

# Package ‘flatr’

November 14, 2017

**Type** Package  
**Title** Transforms Contingency Tables to Data Frames, and Analyses Them  
**Version** 0.01.0  
**Author** Scott D. Graham  
**Maintainer** Scott D. Graham <scott.grah95@gmail.com>  
**Description** Contingency Tables are a pain to work with when you want to run regressions. This package takes them, flattens them into a long data frame, so you can more easily analyse them!  
As well, you can calculate other related statistics. All of this is done so in a 'tidy' manner, so it should tie in nicely with 'tidyverse' series of packages.  
**Depends** R(>= 3.4.2), stats, dplyr, tibble, magrittr  
**License** MIT + file LICENSE  
**Encoding** UTF-8  
**LazyData** true  
**RoxygenNote** 6.0.1  
**NeedsCompilation** no

## R topics documented:

flatten_ct . . . . .	1
goodness_of_fit . . . . .	2
lung_cancer . . . . .	3
print.ct_goodness_of_fit . . . . .	4
<b>Index</b>	<b>5</b>

---

flatten_ct	<i>Flatten i*j*k contingency tables into tidy data.</i>
------------	---------------------------------------------------------

---

### Description

flatten\_ct() takes a i\*j\*k array, and turns it into a tibble

### Usage

flatten\_ct(data)

**Arguments**

`data` An  $i*j*k$  array.

**Value**

A tibble with 3 columns.

**Examples**

```
flatten_ct(lung_cancer)
```

---

<code>goodness_of_fit</code>	<i>Calculate the <math>\chi^2</math> and <math>G^2</math> Statistics</i>
------------------------------	--------------------------------------------------------------------------

---

**Description**

Calculates the goodness of fit test statistics for contingency tables

**Usage**

```
goodness_of_fit(model, response, type = "Chisq", ...)
```

**Arguments**

`model` a GLM regression model.

`response` a string with the same name as the response column in the data

`type` either "Chisq" or "Gsq", which determines the type of goodness of fit test that is ran. Defaults to "Chisq".

`...` Further arguments passed to or from other methods.

**Value**

A list with class "ct\_goodness\_of\_fit" containing the following components:

`test` the type of test used.

`model` the name of the inputted model.

`statistic` The value of the test statistic as determined by the type parameter

`df` The number of degrees of freedom. This equals the number of combinations for explanatory variables less the number of parameters in the model

`p.value` The p-value calculated under a Chi-Squared distribution.

**Examples**

```
lung_logit <-
  lung_cancer %>%
  flatten_ct() %>%
  glm(
    Lung ~ City + Smoking
    ,family = binomial
    ,data = .
  )

goodness_of_fit(model = lung_logit, response = "Lung", type = "Chisq")
lung_logit %>%
  goodness_of_fit(response = "Lung", type = "Gsq")
lung_cancer %>%
  flatten_ct() %>%
  glm(
    Lung ~ City + Smoking
    ,family = binomial
    ,data = .
  ) %>%
  goodness_of_fit(response = "Lung", type = "Chisq")
```

lung\_cancer

*Lung Cancer by whether or not a person smokes and City.***Description**

Based on data in Z. Liu, Int. J. Epidemiol., 21: 197–201, 1992.

**Usage**

```
lung_cancer
```

**Format**

An Array with 2\*2\*8 dimensions

**Smoking** Whether or not a person smokes.

**Lung** Whether or not a person has lung cancer.

**City** Name of the city a person lives in.

**Examples**

```
lung_cancer
```

---

```
print.ct_goodness_of_fit
```

*Print method for goodness\_of\_fit()*

---

**Description**

Creates a nice looking output for the goodness\_of\_fit() function

**Usage**

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

**Arguments**

x	A list
...	Further arguments passed to or from other methods.

# Index

\*Topic **datasets**

lung\_cancer, [3](#)

flatten\_ct, [1](#)

goodness\_of\_fit, [2](#)

lung\_cancer, [3](#)

print.ct\_goodness\_of\_fit, [4](#)