animbook: Visualizing changes in performance measures and demographic affiliations using animation

by Krisanat Anukarnsakulchularp

Abstract An abstract of less than 150 words.

1 Introduction

The concept of "zombie companies" began to attract attention when an article on the proliferation of zombie companies (Caballero, Hoshi, and Kashyap (2008)). The "zombie companies" are generally defined as companies with an interest coverage ratio of less than one for a period of more than three years. However, there is a simpler and more easily understandable way to show these concepts by visualizing the new listings (enters) and the de-listing (exits) of publicly traded companies on a country-by-country basis. This visualization method makes it clear that the concepts of zombie companies are not unique to Japan, as indicated in the OECD report (McGowan, Andrews, and Millot (2017)) that the United States has a faster metabolize (more new listings and exits) relative to Japan.

The visualization above can also be thought of as a movement between groups, which is how many companies have entered the market and how many have exited the market. One example of this visualization is from The New York Times article "Extensive data shows punishing reach of racism for black boys" (Badger et al. (2018)). The animation portrays the trajectory of how boys with different demographics could land up in society. It can be adapted to our problems by capturing the movement between the groups over a specific period.

With an advancement in technology, this results in an increase in both the size and complexity of the data. It requires an experienced hand to convey the right messages from the data. However, as shown in The New York Times article (Badger et al. (2018)), the animation is not only created just for eye-catching graphics. Animation can be used as a tool that helps communicate complex data, enhancing the narrative and keeping it engaged for the audience. Based on Mayer and Moreno (2002), animation can improve learning, especially when the goal is to promote deep understanding. It also requires the designer to understand how people learn. The cognitive theory of multimedia learning (Mayer (2005)) will be considered during the designing period.

The challenge of the New York Times article animation was the amount of code needed to reproduce. Additionally, the business data may or may not have the same information as the one in the article. These two purposes establish the objective for creating the R package that could generalize the animation to a wider range of data.

The structure of this paper will consist of both the visualization design and software design. The first section will explain about the animation in The New York Times article. Secondly, what are the expected data to be input into the plot, and how it is processed. The third section is about the animation tools, for example, **gganimate** (Pedersen and Robinson (2020)) and **plotly** (Sievert (2020)). The next section will be about the design of the package and visualization. The last two sections explore the usage of the R package and the application.

2 Explanation of the New York Times visualization

The flow chart featured in one of the New York Time articles unveils the issue of income disparities between black and white children who were raised in families with comparable income according to the Chetty et al. (2020). This visualization reveals that, compared to white children, black children are more likely to drop down to the lower-income group, given that they both grew up in wealthy families.

In the visualization, each observation is initially classified into one group at the start and potentially transitions into either the same group or a different group. This dynamics visualization constructs questions on the broader use of this visualization to other types of data. As mentioned in the introduction, one of the data that will make use of this is accounting data. Additionally, the marketing data that exhibits the movement of customers shifting the product interest to another competitor can be valuable insight into the data. It could be extended to incorporate demographic information of the customers. It would allow the marketing analyst to gain a significant understanding of both the

company's products and the overall market. This also applies to the election data for the analyst to consolidate the campaign for their political party.

This animation was developed using two software based on JavaScript, D3.js (Bostock (2012)), and WebGL (reference). The D3 JavaScript is one of the most widely known libraries for creating an interactive and dynamic visualization. It enables the designers to bind both the data and graphical elements to the DOM (Document Object Model). On the other hand, WebGL functions as a JavaScript API for rendering interactive 2D and 3D graphics within any compatible web browser without the use of plug-ins. For the animation in this paper, the programming languages that will be used for recreating and revising the visualization done by The New York Times articles is R (R Core Team (2021)).

3 Data

Before any visualizations can be performed, the users must first understand the data concept known as `tidy data` (Wickham (2014)). There are three fundamental key principles: each variable forms a column, each observation forms a row, and each type of observational unit forms a table. These three keys are Codd's third normal form (Codd (1990)), but phrase in the language of statistics. The tidy data format focuses on a single dataset instead of many connected datasets typically found in a relational database.

The accounting data that will be used in this section and in most of the examples for this paper was collected from Bureau van Dijk (reference). This data set comprises 30,000 rows and 94 variables of information on listed and major unlisted/delisted companies worldwide. The only variables of interest from this data set are ID, year, country, and sales. A subset version of this data set, which only contained variables of interest from 2006 to 2018, is included in this package.

Now that the tidy data concept is established and the example data is introduced, let's explore how the data got transformed. As seen in the visualization, each observation is classified into a group. In the accounting data, however, it is represented in a numerical form instead of categories. The numerical value needs to be somehow mapped to a category. One way to handle this is by ranking the sales and grouping the rankings into quantiles. In some cases, this may not be the best option. As the observation is moved up by quantiles, one is bound to move down. It can be resolved by using an alternative method, which is grouping values based on their absolute values. Users may also be interested in grouping the data based on different demographics, for example, ranking within a specific country. This generalization leads to a total of four scaling methods for the numerical data.

Original data:

```
#> # A tibble: 12 x 4
#>    ID           year    sales country
#>    <fct>       <int>    <dbl> <fct>
#>  1 AU004085330  2007  1043282 AU
#>  2 AU004085330  2006   846245 AU
#>  3 AU009134114  2006    67842 AU
#>  4 AU009134114  2007    47868 AU
#>  5 AU009219809  2007   326335 AU
#>  6 AU009219809  2006   218384 AU
#>  7 US161229730  2007     5991 US
#>  8 US161229730  2006     4308 US
#>  9 US470731996  2007  8126500 US
#> 10 US470731996  2006  7205000 US
#> 11 US751825172  2007 22935000 US
#> 12 US751825172  2006 22563000 US
```

1. Ranking by year.

```
#> # A tibble: 12 x 4
#>    id           time qtile country
#>    <fct>       <int> <dbl> <fct>
#>  1 AU004085330  2006     2 AU
#>  2 AU004085330  2007     2 AU
#>  3 AU009134114  2006     3 AU
#>  4 AU009134114  2007     3 AU
#>  5 AU009219809  2006     3 AU
#>  6 AU009219809  2007     2 AU
```

```
#>  7 US161229730  2006     4 US
#>  8 US161229730  2007     4 US
#>  9 US470731996  2006     1 US
#> 10 US470731996  2007     1 US
#> 11 US751825172  2006     1 US
#> 12 US751825172  2007     1 US
```

2. Fix bins relative to absolute values by year.

```
#> # A tibble: 12 x 4
#>    id           time qtile country
#>    <fct>       <int> <dbl> <fct>
#>  1 AU004085330  2006     5 AU
#>  2 AU004085330  2007     5 AU
#>  3 AU009134114  2006     5 AU
#>  4 AU009134114  2007     5 AU
#>  5 AU009219809  2006     5 AU
#>  6 AU009219809  2007     5 AU
#>  7 US161229730  2006     5 US
#>  8 US161229730  2007     5 US
#>  9 US470731996  2006     4 US
#> 10 US470731996  2007     4 US
#> 11 US751825172  2006     1 US
#> 12 US751825172  2007     1 US
```

3. Ranking by year within a group.

```
#> # A tibble: 12 x 4
#>    id           time qtile country
#>    <fct>       <int> <dbl> <fct>
#>  1 AU004085330  2006     4 AU
#>  2 AU004085330  2007     4 AU
#>  3 AU009134114  2006     4 AU
#>  4 AU009134114  2007     4 AU
#>  5 AU009219809  2006     4 AU
#>  6 AU009219809  2007     4 AU
#>  7 US161229730  2006     4 US
#>  8 US161229730  2007     3 US
#>  9 US470731996  2006     1 US
#> 10 US470731996  2007     1 US
#> 11 US751825172  2006     1 US
#> 12 US751825172  2007     1 US
```

4. Fix bins relative to absolute values by year within a group (bug in code).

```
#> # A tibble: 12 x 4
#>    id           time qtile country
#>    <fct>       <int> <dbl> <fct>
#>  1 AU004085330  2006     5 AU
#>  2 AU004085330  2007     5 AU
#>  3 AU009134114  2006     5 AU
#>  4 AU009134114  2007     5 AU
#>  5 AU009219809  2006     5 AU
#>  6 AU009219809  2007     5 AU
#>  7 US161229730  2006     5 US
#>  8 US161229730  2007     5 US
#>  9 US470731996  2006     4 US
#> 10 US470731996  2007     4 US
#> 11 US751825172  2006     1 US
#> 12 US751825172  2007     1 US
```

For the first and third scaling methods, group splitting is executed using the `quantiles()` and `cut()` functions. The `quantile` function from the `stats` R package (R Core Team (2013)) takes a numeric vector and outputs the corresponding quantiles to the given probabilities. The output from

the `quantile` function is then used as the `breaks` argument for the cut function that is part of the base R packages.

In contrast, the second and fourth scaling methods calculate the quantile based on the absolute values scales. The default approach is to break the group equally using the `seq()` function. The `seq()` function takes input values from the minimum and maximum values and increments by equal steps depending on the number of groups of interest.

Based on Figure 1, these are only the initial steps in formatting the data into a category. Now that there is a method to transform the data from the raw into a categorized format, the next step is to modify it into an animbook data structure. It is carried out by assigning the frame to each individual observation, ensuring that each ID does not contain repeat frame values. Its lets the **gganimate** or **plotly** to perceive where the observation would be on the plot at a given frame.

The frame variable is assigned by sorting the data based on the id and time using the `arrange()` function, followed by applying the `group_by()` function on the id, allowing the `row_number()` function to be performed within each group. The functions mentioned in this paragraph are from the `dplyr` packages (Wickham et al. (2023)).

4 Animation tools

Literature review of animation tools

5 Visualization design

This explains how to get from data to the animations, including different sorts of plots.

6 Software

Installation

Overview of functions

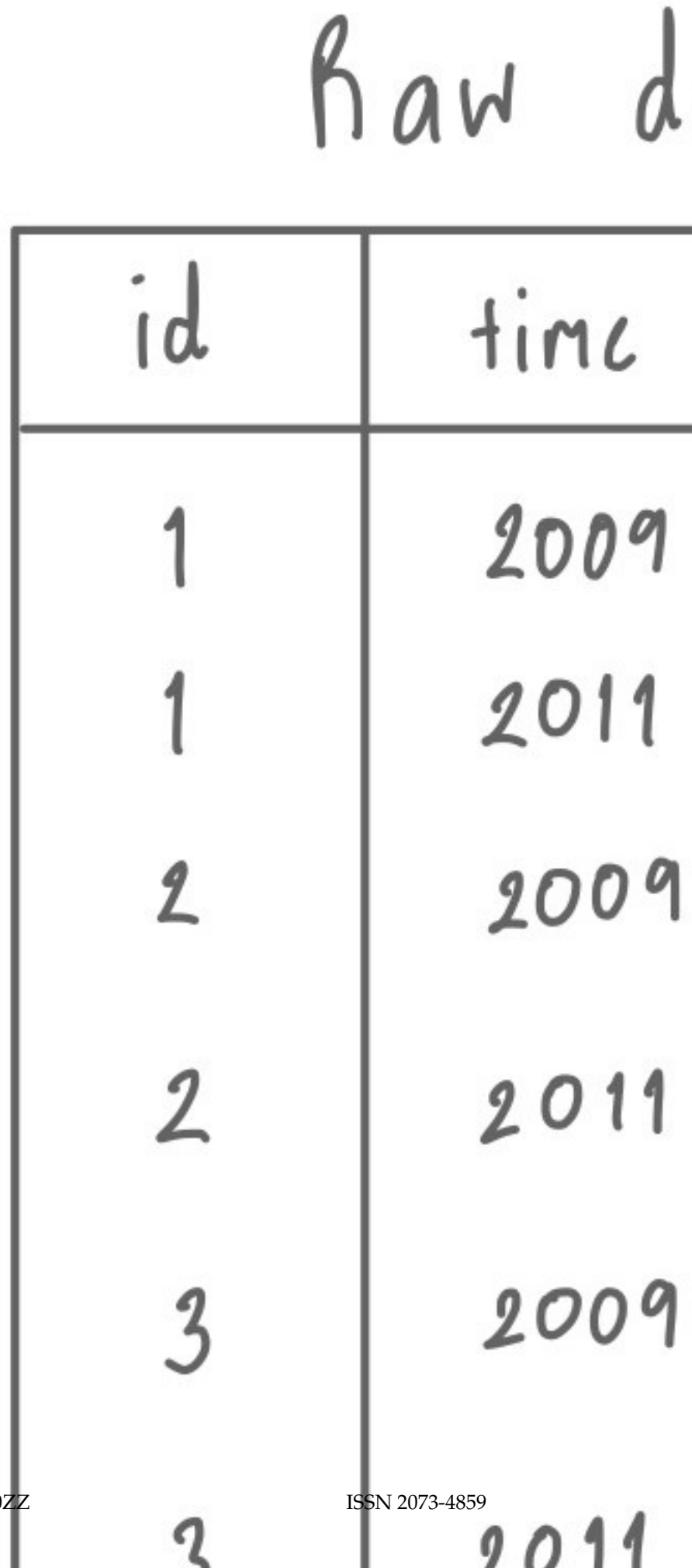
In designing the package for reproducing the New York Times animated plot, the package ended up with a three-step process in recreating the animation. The first step is to turn the data into the right format for the plot function. The next stage is to create a ggplot object, which can then be inputted into the animation function. The last step is adding the animation settings to the ggplot object so the user can animate the plot using the `gganimate::animate()` function. The reason for this three-step process is that it allows the user who does not have a lot of experience to reproduce the animation while keeping the customization for an experienced user.

Prepare the data

The first step can be done using the `anim_prep()` function. This function required that the user data contained ID, which is used to identify each individual observation, values, which are numerical values to be used to group the observation together, and time, the variable associated with the changes of the observation. In the cases when the users already have the values as a group variable, the `anim_prep_cat()` function can then be used instead.

The additional options for the `anim_prep()` that allow for more customization to the data structure or the plot are as follows:

- label: group labeling.
- ngroup: number of groups we want to split the values into.
- breaks: the group bins' size
- group_scaling: the grouping variable for the bins calculations.
- color: the variable used to color the observations.
- time_dependent: logical. Whether we want the observations to start at the same time or not.
- scaling: the scaling method, either `rank` or `absolute`.
- runif_min: minimum value for random addition to frame numbers
- runif_max: maximum value for random addition to frame numbers


Raw d
id
time
1
2009
1
2011
2
2009
2
2011
3
2009
3
2011

Using the different combinations of the additional options, the users could end up with four different scales, as mentioned in the 2.3 section.

1. Rank scaling
2. Absolute scaling
3. Rank scaling by group
4. Absolute scaling by group

For the `anim_prep_cat()` function, the additional options are:

- label: group labeling.
- order: the ordering of the group.
- color: the variable used to color the observations.
- time_dependent: logical. Whether we want the observations to start at the same time or not.
- runif_min: minimum value for random addition to frame numbers
- runif_max: maximum value for random addition to frame numbers

Both the `anim_prep()` and `anim_prep_cat()` functions will return the "animbook" object containing a list of the standard format data and settings.

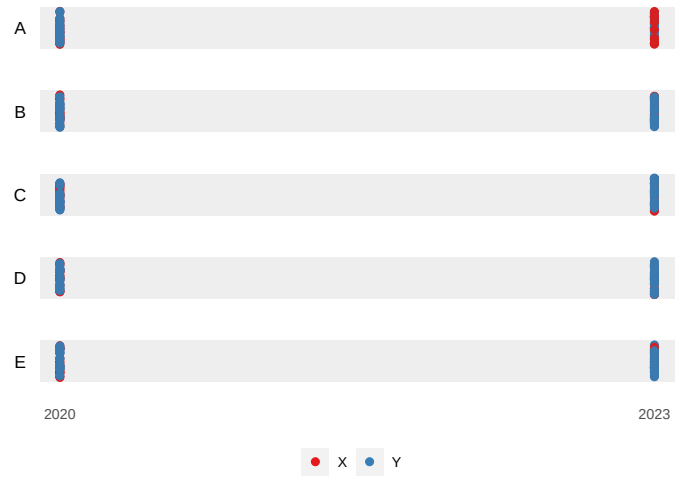
Plotting the data

Once the data is prepared. The next step is to create the ggplot object as a basis for the animation. There are three plots available in this package. Two of the plots could be used for the animation, and another plot is used as a static visualization.

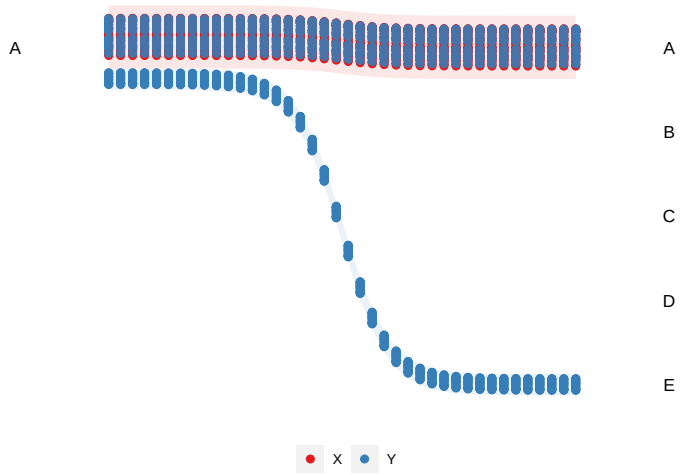
- `kangaroo_plot()`: plots the observation's movement over time.
- `wallaby_plot()`: the subset plot of the `kangaroo_plot` with the time limit to only start and end.
- `funnel_web_plot()`: the faceted static plot by time variable.

All of the plots have an internal function that converts the standard data format into the required structure for each plotting function.

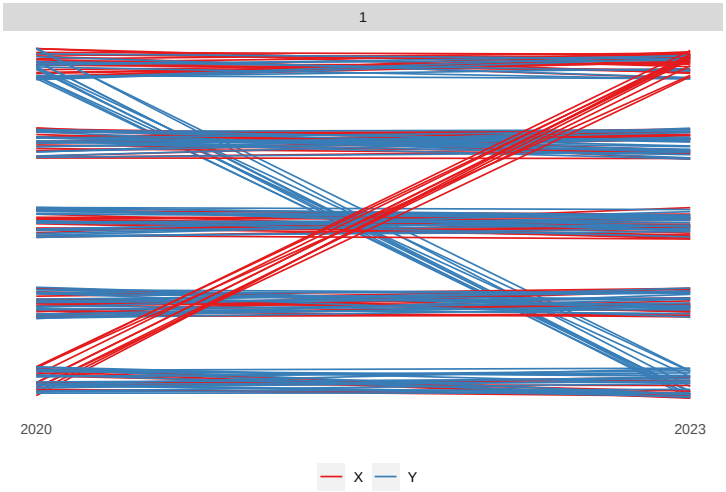
1. Kangaroo's plot



2. Wallaby's plot



3. Funnel web spider's plot



Animating the plot

To animate the plot, the plot need to be save as an object before passed on to the final function `anim_animate()`.

Example usage

7 Application

Accounting database: osiris

Voter behavior

Based on the 2016 Australian election results, how does the top party perform in keeping the old voters for different genders.

References

Badger, Emily, Claire Cain Miller, Adam Pearce, and Kevin Quealy. 2018. "Extensive Data Shows Punishing Reach of Racism for Black Boys." *The New York Times*. The New York Times. https://www.nytimes.com/interactive/2018/03/19/upshot/race-class-white-and-black-men.html.

Bostock, Mike. 2012. "D3.js - Data-Driven Documents." 2012. http://d3js.org/.

Caballero, Ricardo J., Takeo Hoshi, and Anil K. Kashyap. 2008. "Zombie Lending and Depressed Restructuring in Japan." *The American Economic Review* 98 (5): 1943–77. http://www.jstor.org/stable/29730158.

Chetty, Raj, Nathaniel Hendren, Maggie Jones, and Sonya Porter. 2020. "Race and Economic Opportunity in the United States: An Intergenerational Perspective*." *The Quarterly Journal of Economics* 135 (May): 711–83. https://doi.org/10.1093/qje/qjz042.

Codd, E. F. 1990. *The Relational Model for Database Management: Version 2*. USA: Addison-Wesley Longman Publishing Co., Inc.

Mayer, Richard E. 2005. "Cognitive Theory of Multimedia Learning." In *The Cambridge Handbook of Multimedia Learning*, edited by RichardEditor Mayer, 31–48. Cambridge Handbooks in Psychology. Cambridge University Press. https://doi.org/10.1017/CBO9780511816819.004.

Mayer, Richard E., and Roxana Moreno. 2002. *Educational Psychology Review* 14 (1): 87–99. https://doi.org/10.1023/a:1013184611077.

McGowan, Müge Adalet, Dan Andrews, and Valentine Millot. 2017. "The Walking Dead?" no. 1372. https://doi.org/https://doi.org/https://doi.org/10.1787/180d80ad-en.

Pedersen, Thomas Lin, and David Robinson. 2020. "Gganimate: A Grammar of Animated Graphics." *R Package Version* 1 (7): 403–8.

R Core Team. 2013. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. http://www.R-project.org/.

———. 2021. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Sievert, Carson. 2020. "Interactive Web-Based Data Visualization with r, Plotly, and Shiny." Chapman; Hall/CRC. https://plotly-r.com.

Wickham, Hadley. 2014. "Tidy Data." *Journal of Statistical Software* 59 (10): 1–23. https://doi.org/10.18637/jss.v059.i10.

Wickham, Hadley, Romain François, Lionel Henry, Kirill Müller, and Davis Vaughan. 2023. *Dplyr: A Grammar of Data Manipulation*.

Krisanat Anukarnsakulchularp
Monash University
Faculty of Business and Economics
Melbourne, Australia
ORCiD: 0009-0008-5638-7124
kanu0003@student.monash.edu