Package: ChaosGame (via r-universe)

October 14, 2026

Type Package

Title Chaos Game

Version 1.5

Description The main objective of the package is to enter a word of at least two letters based on which an Iterated Function System with Probabilities is constructed, and a two-dimensional fractal containing the chosen word infinitely often is generated via the Chaos Game. Additionally, the package allows to project the two-dimensional fractal on several three-dimensional surfaces and to transform the fractal into another fractal with uniform marginals.

Depends R (>= 2.10), rgl, colorRamps

Imports ggplot2, gridExtra, plot3D, RColorBrewer,

License GPL-2

Encoding UTF-8

RoxygenNote 7.1.1

NeedsCompilation no

Maintainer Lea Maislinger <lea.maislinger@plus.ac.at>

Author Lea Maislinger [aut, cre], Thimo Kasper [aut], Florian Griessenberger [aut], Manuela Schreyer [aut], Johannes Bartel [aut], Wolfgang Trutschnig [aut]

Config/pak/sysreqs cmake libfreetype6-dev libglu1-mesa-dev make texlive libpng-dev libuv1-dev libgl1-mesa-dev zlib1g-dev

Repository https://leamais1.r-universe.dev

Date/Publication 2025-07-05 23:00:11 UTC

RemoteUrl https://github.com/cran/ChaosGame

RemoteRef HEAD

RemoteSha 78b781ed9a7ea2b43a78d56639cd104a1632506e

Contents

plot_word 2
plot_word3D 3

Index **6**

`plot_word` *Plot the 2D fractal containing the chosen word*

Description

The function allows to enter a word of at least two letters based on which an Iterated Function System with Probabilities (IFSP) is constructed. This IFSP is then used to generate a two-dimensional fractal containing the chosen word infinitely often, which is then plotted (and optionally probability-integral-transformed).

Usage

```
plot_word(
  word = "copula",
  R = 20,
  phi = 0,
  copula = FALSE,
  portion = 0.2,
  shift = 1.2,
  orbit = 3000,
  letter_type = 1
)
```

Arguments

`word`	Word which the fractal should contain infinitely often.
`R`	Number of runs of the chaos game.
`phi`	Angle of the rotation.
`copula`	logical. If TRUE the sample is (approximately) probability-integral-transformed.
`portion`	Portion based on which the empirical distribution functions are calculated, if `copula = TRUE`.
`shift`	Distance between letters.
`orbit`	Number of steps in each run of the chaos game.
`letter_type`	integer, which indicates the type of the letters. Options are 1 (default) or 2 (round letters).

Examples

```
#Function with word as input, constructs the IFSP and runs the chaos game:

# for nice results use, for example, R = 20 and orbit = 3000
A <- plot_word(word = "copula", R = 50, orbit = 100)
#plot without histograms of the marginal distributions
plot(A, pch =19, col = 4, cex = 0.1)

# further examples:
# with round letters
# A <- plot_word(word = "copula", R = 100, orbit = 300, letter_type = 2)
# with rotation
# A <- plot_word(word = "copula", R = 100, orbit = 300, phi = pi/8)
# A <- plot_word(word = "fractal", R = 100, orbit = 300, phi = pi/6)

# (approximately) probability-integral-transformed
# A <- plot_word(word = "copula", R = 100, orbit = 300, phi = pi/8, copula = TRUE)
# A <- plot_word(word = "fractal", R = 100, orbit = 300, phi = pi/6, copula = TRUE)
```

---

plot_word3D *Plot the 3D fractal containing the chosen word*

Description

The function allows to enter a word of at least two letters based on which an Iterated Function System with Probabilities (IFSP) is constructed. This IFSP is then used to generate a two-dimensional fractal containing the chosen word infinitely often, which is then projected onto several three-dimensional surfaces. Optionally, the projection is transformed into another fractal with uniform marginals.

Usage

```
plot_word3D(
  word = "copula",
  R = 20,
  plot.rgl = TRUE,
  copula = TRUE,
  portion = 0.2,
  color.rgl.plot = "green2magenta",
  plot.surface = "Sphere",
  histogram = TRUE,
  shift = 1.2,
  orbit = 3000,
  cex.label = 0.7,
  size.lines = 0.1,
  Theta = 40,
  Phi = 30,
  Box = TRUE,
```

```
  projection = TRUE,
  letter_type = 1
)
```

Arguments

word	Word which the fractal should contain infinitely often.
R	Number of runs of the chaos game.
plot.rgl	If plot.rgl = TRUE a rgl-plot is generated. Otherwise a scatter-plot with plot3D is produced.
copula	logical. If TRUE the sample is (approximately) probability-integral-transformed.
portion	Portion based on which the empirical distribution functions are calculated, if copula = TRUE.
color.rgl.plot	Plotting color/color-range for the rgl-plot. One can choose between "gray", "blue2green", "green2red", "blue2yellow", "ygobb", "magenta2green" and "green2magenta".
plot.surface	Three-dimensional surface on which the two-dimensional fractal is projected. Options are "Sphere", "Helix", "Torus", "EnneperMinimalSurface" and "CatalanSurface".
histogram	It is an option available only under the rgl-plot option (i.e if plot.rgl = TRUE). If histogram = TRUE, two-dimensional and one-dimensional marginal histograms are plotted in the rgl-plot.
shift	Distance between letters.
orbit	Number of steps in each run of the chaos game.
cex.label	Font size, for exporting as a pdf or png file (see examples).
size.lines	Line width, for exporting as a pdf or png file (see examples).
Theta	Angles defining the viewing direction. Theta gives the azimuthal direction and Phi the colatitude in the scatter-plot (see Package plot3D). Can be chosen only under the scatter-plot option (i.e if plot.rgl = FALSE).
Phi	Angles defining the viewing direction. Theta gives the azimuthal direction and Phi the colatitude in the scatter-plot (see Package plot3D). Can be chosen only under the scatter-plot option (i.e if plot.rgl = FALSE).
Box	If TRUE, axis, two-dimensional projections (if projection = TRUE) and marginal histograms are plotted.
projection	An option available only if Box = TRUE. If projection = TRUE, the two-dimensional projections are plotted together with axis and marginal histograms.
letter_type	integer, which indicates the type of the letters. Options are 1 (default) or 2 (round letters).

Examples

```
# function with a word as input, runs the chaos game,
# calculates the copula transformation and projects the result on a sphere:
# for nice results use, for example, R = 20 and orbit = 3000
```

```
# A <- plot_word3D(word = "copula", copula = FALSE, R = 50, orbit = 100)

# further examples:
# projection of the fractal on the Enneper Minimal Surface:
# A <- plot_word3D(word = "copula", R = 75, orbit = 300, copula = FALSE,
#                  plot.surface = "EnneperMinimalSurface", histogram = FALSE)
# same example as before, now with histogram = TRUE
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = FALSE,
#                  plot.surface = "EnneperMinimalSurface")
# same example as before (approximately) probability-integral-transformed (i.e. copula = TRUE)
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = TRUE,
#                   plot.surface = "EnneperMinimalSurface")

# projection of the fractal on a Catalan Surface
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = FALSE,
#                  color.rgl.plot = "blue2green", plot.surface = "CatalanSurface")

# projection of the fractal on a Helix
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = FALSE,
#                  color.rgl.plot = "green2red", plot.surface = "Helix")

# projection of the fractal on a Torus
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = FALSE,
#                  color.rgl.plot = "blue2yellow", plot.surface = "Torus")

# projection of the fractal on a Sphere
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = FALSE,
#                  color.rgl.plot = "ygobb", plot.surface = "Sphere")
# Sphere (approximately) probability-integral-transformed (i.e. copula = TRUE)
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = TRUE,
#                  color.rgl.plot = "ygobb", plot.surface = "Sphere")

# scatter-plot with plot3D (i.e. plot.rgl = FALSE) for exporting as a pdf file
# pdf(file = "Sphere.pdf", width = 30, height = 25)
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = FALSE, plot.rgl = FALSE,
#                  plot.surface = "Sphere", cex.label = 1.8, size.lines = 0.001)
# dev.off()

#' # scatter-plot with plot3D (i.e. plot.rgl = FALSE) for exporting as a png file
# png(file = "Sphere.png", width = 5000, height = 4000)
# A <- plot_word3D(word = "copula", R = 100, orbit = 300, copula = FALSE, plot.rgl = FALSE,
#                  plot.surface = "Sphere", cex.label = 5, size.lines = 2)
# dev.off()
```

Index

plot_word, 2
plot_word3D, 3