Using the gt-package and rayshader with the ChainLadder-package

Andreas Slåttelid

Table of Contents

The idea here is to show, how the gt-package could be used in combination with the ChainLadder-package in R. What makes this difficult is the transformation from the socalled “triangle”-object in ChainLadder, to a Tidyverse-dataframe.

Furthermore, we will also see how one could plot the chainladder triangle in 3d.

# ChainLadder

library(tidyverse) # wrangeling of data + %>%  
library(gt) # Creating runoff triangles  
library(ChainLadder) # contains the RAA dataset  
library(rayshader) # 3d-plot of triangle  
library(viridis) # coloring of heatmap  
library(viridisLite) # coloring of heatmap

In the [ChainLadder-package](https://cran.r-project.org/web/packages/ChainLadder/vignettes/ChainLadder.html), there is a data set called RAA, the goal will be to go from:

triangle <- RAA  
triangle

## dev  
## origin 1 2 3 4 5 6 7 8 9 10  
## 1981 5012 8269 10907 11805 13539 16181 18009 18608 18662 18834  
## 1982 106 4285 5396 10666 13782 15599 15496 16169 16704 NA  
## 1983 3410 8992 13873 16141 18735 22214 22863 23466 NA NA  
## 1984 5655 11555 15766 21266 23425 26083 27067 NA NA NA  
## 1985 1092 9565 15836 22169 25955 26180 NA NA NA NA  
## 1986 1513 6445 11702 12935 15852 NA NA NA NA NA  
## 1987 557 4020 10946 12314 NA NA NA NA NA NA  
## 1988 1351 6947 13112 NA NA NA NA NA NA NA  
## 1989 3133 5395 NA NA NA NA NA NA NA NA  
## 1990 2063 NA NA NA NA NA NA NA NA NA

And end up with the table displayed on the **last page** :)

The gt-package is all about formatting tables nicely, and it works very good with dataframes, however, we have:

class(RAA)

## [1] "triangle" "matrix"

namely a triangle/matrix object, the job will therefore be to transform objects from triangles to dataframes, which is not as obvious as one should expect

# Transformation process

triangle\_tibble <- triangle %>% as\_tibble() %>% cbind(rownames(triangle), .)  
  
#corresponds to development years j: 1,2, ...  
development\_years <- colnames(triangle\_tibble)[2:length(colnames(triangle\_tibble))]  
  
colnames(triangle\_tibble) <- c("Accident year i", development\_years)  
  
#select everything except "Accident year"  
tmp <- triangle\_tibble %>%   
 select(-"Accident year i") %>%   
 as.character()  
  
#matrix of zeros  
m <- matrix(0, ncol = length(development\_years),   
 nrow = length(rownames(triangle)))  
  
#append the data from traingle object to matrix  
for (i in 1:length(development\_years)){  
 m[i, ] <- eval(parse(text = tmp[i])) #evaluate vector camouflaged as character.  
}  
m

## [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]  
## [1,] 5012 106 3410 5655 1092 1513 557 1351 3133 2063  
## [2,] 8269 4285 8992 11555 9565 6445 4020 6947 5395 NA  
## [3,] 10907 5396 13873 15766 15836 11702 10946 13112 NA NA  
## [4,] 11805 10666 16141 21266 22169 12935 12314 NA NA NA  
## [5,] 13539 13782 18735 23425 25955 15852 NA NA NA NA  
## [6,] 16181 15599 22214 26083 26180 NA NA NA NA NA  
## [7,] 18009 15496 22863 27067 NA NA NA NA NA NA  
## [8,] 18608 16169 23466 NA NA NA NA NA NA NA  
## [9,] 18662 16704 NA NA NA NA NA NA NA NA  
## [10,] 18834 NA NA NA NA NA NA NA NA NA

#store the matrix as a dataframe  
df <- data.frame(m)  
colnames(df)

## [1] "X1" "X2" "X3" "X4" "X5" "X6" "X7" "X8" "X9" "X10"

#include the accident years:  
df <- df %>%   
 mutate(year = rownames(RAA), .before = X1)  
  
colnames(df) <- c("year", development\_years)

# Display ChainLadder as gt-table

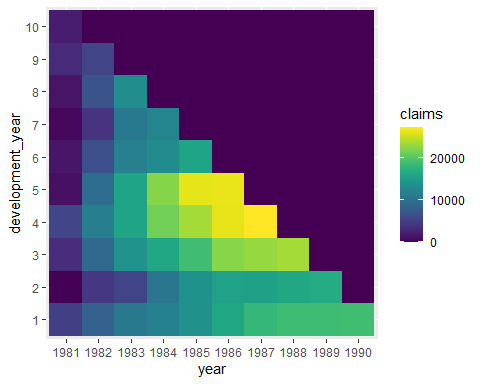
maximum\_color\_value <- max(m, na.rm = TRUE)  
  
columns\_formatted <- colnames(df)[2:length(colnames(df))]  
  
gt\_triangle <- df %>%   
 gt(rowname\_col = "year") %>%   
 tab\_stubhead(label = "Accident year i") %>%   
 fmt\_number(columns = columns\_formatted,   
 decimals = 0,  
 accounting = TRUE) %>%   
 tab\_spanner(  
 label = "Development year j",  
 columns = all\_of(colnames(df))  
 ) %>%   
 sub\_missing(missing\_text = "-") %>%   
 data\_color(  
 columns = columns\_formatted,  
 colors = scales::col\_numeric(  
 palette = c("blue", "green", "orange", "red"),  
 domain = c(0, maximum\_color\_value)  
 )  
 )  
  
gt\_triangle

| Accident year i | Development year j |
| --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1981 | 5,012 | 106 | 3,410 | 5,655 | 1,092 | 1,513 | 557 | 1,351 | 3,133 | 2,063 |
| 1982 | 8,269 | 4,285 | 8,992 | 11,555 | 9,565 | 6,445 | 4,020 | 6,947 | 5,395 | - |
| 1983 | 10,907 | 5,396 | 13,873 | 15,766 | 15,836 | 11,702 | 10,946 | 13,112 | - | - |
| 1984 | 11,805 | 10,666 | 16,141 | 21,266 | 22,169 | 12,935 | 12,314 | - | - | - |
| 1985 | 13,539 | 13,782 | 18,735 | 23,425 | 25,955 | 15,852 | - | - | - | - |
| 1986 | 16,181 | 15,599 | 22,214 | 26,083 | 26,180 | - | - | - | - | - |
| 1987 | 18,009 | 15,496 | 22,863 | 27,067 | - | - | - | - | - | - |
| 1988 | 18,608 | 16,169 | 23,466 | - | - | - | - | - | - | - |
| 1989 | 18,662 | 16,704 | - | - | - | - | - | - | - | - |
| 1990 | 18,834 | - | - | - | - | - | - | - | - | - |

# 3D-plot of ChainLadder

#replace NA's with 0  
df2 <- data.frame(m)  
df2[is.na(df2)] <- 0  
  
df2 <- df2 %>%   
 mutate(year = rownames(RAA), .before = X1)  
  
colnames(df2) <- c("year", colnames(RAA))

df2\_long <- df2 %>%  
 pivot\_longer(!year, names\_to = "development\_year", values\_to = "claims") %>%   
 mutate(development\_year\_helper = as.numeric(development\_year))  
  
#sort properly development year, had 1, 10, 2, 3, ..., 9  
df2\_long$development\_year <- reorder(df2\_long$development\_year, df2\_long$development\_year\_helper)  
  
triangle\_dim2\_plt <- ggplot(df2\_long, aes(year, development\_year, fill= claims)) +  
 geom\_tile() +   
 scale\_fill\_viridis()  
triangle\_dim2\_plt



Using the [rayshader-package](https://www.rayshader.com/reference/plot_gg.html#examples), one can easily go from a 2d-ggplot to a 3d-plot.

#triangle\_dim3\_plt <- rayshader::plot\_gg(triangle\_dim2\_plt,  
# width = 5,   
# height = 5)