

Package ‘charlatan’

June 16, 2017

Type Package

Title Make Fake Data

Description Make fake data, supporting addresses, person names, dates, times, colors, coordinates, currencies, digital object identifiers ('DOIs'), jobs, phone numbers, 'DNA' sequences, doubles and integers from distributions and within a range.

Version 0.1.0

License MIT + file LICENSE

URL <https://github.com/ropensci/charlatan>

BugReports <https://github.com/ropensci/charlatan/issues>

LazyData true

Encoding UTF-8

VignetteBuilder knitr

Imports R6 (>= 2.2.0), tibble (>= 1.2), whisker

Suggests roxygen2 (>= 6.0.1), testthat, knitr, rmarkdown

RoxygenNote 6.0.1

Collate 'base-provider.R' 'datetime-provider.R' 'address-provider.R'
'available_locales.R' 'charlatan-package.R'
'charlatan_settings.R' 'color-provider-en_US.R'
'color-provider-uk_UA.R' 'color-provider.R' 'color.R'
'company-provider-bg_BG.R' 'company-provider-cs_CZ.R'
'company-provider-de_DE.R' 'company-provider-en_US.R'
'company-provider-es_MX.R' 'company-provider-fa_IR.R'
'company-provider-fr_FR.R' 'company-provider-hr_HR.R'
'company-provider-it_IT.R' 'company-provider.R' 'company.R'
'coordinate-provider.R' 'coordinates.R'
'credit_card-provider.R' 'credit_card.R' 'currency-provider.R'
'currency.R' 'date_time.R' 'doi-provider.R' 'doi.R'
'fraudster.R' 'generate.R' 'job.R' 'jobs-provider-en_US.R'
'jobs-provider-fa_IR.R' 'jobs-provider-fr_CH.R'
'jobs-provider-fr_FR.R' 'jobs-provider-hr_HR.R'

'jobs-provider-pl_PL.R' 'jobs-provider-ru_RU.R'
 'jobs-provider-uk_UA.R' 'jobs-provider-zh_TW.R'
 'jobs-provider.R' 'missing-data-provider.R' 'missing.R'
 'name.R' 'numerics-provider.R' 'numerics.R' 'onload.R'
 'person-provider-bg_BG.R' 'person-provider-es_CZ.R'
 'person-provider-de_AT.R' 'person-provider-de_DE.R'
 'person-provider-dk_DK.R' 'person-provider-en_US.R'
 'person-provider-es_ES.R' 'person-provider-es_MX.R'
 'person-provider-fa_IR.R' 'person-provider-fi_FI.R'
 'person-provider-fr_CH.R' 'person-provider-fr_FR.R'
 'person-provider-hr_HR.R' 'person-provider-it_IT.R'
 'person-provider-ja_JP.R' 'person-provider-ko_KR.R'
 'person-provider-lt_LT.R' 'person-provider-lv_lv.R'
 'person-provider-ne_NP.R' 'person-provider-nl_NL.R'
 'person-provider-no_NO.R' 'person-provider-pl_PL.R'
 'person-provider.R' 'phone-number-provider-all.R'
 'phone_number.R' 'phonenumbers-provider.R'
 'sequence-provider.R' 'sequences.R' 'taxonomy-provider.R'
 'taxonomy.R' 'zzz.R'

NeedsCompilation no

Author Scott Chamberlain [aut, cre],

Brooke Anderson [ctb] (Brooke Anderson reviewed the package for
 rOpenSci, see <https://github.com/ropensci/onboarding/issues/94>),

Tristan Mahr [ctb] (Tristan Mahr reviewed the package for rOpenSci, see
<https://github.com/ropensci/onboarding/issues/94>)

Maintainer Scott Chamberlain <myrmecocystus+r@gmail.com>

Repository CRAN

Date/Publication 2017-06-16 19:41:21 UTC

R topics documented:

charlatan-package	3
charlatan_locales	4
charlatan_settings	4
ch_color	5
ch_company	6
ch_credit	6
ch_currency	7
ch_doi	8
ch_generate	8
ch_gene_sequence	9
ch_job	10
ch_missing	10
ch_name	11
ch_phone_number	11
coordinates	12

date_time	13
fraudster	13
numerics	14
taxonomy	15

Index	17
--------------	-----------

charlatan-package	<i>charlatan</i>
-------------------	------------------

Description

Make fake data, supporting addresses, person names, dates, times, colors, coordinates, currencies, digital object identifiers (DOIs), jobs, phone numbers, DNA sequences, doubles and integers from distributions and within a range.

Package API

- `ch_generate`: generate a data.frame with fake data
- `fraudster`: single interface to all fake data methods
- High level interfaces: There are high level functions prefixed with `ch_` that wrap low level interfaces, and are meant to be easier to use and provide easy way to make many instances of a thing.
- Low level interfaces: All of these are R6 objects that a user can initialize and then call methods on the them.

Author(s)

Scott Chamberlain <myrmecocystus+r@gmail.com>

Examples

```
# generate individual types of data
ch_name()
ch_phone_number()
ch_job()

# generate a data.frame
ch_generate()

# one interface to all data types - generate the class first
# reports the locale to be used, can change optionally
(x <- fraudster())
x$job()
x$name()
x$color_name()
x$hex_color()

# low level interfaces to "data providers"
```

```
# these are exported by hidden from package man page
# as most users will likely not interact with these
x <- ColorProvider$new()
x$color_name()
x$hex_color()
```

charlatan_locales *Available locales*

Description

Vector of available locales - not all are available across all types of data.

Examples

```
charlatan_locales()
```

charlatan_settings *charlatan settings*

Description

charlatan settings

Usage

```
charlatan_settings(messy = NULL)
```

Arguments

`messy` (logical) make some messy data. Default: NULL

More deets

- `messy` - When FALSE, nothing is different from normal. When TRUE, we select incorrect/wrong values with probability X. Messy mode is only available for **en-US** for now, and only for some data types. The default setting is NULL, meaning it is ignored.

Examples

```
charlatan_settings()
charlatan_settings(messy = TRUE)
charlatan_settings(messy = FALSE)

# with PersonProvider - overrides local messy param in all cases
x <- PersonProvider$new()
x$messy
charlatan_settings(messy = TRUE)
x <- PersonProvider$new()
x$messy
```

ch_color	<i>Create fake colors</i>
----------	---------------------------

Description

Create fake colors

Usage

```
ch_color_name(n = 1, locale = NULL)
```

```
ch_safe_color_name(n = 1)
```

```
ch_hex_color(n = 1)
```

```
ch_safe_hex_color(n = 1)
```

```
ch_rgb_color(n = 1)
```

```
ch_rgb_css_color(n = 1)
```

Arguments

n	(integer) number of things to get, any non-negative integer
locale	(character) the locale to use. options: en_US (default), or uk_UA. Only affects ch_color_name function

Examples

```
ch_color_name()  
ch_color_name(10)  
ch_color_name(500)
```

```
ch_safe_color_name()  
ch_safe_color_name(10)
```

```
ch_hex_color()  
ch_hex_color(10)  
ch_hex_color(1000)
```

```
ch_safe_hex_color()  
ch_safe_hex_color(10)
```

```
ch_rgb_color()  
ch_rgb_color(10)
```

```
ch_rgb_css_color()  
ch_rgb_css_color(10)
```

```
ch_color_name(locale = "uk_UA")
ch_color_name(n = 10, locale = "uk_UA")
```

ch_company *Create fake company names and other company bits*

Description

Create fake company names and other company bits

Usage

```
ch_company(n = 1, locale = NULL)
```

Arguments

n (integer) number of things to get, any non-negative integer

locale (character) the locale to use. options: en_US (default), bg_bg, cs_cz, de_de, es_mx, fa_ir, fr_fr, hr_hr, it_it

Examples

```
ch_company()
ch_company(10)
ch_company(500)

ch_company(locale = "fr_FR", n = 10)
ch_company(locale = "cs_CZ", n = 10)
ch_company(locale = "es_MX", n = 10)
ch_company(locale = "hr_HR", n = 10)
```

ch_credit *Create fake credit card data*

Description

Create fake credit card data

Usage

```
ch_credit_card_provider(n = 1)

ch_credit_card_number(n = 1)

ch_credit_card_security_code(n = 1)
```

Arguments

n (integer) number of things to get, any non-negative integer

Examples

```
ch_credit_card_provider()  
ch_credit_card_provider(n = 4)
```

```
ch_credit_card_number()  
ch_credit_card_number(n = 10)  
ch_credit_card_number(n = 500)
```

```
ch_credit_card_security_code()  
ch_credit_card_security_code(n = 10)  
ch_credit_card_security_code(n = 500)
```

ch_currency *Create fake currencies*

Description

Create fake currencies

Usage

```
ch_currency(n = 1)
```

Arguments

n (integer) number of things to get, any non-negative integer

Examples

```
ch_currency()  
ch_currency(10)  
ch_currency(500)
```

ch_doi	<i>Create fake DOIs (Digital Object Identifiers)</i>
--------	--

Description

Create fake DOIs (Digital Object Identifiers)

Usage

```
ch_doi(n = 1)
```

Arguments

n (integer) number of things to get, any non-negative integer

Examples

```
ch_doi()
ch_doi(10)
ch_doi(100)
```

ch_generate	<i>Generate a fake dataset</i>
-------------	--------------------------------

Description

Generate a fake dataset

Usage

```
ch_generate(..., n = 10, locale = NULL)
```

Arguments

... columns to include. must be in the allowed set. See **Allowed column names** below. Three default columns are included (name, job, phone_number) if nothing is specified - but are overridden by any input.

n (integer) number of things to get, any non-negative integer

locale (character) the locale to use. options: only supported for data types that have locale support, See each data provider for details.

Allowed column names

- name (default included)
- job (default included)
- phone_number (default included)
- currency
- color_name
- rgb_color
- rgb_css_color

Examples

```
ch_generate()
ch_generate(n = 1)
ch_generate(n = 100)

ch_generate('job')
ch_generate('job', 'name')
ch_generate('job', 'color_name')

# locale
ch_generate(locale = "en_US")
ch_generate(locale = "fr_FR")
ch_generate(locale = "fr_CH")
```

ch_gene_sequence	<i>Create fake gene sequences</i>
------------------	-----------------------------------

Description

Create fake gene sequences

Usage

```
ch_gene_sequence(n = 1, length = 30)
```

Arguments

n (integer) number of things to get, any non-negative integer
length (integer) length of sequence to create

Examples

```
ch_gene_sequence()
ch_gene_sequence(10)
ch_gene_sequence(100)

ch_gene_sequence(length = 500)
ch_gene_sequence(10, length = 500)
```

ch_job	<i>Create fake jobs</i>
--------	-------------------------

Description

Create fake jobs

Usage

```
ch_job(n = 1, locale = NULL)
```

Arguments

n	(integer) number of things to get, any non-negative integer
locale	(character) the locale to use. options: en_US (default), fr_FR, fr_CH, hr_HR, fa_IR, pl_PL, ru_RU, uk_UA, zh_TW.

Examples

```
ch_job()  
ch_job(10)  
ch_job(500)
```

```
ch_job(locale = "fr_FR", n = 10)  
ch_job(locale = "fr_CH", n = 10)  
ch_job(locale = "hr_HR", n = 10)  
ch_job(locale = "fa_IR", n = 10)  
ch_job(locale = "pl_PL", n = 10)  
ch_job(locale = "ru_RU", n = 10)  
ch_job(locale = "uk_UA", n = 10)  
ch_job(locale = "zh_TW", n = 10)
```

ch_missing	<i>Create missing data</i>
------------	----------------------------

Description

Create missing data

Usage

```
ch_missing(x, n = 1)
```

Arguments

x	Input vector, can be any class - only 1 vector
n	(integer) number of things to get, any non-negative integer

Examples

```
ch_missing(letters)
ch_missing(letters, 10)
ch_missing(letters, 20)
```

ch_name	<i>Create fake person names</i>
---------	---------------------------------

Description

Create fake person names

Usage

```
ch_name(n = 1, locale = NULL, messy = FALSE)
```

Arguments

n	(integer) number of things to get, any non-negative integer
locale	(character) the locale to use. options: en_US (default), fr_FR, fr_CH, fi_FI, fa_IR, es_ES, de_DE, de_AT, cs_CZ, bg_BG, dk_DK
messy	(logical) make some messy data. Default: FALSE

Examples

```
ch_name()
ch_name(10)
ch_name(500)

ch_name(locale = "fr_FR", n = 10)
ch_name(locale = "fr_CH", n = 10)
ch_name(locale = "fa_IR", n = 10)
ch_name(locale = "fi_FI", n = 10)
```

ch_phone_number	<i>Create fake phone numbers</i>
-----------------	----------------------------------

Description

Create fake phone numbers

Usage

```
ch_phone_number(n = 1, locale = NULL)
```

Arguments

n (integer) number of things to get, any non-negative integer

locale (character) the locale to use. options: en_US (default), en_US, es_ES, es_MX, en_GB, en_CA, el_GR, dk_DK, de_DE, cs_CZ, bs_BA, bg_BG, fa_IR, fi_FI, fr_CH, fr_FR, hi_IN, hr_HR, hu_HU, it_IT, ja_JP, ko_KR, it_LT, lv_LV, ne_NP, nl_BE, nl_NL, no_NO, pl_PL, pt_BR, pt_PT, ru_RU, sk_SK, sl_SL, sv_SE, tr_TR, uk_UA, zh_TW

Examples

```
ch_phone_number()
ch_phone_number(10)
ch_phone_number(500)

# locales
ch_phone_number(locale = "fr_FR")
ch_phone_number(locale = "uk_UA")
ch_phone_number(locale = "en_CA")
ch_phone_number(locale = "lv_LV")
```

coordinates

Create fake coordinates

Description

Create fake coordinates

Usage

```
ch_lon(n = 1)

ch_lat(n = 1)

ch_position(n = 1, bbox = NULL)
```

Arguments

n (integer) number of things to get, any non-negative integer

bbox a bounding box of the form [w,s,e,n]

Examples

```
ch_lon()
ch_lon(10)

ch_lat()
ch_lat(10)
```

```
ch_position()  
ch_position(10)  
ch_position(bbox = c(-120, 30, -110, 60))
```

date_time	<i>Create dates and times</i>
-----------	-------------------------------

Description

Create dates and times

Usage

```
ch_timezone(n = 1)  
ch_unix_time(n = 1)  
ch_date_time(n = 1)
```

Arguments

n (integer) number of things to get, any non-negative integer

Examples

```
ch_timezone()  
ch_timezone(10)  
  
ch_unix_time()  
ch_unix_time(20)  
  
ch_date_time()  
ch_date_time(20)
```

fraudster	<i>Fraudster - catch all client to make all types of fake data</i>
-----------	--

Description

Fraudster - catch all client to make all types of fake data

Usage

```
fraudster(locale = NULL)
```

Arguments

locale (character) the locale to use. options: en_US (default), fr_FR, fr_CH, hr_FR, fa_IR, pl_PL, ru_RU, uk_UA, zh_TW.

Examples

```
# English - the default locale
(x <- fraudster())
x$job()
x$name()
x$color_name()
x$safe_color_name()
x$hex_color()
x$safe_hex_color()
x$rgb_color()
x$rgb_css_color()

# different locales
## French
(y <- fraudster(locale = "fr_FR"))
y$job()

## Croatian
(z <- fraudster(locale = "hr_HR"))
z$job()

## Ukranian
(w <- fraudster(locale = "uk_UA"))
w$job()
w$color_name()

# geospatial
x$lat()
x$lon()
x$position()

# DOIs (Digital Object Identifier)
x$doi()
```

numerics

Create numbers

Description

Create numbers

Usage

```
ch_double(n = 1, mean = 0, sd = 1)
ch_integer(n = 1, min = 1, max = 1000)
ch_unif(n = 1, min = 0, max = 9999)
ch_norm(n = 1, mean = 0, sd = 1)
ch_lnorm(n = 1, mean = 0, sd = 1)
ch_beta(n = 1, shape1, shape2, ncp = 0)
```

Arguments

n	(integer) number of things to get, any non-negative integer
mean	mean value
sd	standard deviation
min	minimum value
max	maximum value
shape1, shape2	non-negative parameters of the Beta distribution
ncp	non-centrality parameter

Examples

```
ch_double()
ch_double(10)
ch_double(100)

ch_integer()
ch_integer(10)
ch_integer(100)

ch_unif()
ch_norm()
ch_lnorm()
ch_beta(shape1 = 1, shape2 = 1)
```

taxonomy

Create fake taxonomic names

Description

Create fake taxonomic names

Usage

```
ch_taxonomic_genus(n = 1)

ch_taxonomic_epithet(n = 1)

ch_taxonomic_species(n = 1)
```

Arguments

n (integer) number of things to get, any non-negative integer

Names

Names were taken from Theplantlist. 500 genera names and 500 epithets were chosen at random from the set of 10,000 names in the dataset in the `taxize` package. Theplantlist is, as it says on the tin, composed of plant names - so these fake names are derived from plant names if that matters to you. These may generate names that match those of real taxa, but may not as well.

Taxonomic authority

Randomly, the taxonomic authority is in parentheses - which represents that the given authority was not the original authority.

Examples

```
ch_taxonomic_genus()
ch_taxonomic_genus(10)
ch_taxonomic_genus(500)

ch_taxonomic_epithet()
ch_taxonomic_epithet(10)
ch_taxonomic_epithet(500)

ch_taxonomic_species()
ch_taxonomic_species(10)
ch_taxonomic_species(500)
```


Index

*Topic **package**

charlatan-package, 3

ch_beta (numerics), 14

ch_color, 5

ch_color_name (ch_color), 5

ch_company, 6

ch_credit, 6

ch_credit_card_number (ch_credit), 6

ch_credit_card_provider (ch_credit), 6

ch_credit_card_security_code
(ch_credit), 6

ch_currency, 7

ch_date_time (date_time), 13

ch_doi, 8

ch_double (numerics), 14

ch_gene_sequence, 9

ch_generate, 3, 8

ch_hex_color (ch_color), 5

ch_integer (numerics), 14

ch_job, 10

ch_lat (coordinates), 12

ch_lnorm (numerics), 14

ch_lon (coordinates), 12

ch_missing, 10

ch_name, 11

ch_norm (numerics), 14

ch_phone_number, 11

ch_position (coordinates), 12

ch_rgb_color (ch_color), 5

ch_rgb_css_color (ch_color), 5

ch_safe_color_name (ch_color), 5

ch_safe_hex_color (ch_color), 5

ch_taxonomic_epithet (taxonomy), 15

ch_taxonomic_genus (taxonomy), 15

ch_taxonomic_species (taxonomy), 15

ch_timezone (date_time), 13

ch_unif (numerics), 14

ch_unix_time (date_time), 13

charlatan (charlatan-package), 3

charlatan-package, 3

charlatan_locales, 4

charlatan_settings, 4

coordinates, 12

date_time, 13

fraudster, 3, 13

numerics, 14

taxonomy, 15