User Guide for prettycloud Package

Peng Miao[‡,*], Xiaoyong Sun[‡,†]

June 19, 2015

[‡]Agricultural Big-Data Research Center
College of Information Science and Engineering
Shandong Agricultural University
Taian, Shandong 271018, China

Contents

1 Introduction **1**

2 Installation **2**

3 Algorithms for Prettycloud **2**
3.1 Folium of Descartes 2
3.2 Involute of a circle 3
3.3 Two Hyperbolas 3
3.4 Tricuspoid 4

4 Function **4**

5 Implementation **4**
5.1 Parameter Declaration 4
5.2 Examples 5

1 Introduction

wordcloud is a a graphical representation that is used to draw beautiful words. It can control the word font as well as the word color accordingly. Generally this grahical approach arranges the word either horizontally or vertically, which lacks flexibility.

prettycloud package takes a different approach by relying on the curve function to control the layout of the words. Currently it supports 4 kinds of curves to display the words following interesting graphical patterns. In the near future, we will use additional curve functions to provide more choices for users.

[*]aaa132452@outlook.com
[†]johnsunx1@gmail.com, johnsunx1@126.com

This package provides a few arguments such as group, weight, maxsize and minsize to control the graphical layout. User can also set the background color and font color as well. prettycloud is a very handy tool for drawing words with appropriate weight and group.

2 Installation

Please install:

1. Install packages from CRAN
`install.packages("prettycloud")`

3 Algorithms for Prettycloud

In this section, we will discuss the main algorithms to control the layout of the words.

3.1 Folium of Descartes

Cartesian equation:

$$x^3 + y^3 = 3axy \tag{1}$$

or parametrically:

$$x = \frac{3at}{1+t^3}, y = \frac{3at^2}{1+t^3} \tag{2}$$

The graph is Figure 1.

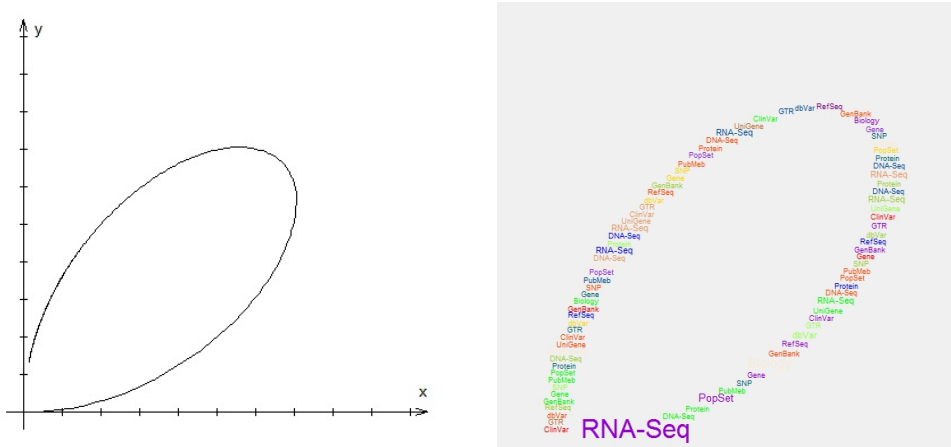


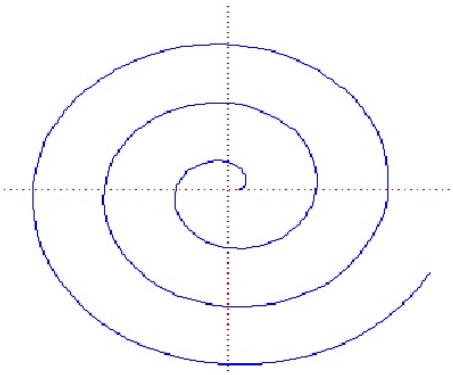
Figure 1: Left figure: match curve. Right figure: word cloud.

3.2 Involute of a circle

Parametric Cartesian equation:

$$x = a(cos(t) + sin(t)), y = a(sin(t) - tcos(t)) \tag{3}$$

The graph is Figure 2.



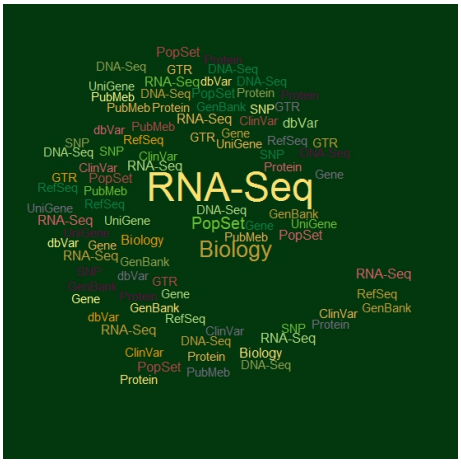


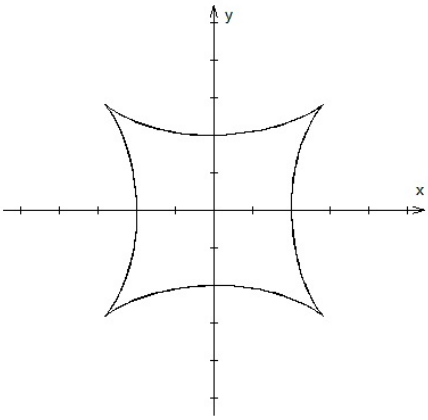
Figure 2: Left figure: match curve. Right figure: word cloud.

3.3 Two Hyperbolas

Parametric Cartesian equation:

$$x = a(3cos(t) - cos(3t)), y = a(3sin(t) + sin(3t)) \tag{4}$$

The graph is Figure 3.



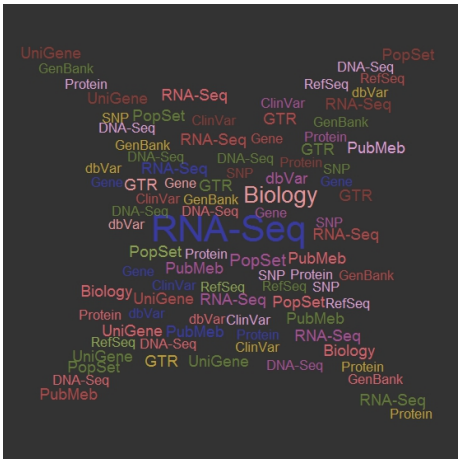


Figure 3: Left figure: match curve. Right figure: word cloud.

3.4 Tricuspoid

Cartesian equation:

$$(x^2 + y^2 + 12ax + 9a^2)^2 = 4a(2x + 3a)^3 \tag{5}$$

or parametrically:

$$x = a(2cos(t) + cos(2t)), y = a(2sin(t) - sin(2t)) \tag{6}$$

The graph is Figure 4.

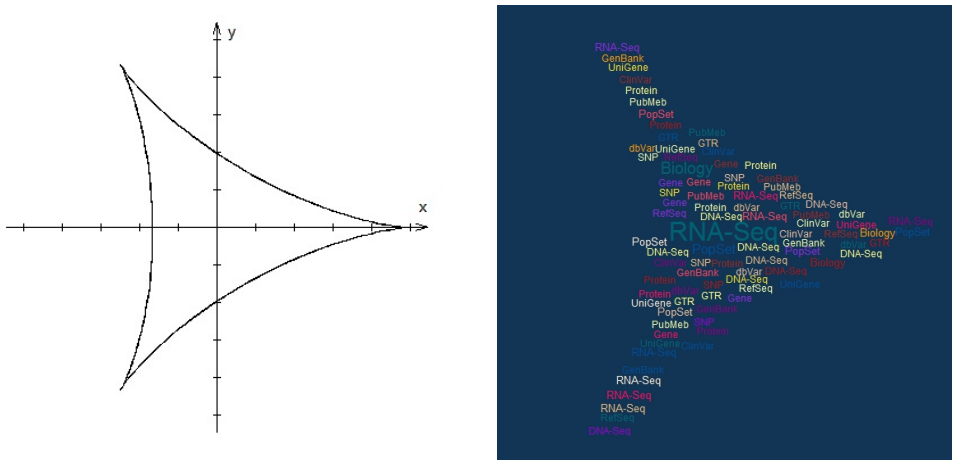


Figure 4: Left figure: match curve. Right figure: word cloud.

4 Function

```
prettycloud <- function(words,weight,background="null",fontcolor="null",
group="null",style="null",maxsize="null",minsize="null",replication=FALSE)
```

5 Implementation

5.1 Parameter Declaration

There are eight parameters for users:

- **words**: The words you want to show.
- **weight**:The relative size of each word. Big weight has large font size.
- **background**: The background color
- **fontcolor**: The font color
- **group**: The words in the same group can be showed in same color. The default value is null(no grouping). If group is set, each word should be assigned with a group number.

- **style**: The choice from supported curve functions. The default value means random selection from curve functions.
- **maxsize**: The maximum value of fontsize(relative).If NULL, use default value.
- **minsize**: The minimum value of fontsize(relative).If NULL, use default value.
- **replication**: A logical value.If true, the words will be repeated in the figure.

5.2 Examples

The data file is showed in figure 5.

```
prettycloud(words=c(1:50),weight=c(1:50), style="1")
prettycloud(words=c(1:50),weight=c(1:50), style="2")
prettycloud(words=c(1:50),weight=c(1:50), style="3")
prettycloud(words=c(1:50),weight=c(1:50), style="4")
```

The result is in one of figure 1,2,3,4.

```
library(prettycloud)

#show 1 to 200
prettycloud(c(1:200), c(1:200))
#show 1 to 100 twice
prettycloud(c(1:100), c(1:100), replication=T)
#read data from csv and set maximum value and minimum value.
data(bioList)
words<-bioList$words
weight<-bioList$weight
prettycloud(words, weight, maxsize=1, minsize=0.1)
#group example
prettycloud(words=c(1:5),weight=c(1:5),group=c(1,1,2,2,3))
#style example(style 1 to 4)
prettycloud(words,weight,style="3")
#backgroud color and fontcolor
#Notice:If use background and fontcolor,the group will lose efficacy.
prettycloud(letters[1:5],c(1:5),background="#000000",
fontcolor=c("red","blue","#009ACD","green","white"),maxsize=4,minsize=1)
```

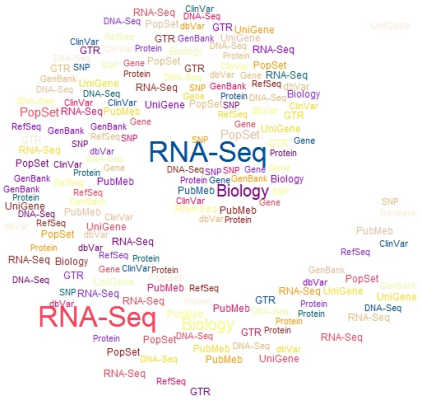


Figure 5: wordcloud for bioinformatics