Package: DHBins (via r-universe)

September 5, 2024

Title Hexmaps for NZ District Health Boards

Version 1.3

Author Thomas Lumley

Description Draws stylized choropleth maps -- hexagonal maps and triangular multiclass hex maps -- for New Zealand District Health Boards and Regional Council areas. These allow faceted, coloured displays of quantitative information for comparison across District Health Boards or Regional Councils. The preprint Lumley (2019) <arXiv:1912.04435> is based on the methods in this package.

Imports graphics, grDevices,grid

Depends ggplot2, R (>= 3.6.0)

Suggests knitr

VignetteBuilder knitr

License GPL-3

Encoding UTF-8

Maintainer Thomas Lumley <t.lumley@auckland.ac.nz>

Repository https://tslumley.r-universe.dev

RemoteUrl https://github.com/tslumley/dhbins

RemoteRef HEAD

RemoteSha 8f23312cd0a21e2aba5d5d745263a8fdda109ee2

Contents

dhbin *Hexmaps of New Zealand Adminstrative Units*

Description

The 20 District Health Boards are the main administrative and funding units of the NZ national health system. The Regions are the second-level government divisions . These functions draw stylised maps of the DHBs and regions, along the lines of "statebins" for the US states. The hexagon for each location can be coloured and sized to show numeric data, and it can also be split into six triangles to show proportion or composition data.

Usage

```
dhbin(radius = NULL, hex_colours = "lightskyblue", DHB_names=NULL, text_colour = "black",
 legend_opts = NULL, border = NULL, short = FALSE,cex=0.8)
dhtri(radius = NULL, tri_colours, DHB_names=NULL, text_colour = "black",
 legend_opts = NULL, short=FALSE,cex=0.8)
regionbin(radius = NULL, hex_colours = "lightskyblue", region_names=NULL,
 text_colour = "black", legend_opts = NULL, border = NULL, short = FALSE, tasman=TRUE,
 cex=0.7)
regiontri(radius = NULL, tri_colours, region_names=NULL, text_colour = "black",
 legend_opts = NULL,short=FALSE, tasman=TRUE, cex=0.7)
```

Arguments

radius	Vector of radius measurements for the coloured part of each hexagon, with 1 as a full hexagon. If any are greater than 1, the vector is scaled to have maximum value 0.95, which is also the default value for all hexagons. If it has names, they are matched to the DHB names or common alternative names.
hex_colours	Vector of colours as names or hashtag colours. If it has names, they are matched to the DHB names or common alternative names.
tri_colours	6-column matrix of colours for the six triangles in each hexagon. If it has row names, they are matched to the DHB names or common alternative names.
DHB_names, region_names	Vector of DHB names or region names. If NULL, the function will look for this as the names of radius then as the names of hex_colours or the rownames of tri_colours
text_colour	Colour for the label on all the hexagons.
legend_opts	a list with arguments to graphics::legend, not including x and y, which are supplied, and bty, which is forced to "n". If not NULL a legend is drawn.
border	NULL or a colour name to draw a border around each hex.
short	If TRUE, label each hexagon with a 1-3 character short abbreviation rather than the full name. Especially useful for faceting.

tasman	The Tasman District and Nelson City are considered as separate regions in most contexts, but as forming the Nelson Region in some contexts. Use tasman=FALSE when they are combined
cex	character expansion factor

Value

Called for its side-effect

References

Lumley T (2019) Stylised Choropleth Maps for New Zealand Regions and District Health Boards. arXiv:1912.04435

See Also

tri_alloc for the Sainte-Laguë method to allocate triangles based on counts

Examples

```
data(dhb_cars)
tris<-tri_alloc(dhb_cars[,-1], c("green","gold","orange","goldenrod"), names=dhb_cars$dhb )

dhtri(tri_colours=tris,
  legend=list(fill=c("green","gold","orange","goldenrod"),
         border=NA,
         legend=c("0","1","2","3+"),
         title="Cars/Household")
)


households<-rowSums(dhb_cars[,-1])
names(households)<-dhb_cars$dhb
dhbin(radius=sqrt(households))
title(main="Number of households in private dwellings")


opar<-par(mfrow=c(2,3),mar=c(1,1,1,1))
z<-rnorm(20)
for(i in 1:6){
  z<-(rnorm(20)+z)/sqrt(2)
  z1<- (z+4)/8
  col_z<-rgb(colorRamp(c("blue", "white","red"))(z1),max=255)
  dhbin(hex_colours=col_z,border="grey",short=TRUE,
        legend_opts=list(fill=c("red","white","blue"),
                         legend=c("High","Medium","Low"),
                         title=paste("Thing",i))
  )
}
par(opar)
```

```
data(region_ethnicity)
cols<-tri_alloc(as.matrix(region_ethnicity[,-1]),
 colours=hcl.colors(5,"Set2"),names=region_ethnicity$Area)

regiontri(tri_colours=cols,text_colour="white",
 legend_opts=list(fill=hcl.colors(5,"Set2"),
    title="Ethnicity",
    legend=names(region_ethnicity)[2:6]))
title(main="New Zealand regions")
```

---

dhb_cars *Cars per household in New Zealand*

Description

The distribution of cars per household across New Zealand District Health Boards, based on data from the 2013 Census

Usage

```
data("dhb_cars")
```

Format

A data frame with 20 observations on the following 5 variables.

`dhb` District Health Board name

`none` Number of households with no cars

`one` Number of households with one car

`two` Number of households with two cars

`more` Number of households with three or more cars

Source

Statistics New Zealand http://archive.stats.govt.nz/Census/2013-census/data-tables/dhb-tables.aspx

Examples

```
data(dhb_cars)
tris<-tri_alloc(dhb_cars[,-1], c("green","gold","orange","goldenrod"), names=dhb_cars$dhb )

dhtri(tri_colours=tris,
  legend=list(fill=c("green","gold","orange","goldenrod"),
         border=NA,
         legend=c("0","1","2","3+"),
         title="Cars/Household")
)
```

dhb_fixname *Standardise DHB names*

Description

Converts popular variants of the DHB or region names into the versions used as keys in this package.

Usage

```
dhb_fixname(names)
region_fixname(names)
```

Arguments

names	vector of strings with DHB or region names

Value

vector of strings with standardised DHB or region names

Examples

```
dhb_fixname( c("Hawkes Bay","Capital & Coast","Counties"))

region_fixname(c("Nelson City","Wellington region","Auckland","Tasman district","Nelson Tasman"))
```

geom_dhb *Geom for DHB hexmap*

Description

A ggplot2 geom for the District Health Board hexmap: geom_dhb wraps geom_map and geom_label_dhb wraps geom_text.

Usage

```
geom_dhb(mapping = NULL, data = NULL, stat="identity",...,
  na.rm = FALSE, show.legend = NA, inherit.aes = TRUE,coord=TRUE)
geom_label_dhb(mapping = NULL, data = NULL, ...,
  na.rm = FALSE, show.legend = NA, inherit.aes = TRUE, short=FALSE)
geom_dhbtri(mapping = NULL, data = NULL, stat="identity",...,
  na.rm = FALSE, show.legend = NA, inherit.aes = TRUE,coord=TRUE)
geom_region(mapping = NULL, data = NULL, stat="identity",...,
  na.rm = FALSE, show.legend = NA, inherit.aes = TRUE,coord=TRUE)
geom_label_region(mapping = NULL, data = NULL, ...,
  na.rm = FALSE, show.legend = NA, inherit.aes = TRUE,short=FALSE)
geom_regiontri(mapping = NULL, data = NULL,stat="identity", ...,
  na.rm = FALSE, show.legend = NA, inherit.aes = TRUE,coord=TRUE)
```

Arguments

mapping	Set of aesthetic mappings created by aes() or aes_(). For geom_dhb and geom_region you must specify map_id with the unit names, and for geom_dhbtri and geom_regiontri additionally specify class_id giving the order of triangles within a hex. The radius aesthetic controls the size of the hexes
data	The data to be displayed in this layer. Usually inherited from the plot data
stat	The statistical transformation to use on the data for this layer, as a string.
...	Other arguments passed on to layer(), such as aesthetics, used to set an aesthetic to a fixed value, like colour = "white" or cex = 3.
na.rm	If FALSE, warn when missing values are removed
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.
coord	If TRUE, set the aspect ratio and plot limits. The main reason to have coord=FALSE is if you are stacking two geom_dhb layers
short	If TRUE, use the 'short' (mostly 1-character) abbreviations for DHB or region names

Value

The geoms

See Also

dhb_fixname

Examples

```
data(immune)
summary(immune)

immune$vax_pct<-cut(immune$Pcttotal,c(0,.80,.85,.90,.95,1))
ggplot(immune)+
 geom_dhb(aes(fill=vax_pct,map_id=dhb_fixname(DHB)))+
 scale_fill_viridis_d(drop=FALSE)+
 geom_label_dhb(short=TRUE,colour="white")

## size variation, plus stacking to show the full-size hexes underneath,
## plus no labels
ggplot(immune)+
 geom_dhb(aes(map_id=dhb_fixname(DHB)),fill="white",colour="grey",coord=FALSE)+
 geom_dhb(aes(fill=vax_pct,map_id=dhb_fixname(DHB),radius=sqrt(Ntotal)),alpha=0.5)+
 scale_fill_viridis_d(drop=FALSE)


data(immune_long)
ggplot(immune_long) +
```

```
    geom_dhb(aes(map_id=dhb_fixname(DHB),fill=pct_vax),
  alpha=0.5,colour="lightgrey") +
    scale_fill_viridis_d(drop=FALSE)+
    geom_label_dhb(short=TRUE,colour="black",cex=3)+
    facet_wrap(~ethnicity)


data(dhb_cars)
tris<-tri_alloc(dhb_cars[,-1], c("0","1","2","3+"), names=dhb_cars$dhb )
tri_data<-data.frame(DHB=rep(rownames(tris),6),
                     cars=as.vector(tris),
       tri_id=rep(1:6,each=nrow(tris)))

ggplot(tri_data)+
 geom_dhbtri(aes(map_id=DHB,class_id=tri_id, fill=cars),alpha=0.5)+
 scale_fill_viridis_d()+
 geom_label_dhb(short=TRUE)



data(region_ethnicity)
tri_eth<-tri_alloc(as.matrix(region_ethnicity[,-1]),
 classes=c("Asian","Euro","Maori","MELAA","Pacific"),
 names=region_ethnicity$Area)


tri_data<-data.frame(Region=rep(rownames(tri_eth),6),
                     ethnicity=as.vector(tri_eth),
       tri_id=rep(1:6,each=nrow(tri_eth)))

ggplot(tri_data)+
 geom_regiontri(aes(map_id=Region,class_id=tri_id, fill=ethnicity))+
 geom_label_region(colour="Black", short=TRUE,cex=3)
```

`immune` *Immunisation coverage*

Description

The report measures the number of children who turned the milestone age of 5 years between 01-Jul-2019 and 30-Sep-2019 and who have completed their age appropriate immunisations by the time they turned the milestone age. The target is 95%. Children are only counted once, according to 'prioritised ethnicity': the priority order is Māori, Pacific, Asian, other, NZ European. The `immune_long` variant is in long form with separate rows for each ethnicity, and with missing values where the population size is too small.

Usage

```
data("immune")
```

Format

For immune, a data frame with 20 observations on the following 19 variables.

DHB name of District Health Board

Ntotal Number of children

Vaxtotal Number who completed their vaccinations

Pcttotal Proportion

NNZE,VaxNSE,PctNZE the same, for children of NZ European ethnicity

NMaori,VaxMaori,PctMaori the same, for children of Māori ethnicity

NPacific,VaxPacific,PctPacific the same, for children of Pacific ethnicities

NAsian,VaxAsian,PctAsian the same, for children of Asian ethnicities

NOther,VaxOther,PctOther the same, for children of other ethnicities

For immune_long, a data frame with 120 observations on the following 3 variables.

DHB name of District Health Board

ethnicity ethnicity: a factor with levels total NZE Maori Pacific Asian Other

pct_vax percent vaccinated, a factor with levels [0,0.8) [0.8,0.85) [0.85,0.9) [0.9,0.95) [0.95,1)

Source

New Zealand Health Indicators, 2019

References

Statistics New Zealand. Health Indicators. http://archive.stats.govt.nz/browse_for_stats/snapshots-of-nz/nz-social-indicators/Home/Health/childhood-immunisation.aspx

Examples

```
data(immune)
summary(immune)

immune$vax_pct<-cut(immune$Pcttotal,c(0,.80,.85,.90,.95,1))
ggplot(immune)+
 geom_dhb(aes(fill=vax_pct,map_id=dhb_fixname(DHB)))+
 scale_fill_viridis_d(drop=FALSE)+
 geom_label_dhb(short=TRUE,colour="white")

data(immune_long)
ggplot(immune_long) +
    geom_dhb(aes(map_id=dhb_fixname(DHB),fill=pct_vax),alpha=0.5,colour="lightgrey") +
    scale_fill_viridis_d(drop=FALSE)+
    geom_label_dhb(short=TRUE,colour="black")+
    facet_wrap(~ethnicity)

cols<-c("goldenrod","red","orange","gold","springgreen")
```

```
with(immune, dhbin(hex_colours=cols[vax_pct], legend_opts=list(fill=cols,
legend=c("<80","80-85","85-90","90-95","95+"), title="Immunisation coverage (pct)")
 ))

## Add some transparency
with(immune, dhbin(hex_colours=adjustcolor(cols[vax_pct],alpha.f=.5), legend_opts=list(fill=cols,
legend=c("<80","80-85","85-90","90-95","95+"), title="Immunisation coverage (pct)")
 ))
```

`region_ethnicity` *Ethnic makeup of New Zealand, by region*

Description

Number of people, by ethnicity, for the New Zealand regions, based on the 2013 Estimated Resident Population.

Usage

```
data("region_ethnicity")
```

Format

A data frame with 16 observations on the following 6 variables.

`Area` a factor with level `Auckland region` `Bay of Plenty region` `Canterbury region` `Gisborne region` `Hawke\'s Bay region` `Manawatu-Wanganui region` `Marlborough region` `Nelson region` `Northland region` `Otago region` `Southland region` `Taranaki region` `Tasman region` `Waikato region` `Wellington region` `West Coast region`

`Asian` number of people

`Euro_Other` number of people

`Maori` number of people

`MELAA` number of people

`Pacific` number of people

Source

https://figure.nz/table/foip3RBvo69sO3J3, originally from Statistics New Zealand

Examples

```
data(region_ethnicity)
summary(region_ethnicity)
```

tri_alloc *Allocate triangles within hexagons*

Description

This implements the Webster/Sainte-Laguë method to allocate six triangles in each hexagon, in proportion to counts in two or more classes.

Usage

```
tri_alloc(countmatrix, colours, classes=colours, names = rownames(countmatrix))
```

Arguments

countmatrix	A matrix of counts with a column for each class and a row for each hexagon.
colours, classes	A vector of class names, or colour names with one for each class
names	A vector of names, for each row of countmatrix, to be applied to the output

Value

Matrix of class names, with six columns. The rownames will be set to names if provided.

Examples

```
data(dhb_cars)
tri_alloc(dhb_cars[,-1], c("green","gold","orange","goldenrod"), names=dhb_cars$dhb )
tri_alloc(dhb_cars[,-1], c("0","1","2","3+"), names=dhb_cars$dhb )
```

Index