post02

Jason Liu

2017/11/27

Post 02 - Converting Maps into Treemaps Using `maps`, `ggplot2`, and `treemapify`

1. Introduction

In this post, I will introduce the package `treemapify` which can generate treemaps. All the data in this post are from the package `maps`. In this post, I will use the data in `maps` to draw maps of France, Italy, and New Zealand at first, and then convert the three maps into treemaps using the package `treemapify`. I will also give different colors to different regions on a treemap at the end of this post. It is useful to learn how to draw treemaps because treemaps can help us learn the areas of different provinces and other data set more intuitively and directly.

Now, let's start.

2. Load Packages `maps`, `ggplot2`, `treemapify`, and `dplyr`

Firstly, we will load the packages `ggplot2`, `maps`, `treemapify`, and `dplyr`. We are familiar with `ggplot2` and `dplyr`. So let me introduce the other two packages `maps` and `treemapify`. `maps` contains a large amount of map data of various countries and even the whole world. `treemapify` is a package that can draw treemaps based on a given data set. Therefore, with these two packages, we can convert maps into treemaps.

```r
# load packages
library(maps)
library(ggplot2)
library(treemapify)
```

```
## Warning: package 'treemapify' was built under R version 3.4.2
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

3. Maps and Treemaps

Next, let's draw the map of France.

```r
# Define France as the data 'france' in the package `maps`, and use the code `map_data`
France <- map_data('france')

# Define the position of names of province
center <- aggregate(cbind(long, lat) ~ region, data = France,
                    FUN=function(x) mean(range(x)))

# Graph the map
ggplot() +
  geom_polygon(data = France, aes(x = long, y = lat), group = France$group, fill = 'dodgerblue4', alpha = 0.7) +
  coord_fixed(1.2) +

  # add title
  ggtitle('Map of France') +

  # add names of provinces
  geom_text(data = center, aes(long, lat, label = region), size = 2)
```
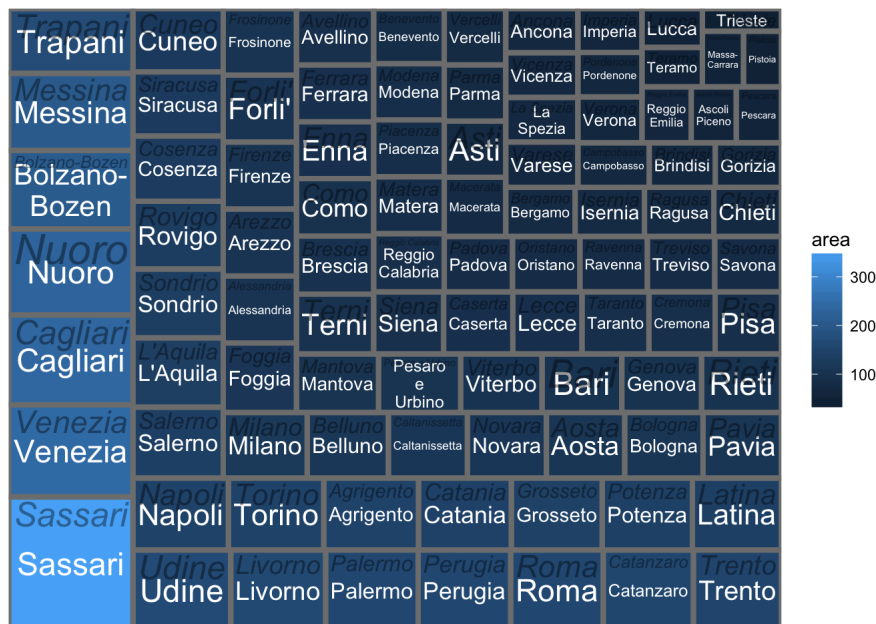
Map of France

This is the map of France. It has names of various provinces on it.

Now, let's convert the map of France into treemap.

```r
# Define FranceData
FranceData <- data.frame(summarise(group_by(France, region), area = length(region)))

# Use ggplot and geom_treemap to transform map into treemap in terms of area
ggplot(FranceData, aes(area = area, fill = area, label = region,
                subgroup = region)) +
  geom_treemap() +
  geom_treemap_subgroup_border() +

  # add names of provinces at the middle of each box
  geom_treemap_text(colour = "white", place = "centre", reflow = T) +

  # add names of provinces as background
  geom_treemap_subgroup_text(place = "topright", grow = T, alpha = 0.4, colour =
                        "black", fontface = "italic", min.size = 0)
```


This is the treemap of France.

Now, let's draw the map of Italy.

```
# Define Italy as the data 'italy' in the package `maps`, and use the code `map_data`
Italy <- map_data('italy')

# Define the position of names of province
center <- aggregate(cbind(long, lat) ~ region, data = Italy,
                    FUN=function(x) mean(range(x)))

# Graph the map
ggplot() +
  geom_polygon(data = Italy, aes(x = long, y = lat), group = Italy$group, fill = 'dodgerblue4', alpha = 0.7) +
  coord_fixed(1.2) +

  # add title
  ggtitle('Map of Italy') +

  # add names of provinces
  geom_text(data = center, aes(long, lat, label = region), size = 2)
```


Map of Italy

This is the map of Italy It has names of various provinces on it.

Now, let's convert the map of Italy into treemap.

```
# Define ItalyData
ItalyData <- data.frame(summarise(group_by(Italy, region), area = length(region)))

# Use ggplot and geom_treemap to transform map into treemap in terms of area
ggplot(ItalyData, aes(area = area, fill = area, label = region,
               subgroup = region)) +
  geom_treemap() +
  geom_treemap_subgroup_border() +

  # add names of provinces at the middle of each box
  geom_treemap_text(colour = "white", place = "centre", reflow = T) +

  # add names of provinces as background
  geom_treemap_subgroup_text(place = "topright", grow = T, alpha = 0.4, colour =
                            "black", fontface = "italic", min.size = 0)
```

This is the treemap of Italy.
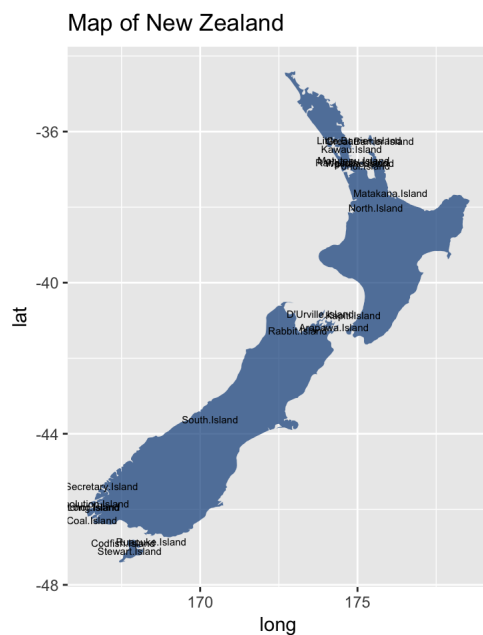
Now, let's draw the map of New Zealand.

```r
# Define NewZealand as the data 'nz' in the package `maps`, and use the code `map_data`
NewZealand <- map_data('nz')

# Define the position of names of islands
center <- aggregate(cbind(long, lat) ~ region, data = NewZealand,
                    FUN=function(x) mean(range(x)))

# Graph the map
ggplot() +
  geom_polygon(data = NewZealand, aes(x = long, y = lat), group = NewZealand$group, fill = 'dodgerblue4', alpha =
0.7) +
  coord_fixed(1.2) +

  # add title
  ggtitle('Map of New Zealand') +

  # add names of provinces
  geom_text(data = center, aes(long, lat, label = region), size = 2)
```


This is the map of New Zealand It has names of various islands on it.

Now, let's convert the map of New Zealand into treemap.

```
# Define NZData
NZData <- data.frame(summarise(group_by(NewZealand, region), area = length(region)))

# Use ggplot and geom_treemap to transform map into treemap in terms of area
ggplot(NZData, aes(area = area, fill = area, label = region,
               subgroup = region)) +
  geom_treemap() +
  geom_treemap_subgroup_border() +

  # add names of provinces at the middle of each box
  geom_treemap_text(colour = "white", place = "centre", reflow = T, min.size = 0) +

  # add names of provinces as background
  geom_treemap_subgroup_text(place = "topright", grow = T, alpha = 0.4, colour =
                             "black", fontface = "italic", min.size = 0)
```
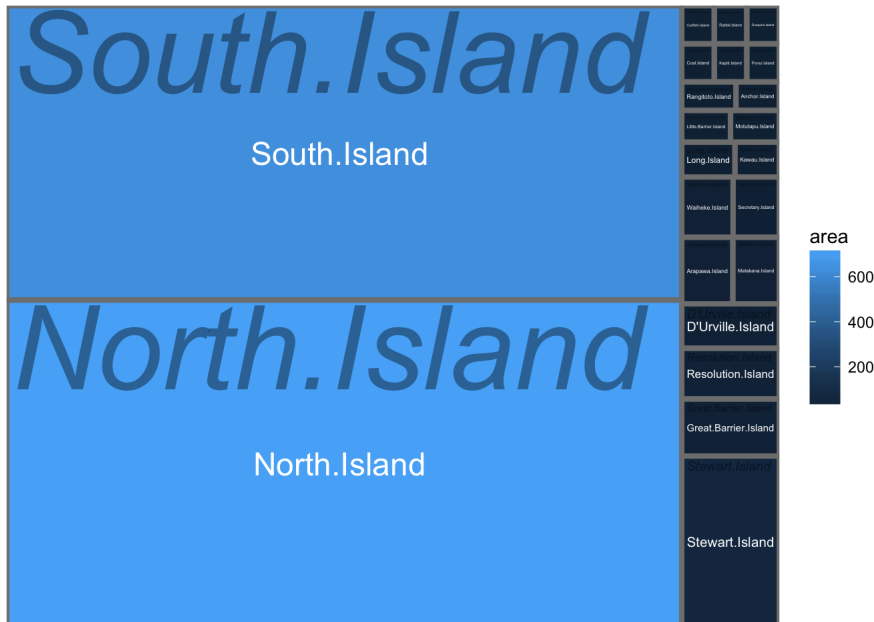

This is the treemap of New Zealand.

Now, let's give different islands of New Zealand different colors.

```
# Use ggplot and geom_treemap to transform map into treemap in terms of area
# Give differnt colors to differnt islands
ggplot(NZData, aes(area = area, fill = region, label = region,
               subgroup = region)) +
  geom_treemap() +

  # add names of provinces at the middle of each box
  geom_treemap_text(colour = "white", place = "centre", reflow = T, min.size = 0)
```


This is the treemap of the islands of New Zealand with different colors.

4. Conclusion

This post tells you how to convert maps into treemaps using the package `maps`, `ggplot2`, `treemapify`, and `dplyr`. We can use `geom_treemap` to draw a treemap and use `geom_treemap_text` to add text on the treemaps. We can also use `geom_treemap_subgroup_text` to add background text in order to glorify the treemaps. In addition, we can use the syntax `fill = region` to give different provinces/regions different colors. After you read the post, you will be able to convert a map into a beautiful treemap.

5. Reference

1. http://ggvis.rstudio.com/0.1/quick-examples.html
2. https://cran.r-project.org/web/packages/maps/maps.pdf
3. https://cran.r-project.org/web/packages/treemapify/treemapify.pdf
4. https://github.com/wilkox/treemapify
5. http://www.stat.columbia.edu/~tzheng/files/Rcolor.pdf
6. https://github.com/ucb-stat133/stat133-fall-2017/blob/master/cheat-sheets/ggplot2-cheatsheet-2.1.pdf
7. https://cran.r-project.org/web/packages/ggplot2/ggplot2.pdf

1. http://ggvis.rstudio.com/0.1/quick-examples.html
2. https://cran.r-project.org/web/packages/maps/maps.pdf
3. https://cran.r-project.org/web/packages/treemapify/treemapify.pdf
4. https://github.com/wilkox/treemapify