary mail for all the applications to the same add. M = email, P = postage prepaid envelope, R = Marianne stamp, C = Cubitus stamp (comics character).
- **email:** answered by email by all the candidates, 1 = yes, 0 = No.

Usage

```
data(origin1)
```

Format

A data frame with 1097 rows and 20 variables

References

Duguet E., Leandri N., L'Horty Y., Petit P., 2010. Are young French jobseekers of ethnic immigrant origin discriminated against? A controlled experiment in the Paris area. *Annals of Economics and Statistics*, 99-100: 187-215.

origin2

Origin discrimination (waiters)

Description

The data were collected by the TEPP team (FR CNRS 2042) between September and November 2006. A candidate is defined by the variables ("nation", "lnation", "fnation"). The variables "educ" and "reput" create sample separations.

- **offer:** add number.
- **date:** September 2006, October 2006 or November 2006.
- **callback:** TRUE if there was a non negative callback.

- **fname:** first name (forename).
- **lname:** last name (family name, surname).
- **educ:** education, BAC = Baccalauréat = A-level, BTS = 2 years of vocational training after the A-level.
- **cartime:** commuting time by car (minutes).
- **cont:** length of labour contract, STC = short term contract (≤ 1 year), LTC = long term contract (> 1 year with no ending date).
- **paris:** job located inside Paris, Yes or No.
- **nation:** nationality, M = Moroccan, F = French.
- **fnation:** first name sounding, M = Moroccan, F = French.
- **lnation:** last name sounding, M = Moroccan, F = French.
- **origin:** summary variable made from nation, lnation and fnation. Example:FMF = French nationality, Moroccan family name and French first name.
- **city:** candidate location.
- **reput:** reputation of the city, P = privileged, U = Unprivileged.
- **cv:** CV template, A or B.
- **natemp:** add obtained from the national employment agency (ANPE at the time of the test, France Travail today).
- **subsid:** the firm is a subsidiary of a large corporation, Yes or No.
- **ansmode:** answering channel, email or ordinary mail for all the applications to the same add. M = email, P = postage prepaid envelope, R = Marianne stamp, C = Cubitus stamp (comics character).
- **email:** answered by email by all the candidates, 1 = yes, 0 = No.

Usage

```
data(origin2)
```

Format

A data frame with 936 rows and 20 variables

References

Petit P., Duguet E., L'Horty Y., 2015. Discrimination résidentielle et origine ethnique:une étude expérimentale sur les serveurs en Ile de France. *Economie et Prevision*, 206-207:55-69.

print.callback	<i>Prints the structure of the experiment</i>
----------------	---

Description

Computes the number of tests available for each pair of candidates

Usage

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

Arguments

x a stat_count object.
... further arguments passed to or from other methods.

Value

Printed output.

Examples

```
data(labour1)  
x <- callback(data=labour1,cluster="offer",candid="hist",callback="callback",  
comp = "all")  
print(x)
```

print.stat_count	<i>Print the callback counts analysis</i>
------------------	---

Description

Computes the callback numbers for each candidate.

Usage

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

Arguments

x a stat_count object.
... further arguments passed to or from other methods.

Value

Printed output.

Examples

```
data(labour1)
print(stat_count(callback(labour1, "offer", "hist", "callback",
comp = "all")))
```

print.stat_glob	<i>Prints the global callback statistics</i>
-----------------	--

Description

Prints the global callback numbers and proportions for each candidate.

Usage

```
## S3 method for class 'stat_glob'
print(x = NULL, method = "wilson", digits = 3, ...)
```

Arguments

x	a stat_glob object.
method	the type of confidence interval: "cp" for Clopper-Pearson, "wilson" for Wilson (the default), "student" for Student.
digits	minimal number of significant digits
...	further arguments passed to or from other methods.

Value

Printed output.

Examples

```
data(labour1)
print(stat_glob(callback(labour1, "offer", "hist", "callback",
comp = "all")))
```

print.stat_mcr *Prints the callback proportions analysis*

Description

Prints the statistics about the callback rates

Usage

```
## S3 method for class 'stat_mcr'  
print(x, digits = 3, ...)
```

Arguments

x a stat_mcr object.
digits minimal number of significant digits.
... further arguments passed to or from other methods.

Value

Printed output.

Examples

```
data(labour1)  
x <- callback(labour1, "offer", "hist", "callback", comp = "all")  
print(stat_mcr(x, level=0.90))
```

print.stat_tcs *Prints the proportions of the total callback shares*

Description

Prints the statistics from stat_tcs() output
Prints the statistics from stat_ecs() output

Usage

```
## S3 method for class 'stat_tcs'  
print(x, digits = 3, ...)  
  
## S3 method for class 'stat_ecs'  
print(x, digits = 3, ...)
```

Arguments

x a stat_ecs object.
digits minimal number of significant digits.
... further arguments passed to or from other methods.

Value

Printed output.
Printed output.

Examples

```
data(labour1)
x <- callback(labour1, "offer", "hist", "callback", comp = "all")
print(stat_tcs(x, level=0.90))
data(labour1)
x <- callback(labour1, "offer", "hist", "callback", comp = "all")
print(stat_ecs(x, level=0.90))
```

stat_colsums

Sums the numeric or logical columns in a data frame.

Description

Sums the numeric or logical columns in a data frame.

Usage

```
stat_colsums(x)
```

Arguments

x A data frame

Value

A data frame with the column sums, under their original names.

Examples

```
data(labour1)
s <- callback(labour1, "offer", "hist", "callback", "all")
stat_colsums(s$pfds[["LTC vs STC"]])
```

stat_count	<i>Callback counts on paired data</i>
------------	---------------------------------------

Description

Computes the callback count statistics from the paired data sets.

Usage

```
stat_count(x)
```

Arguments

x A callback object.

Value

A list with class "stat_count" containing two data frames: counts and props.
counts: a data frame with the callback counts.

- **tests:** number of tests.
- **callback:** number of tests with at least one callback for either candidate.
- **callback1:** number of callbacks for candidate 1.
- **callback2:** number of callbacks for candidate 2.
- **c00:** number of test without a callback.
- **c10:** number of tests with callbacks for candidate 1 only.
- **c01:** number of tests with callbacks for candidate 2 only.
- **c11:** number of tests with callbacks for both candidates
- **calldif:** difference in callback numbers.

props: a data frame with the following variables.

- **p_callback:** callback/tests.
- **p_cand1:** callback1/tests.
- **p_cand2:** callback2/tests.
- **p_c00:** c00/tests.
- **p_c10:** c10/tests.
- **p_c01:** c01/tests.
- **p_c11:** c11/tests.
- **p_cand_dif:** calldif/tests.

Author(s)

Emmanuel Duguet

Examples

```
data(labour2)
x <- callback(data=labour2,cluster="offer",candid="hist",callback="callback")
stat_count(x)
```

stat_ecs

*Exclusive callback shares***Description**

Computes the callback shares and their confidence intervals. The analysis is restricted to the tests with discrimination cases.

Usage

```
stat_ecs(x, level = 0.95)
```

Arguments

x a callback object.
level a number, containing the level of the confidence intervals (0.95 by default).

Value

A list with class "stat_ecs" containing 8 components : level, shares, cp, wilson, student, t.student, t.pearson and t.fisher.

level: the level of the confidence intervals.

shares: a data frame containing the following variables.

- **disc**: number of discrimination cases.
- **c10**: number of tests with the 1st candidate preferred (2nd candidate discriminated against).
- **c01**: number of tests with the 2nd candidate preferred (1st candidate discriminated against).
- **cdif**: net discrimination $c10 - c01$.
- **p_cand1**: 1st candidate callback share ($c10/disc$).
- **p_cand2**: 2nd candidate callback share ($c01/disc$).
- **p_cand_dif**: $p_cand1 - 1/2$.

cp: a data frame containing the Clopper-Pearson confidence intervals, from `binom.test()`, and the p-value of the Fisher test of independence between the candidate type and the callback variable, from `fisher.test()`.

- **inf_p_cand1**: 1st candidate callback rate, lower bound.
- **sup_p_cand1**: 1st candidate callback rate, upper bound.
- **inf_p_cand2**: 2nd candidate callback rate, lower bound.

- **sup_p_cand2:** 2nd candidate callback rate, upper bound.

wilson: a data frame containing the Wilson confidence intervals and the p-value of the equality test of callback shares between the two candidates, from `prop.test()`.

- **inf_p_cand1:** 1st candidate callback share, lower bound.
- **sup_p_cand1:** 1st candidate callback share, upper bound.
- **inf_p_cand2:** 2nd candidate callback share, lower bound.
- **sup_p_cand2:** 2nd candidate callback share, upper bound.
- **inf_cand_dif:** $p_{c10} - p_{c01}$, lower bound.
- **sup_cand_dif:** $p_{c10} - p_{c01}$, upper bound.

student: a data frame containing the Student confidence intervals and the p-value of the equality test of callback shares between the two candidates.

- **inf_p_cand1:** 1st candidate callback share, lower bound.
- **sup_p_cand1:** 1st candidate callback share, upper bound.
- **inf_p_cand2:** 2nd candidate callback share, lower bound.
- **sup_p_cand2:** 2nd candidate callback share, upper bound.
- **inf_cand_dif:** $p_{c10} - p_{c01}$, lower bound.
- **sup_cand_dif:** $p_{c10} - p_{c01}$, upper bound.

t.fisher: a data frame containing the statistics of the Fisher test.

- **p_cand_dif:** 1st candidate callback share - 1/2.
- **p_Fisher:** the p-value of the Fisher test.
- **s_Fisher:** the significance code of the Fisher test.

t.pearson: a data frame containing the statistics of the Pearson test.

- **p_cand_dif:** 1st candidate callback share - 1/2.
- **Pearson:** the value of Pearson's chi-squared test statistic.
- **p_Pearson:** the p-value of the Pearson test.
- **s_Pearson:** the significance code of the Pearson test.

t.student: A data frame containing the statistics of the Student test.

- **p_cand_dif:** 1st candidate callback share - 1/2.
- **Student:** the value of Student's test statistic.
- **p_Student:** the p-value of the Student test.
- **s_Student:** the significance code of the Student test.

Author(s)

Emmanuel Duguet

References

Clopper, C. J. & Pearson, E. S. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26, 404–413. doi:10.2307/2331986.

Wilson, E.B. (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22, 209–212. doi:10.2307/2276774.

Examples

```
data(labour1)
x <- callback(data=labour1, cluster="offer", candid="hist", callback="callback")
str(stat_ecs(x, level=0.9))
```

stat_glob

Unmatched callback rates

Description

Number and proportion of callbacks for all the candidates.

Usage

```
stat_glob(x, level = 0.95)
```

Arguments

x A callback object.

level A number, containing the level of the confidence intervals (0.95 by default).

Value

A `stat_glob` object with 5 components : `level`, `props`, `cp`, `wilson` and `student`.

`level`: the level of the confidence intervals

`props`: a data frame containing the following variables.

- **tests**: number of tests.
- **callback**: number of callbacks.
- **p_callback**: callback rate.

`cp`, `wilson` and `student` are data frames containing the following variables:

- **inf_p_callback**: callback rate lower bound.
- **sup_p_callback**: callback rate upper bound.

Author(s)

Emmanuel Duguet

References

Clopper, C. J. & Pearson, E. S. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26, 404–413. doi:10.2307/2331986.

Wilson, E.B. (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22, 209–212. doi:10.2307/2276774.

Examples

```
data(labour2)
x <- callback(data=labour2,cluster="offer",candid="hist",callback="callback")
str(stat_glob(x))
```

stat_mcr	<i>Matched callback rates</i>
----------	-------------------------------

Description

Computes the callback rates and their confidence intervals.

Usage

```
stat_mcr(x, level = 0.95)
```

Arguments

x a callback object.
level a number, containing the level of the confidence intervals (0.95 by default).

Value

A list with class "stat_mcr" containing 8 components: level, props, cp, wilson, student, t.fisher, t.pearson and t.student.

level: the level of the confidence intervals.

props: a data frame containing the following variables:

- **tests:** number of tests.
- **p_callback:** overall callback rate.
- **p_cand1:** 1st candidate callback rate.
- **p_cand_dif:** p_cand1-p_cand2.

cp: A data frame containing the Clopper-Pearson confidence intervals, from binom.test().

- **inf_p_callback:** overall callback rate, lower bound.
- **p_callback:** overall callback rate.
- **sup_p_callback:** overall callback rate, upper bound.

- **inf_p_cand1:** 1st candidate callback rate, lower bound.
- **p_cand1:** 1st candidate callback rate.
- **sup_p_cand1:** 1st candidate callback rate, upper bound.
- **inf_p_cand2:** 2nd candidate callback rate, lower bound.
- **p_cand2:** 2nd candidate callback rate.
- **sup_p_cand2:** 2nd candidate callback rate, upper bound.
- **inf_cand_dif:** $p_cand1 - p_cand2$, lower bound.
- **sup_cand_dif:** $p_cand1 - p_cand2$, upper bound.

wilson: a data frame containing the Wilson confidence intervals, from `prop.test()`.

- **inf_p_callback:** overall callback rate, lower bound.
- **p_callback:** overall callback rate.
- **sup_p_callback:** overall callback rate, upper bound.
- **inf_p_cand1:** 1st candidate callback rate, lower bound.
- **p_cand1:** 1st candidate callback rate.
- **sup_p_cand1:** 1st candidate callback rate, upper bound.
- **inf_p_cand2:** 2nd candidate callback rate, lower bound.
- **p_cand2:** 2nd candidate callback rate.
- **sup_p_cand2:** 2nd candidate callback rate, upper bound.
- **inf_cand_dif:** $p_cand1 - p_cand2$, lower bound.
- **sup_cand_dif:** $p_cand1 - p_cand2$, upper bound.

student: a data frame containing the Student confidence intervals, from `t.test()`.

- **inf_p_callback:** overall callback rate, lower bound.
- **p_callback:** overall callback rate.
- **sup_p_callback:** overall callback rate, upper bound.
- **inf_p_cand1:** 1st candidate callback rate, lower bound.
- **p_cand1:** 1st candidate callback rate.
- **sup_p_cand1:** 1st candidate callback rate, upper bound.
- **inf_p_cand2:** 2nd candidate callback rate, lower bound.
- **p_cand2:** 2nd candidate callback rate.
- **sup_p_cand2:** 2nd candidate callback rate, upper bound.
- **inf_cand_dif:** $p_cand1 - p_cand2$, lower bound.
- **sup_cand_dif:** $p_cand1 - p_cand2$, upper bound.

t.fisher: a data frame containing the statistics of the Fisher test.

- **p_cand_dif:** callback proportion difference between the candidates.
- **p_Fisher:** the p-value of the Fisher test.
- **s_Fisher:** the significance code of the Fisher test.

t.pearson: a data frame containing the statistics of the Pearson test.

- **p_cand_dif**: callback proportion difference between the candidates.
- **Pearson**: the value of Pearson's chi-squared test statistic.
- **p_Pearson**: the p-value of the Pearson test.
- **s_Pearson**: the significance code of the Pearson test.

t.student: A data frame containing the statistics of the Student test.

- **p_cand_dif**: callback proportion difference between the candidates.
- **Student**: the value of Student's test statistic.
- **p_Student**: the p-value of the Student test.
- **s_Student**: the significance code of the Student test.

Author(s)

Emmanuel Duguet

References

Clopper, C. J. & Pearson, E. S. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26, 404–413. doi:10.2307/2331986.

Wilson, E.B. (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22, 209–212. doi:10.2307/2276774.

Examples

```
data(labour1)
x <- callback(data=labour1, cluster="offer", candid="hist", callback="callback")
str(stat_mcr(x, level=0.9))
```

stat_signif

Significance code of a p-value

Description

Significance code of a p-value

Usage

```
stat_signif(p)
```

Arguments

p A number between 0 and 1

Value

A character string

Examples

```
stat_signif(0.045)
```

stat_tcs

Total callback shares

Description

Computes the callback shares and their confidence intervals. The analysis is restricted to the tests with at least one callback. It is the definition used in Riach and Rich (2006).

Usage

```
stat_tcs(x, level = 0.95)
```

Arguments

`x` a callback object.
`level` a number, containing the level of the confidence intervals (0.95 by default).

Value

A list with class "stat_tcs" containing 8 components : level, props, cp, wilson, student, t.student, t.pearson and t.fisher.

level: the level of the confidence intervals.

props: a data frame containing the following variables.

- **ncall**: number of callbacks.
- **c10**: number of tests with the 1st candidate preferred (2nd candidate discriminated against).
- **c01**: number of tests with the 2nd candidate preferred (1st candidate discriminated against).
- **c11**: number of tests with both candidates called back.
- **p_cand1**: 1st candidate total callback share (c10/ncall).
- **p_cand2**: 2nd candidate total callback share (c01/ncall).
- **p_equal**: equal treatment callback share (c11/ncall).
- **p_cand_dif**: p_cand1-p_cand2.

cp: a data frame containing the Clopper-Pearson confidence intervals, from `binom.test()`, and the p-value of the Fisher test of independence between the candidate type and the callback variable, from `fisher.test()`.

- **inf_p_cand1**: 1st candidate total callback share, lower bound.
- **sup_p_cand1**: 1st candidate total callback share, upper bound.
- **inf_p_cand2**: 2nd candidate total callback share, lower bound.
- **sup_p_cand2**: 2nd candidate total callback share, upper bound.
- **inf_p_equal**: equal treatment total callback rate, lower bound.
- **sup_p_equal**: equal treatment total callback rate, upper bound.

`wilson`: a data frame containing the Wilson confidence intervals and the p-value of the equality test of callback shares between the two candidates, from `prop.test()`.

- **inf_p_cand1**: 1st candidate total callback share, lower bound.
- **sup_p_cand1**: 1st candidate total callback share, upper bound.
- **inf_p_cand2**: 2nd candidate total callback share, lower bound.
- **sup_p_cand2**: 2nd candidate total callback share, upper bound.
- **inf_p_equal**: equal treatment total callback rate, lower bound.
- **sup_p_equal**: equal treatment total callback rate, upper bound.
- **inf_cand_dif**: $p_{c10} - p_{c01}$, lower bound.
- **sup_cand_dif**: $p_{c10} - p_{c01}$, upper bound.

`student`: a data frame containing the Student confidence intervals and the p-value of the equality test of callback shares between the two candidates.

- **inf_p_cand1**: 1st candidate total callback share, lower bound.
- **sup_p_cand1**: 1st candidate total callback share, upper bound.
- **inf_p_cand2**: 2nd candidate total callback share, lower bound.
- **sup_p_cand2**: 2nd candidate total callback share, upper bound.
- **inf_p_equal**: equal treatment total callback rate, lower bound.
- **sup_p_equal**: equal treatment total callback rate, upper bound.
- **inf_cand_dif**: $p_{c10} - p_{c01}$, lower bound.
- **sup_cand_dif**: $p_{c10} - p_{c01}$, upper bound.

`t.fisher`: a data frame containing the statistics of the Fisher test.

- **p_cand_dif**: $p_{cand1} - p_{cand2}$.
- **p_Fisher**: the p-value of the Fisher test.
- **s_Fisher**: the significance code of the Fisher test.

`t.pearson`: a data frame containing the statistics of the Pearson test.

- **p_cand_dif**: $p_{cand1} - p_{cand2}$.
- **Pearson**: the value of Pearson's chi-squared test statistic.
- **p_Pearson**: the p-value of the Pearson test.
- **s_Pearson**: the significance code of the Pearson test.

t.student: A data frame containing the statistics of the Student test.

- **p_cand_dif:** p_cand1-p_cand2.
- **Student:** the value of Student's test statistic.
- **p_Student:** the p-value of the Student test.
- **s_Student:** the significance code of the Student test.

Author(s)

Emmanuel Duguet

References

Clopper, C. J. & Pearson, E. S. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26, 404–413. doi:10.2307/2331986.

Riach, P. A., & Rich, J. (2006). An experimental investigation of sexual discrimination in hiring in the English labor market. *The BE Journal of Economic Analysis & Policy*, 6(2),

Wilson, E.B. (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22, 209–212. doi:10.2307/2276774.

Examples

```
data(labour1)
x <- callback(data=labour1, cluster="offer", candid="hist", callback="callback")
str(stat_ecs(x, level=0.9))
```

Index

* datasets

- address1, 2
- gender1, 5
- gender2, 6
- gender3, 7
- gender4, 8
- inter1, 16
- labour1, 18
- labour2, 19
- mobility1, 19
- origin1, 20
- origin2, 21

address1, 2

callback, 3

g_difp, 13

g_ecs, 14

g_prop, 15

g_tcs, 16

gender1, 5

gender2, 6

gender3, 7

gender4, 8

graph, 9

graph.stat_ecs, 9

graph.stat_glob, 10

graph.stat_mcr, 11

graph.stat_tcs, 12

inter1, 16

is.calc, 17

labour1, 18

labour2, 19

mobility1, 19

origin1, 20

origin2, 21

print.callback, 23

print.stat_count, 23

print.stat_ecs (print.stat_tcs), 25

print.stat_glob, 24

print.stat_mcr, 25

print.stat_tcs, 25

stat_colsums, 26

stat_count, 27

stat_ecs, 28

stat_glob, 30

stat_mcr, 31

stat_signif, 33

stat_tcs, 34