

cardinalR: Generating interesting high-dimensional data structures

by Jayani P. Gamage, Dianne Cook, Paul Harrison, Michael Lydeamore, and Thiyanga S. Talagala

Abstract A high-dimensional dataset is where each observation is described by many features, or dimensions. Such a dataset might contain various types of structures that have complex geometric properties, such as nonlinear manifolds, clusters, or sparse distributions. We can generate data containing a variety of structures using mathematical functions and statistical distributions. Sampling from a multivariate normal distribution will generate data in an elliptical shape. Using a trigonometric function we can generate a spiral. A torus function can create a donut shape. High-dimensional data structures are useful for testing, validating, and improving algorithms used in dimensionality reduction, clustering, machine learning, and visualization. Their controlled complexity allows researchers to understand challenges posed in data analysis and helps to develop robust analytical methods across diverse scientific fields like bioinformatics, machine learning, and forensic science. Functions to generate a large variety of structures in high dimensions are organized into the R package `cardinalR`, along with some already generated examples.

1 Introduction

Generating synthetic datasets with clearly defined geometric properties is essential for evaluating and benchmarking algorithms in various fields, such as machine learning, data mining, and computational biology. Researchers often need to generate data with specific dimensions, noise characteristics, and complex underlying structures to test the performance and robustness of their methods.

There are numerous packages available in R for generating synthetic data, each designed with unique characteristics and focus areas. For example, `geozoo` (Schloerke (2016)) offers a large collection of geometric objects, allowing users to create and analyze specific shapes, primarily in lower-dimensional spaces. Another useful package is `snedata` (Melville (2025)), which provides tools for generating simplified datasets useful for evaluating dimensionality reduction techniques like tSNE, often focusing on understanding and evaluating low-dimensional embeddings of complex data structures. Additionally, `mlbench` (Leisch and Dimitriadou (2024)) includes a collection of well-known benchmark datasets commonly associated with established classification or regression challenges. In the field of single-cell omics, `splatter` (Zappia et al. (2017)) is particularly simulate complex biological data, effectively capturing nuances such as batch effects and differential expression.

There is a valuable opportunity to improve the generation of high-dimensional data structures by integrating geometric principles with advanced noise control and customizable clustering. The `geozoo` package provides a strong foundation but could be enhanced to support high-dimensional extensions with controlled noise and user-defined parameters for clustering. Similarly, while `snedata` focuses on abstract datasets for dimensionality reduction, adding features for generating high-dimensional data from geometric layouts would enhance its usability. The `mlbench` package could also benefit from allowing users to create datasets with specific geometric structures and noise profiles. Additionally, although `splatter` specializes in biological data simulation, it could be expanded to offer a broader framework for generating diverse geometric structures across dimensions, enabling detailed control over noise and clustering. Addressing these areas could lead to more robust high-dimensional data generation tools.

To address this gap, this paper introduces the `cardinalR` R package. This package provides a collection of functions designed to generate customizable data structures in any number of dimensions, starting from basic geometric shapes. `cardinalR` offers important functionalities that extend beyond the capabilities of existing tools, allowing users to: (i) construct high-dimensional datasets based on geometric shapes, including the option to enhance dimensionality by adding controlled noise dimensions; (ii) introduce adjustable levels of background noise to these structures; and (iii) create complex clustered data arrangements by using fundamental geometric forms, while maintaining their positions, scales, orientations, and sample sizes in arbitrary dimensional spaces. By providing these integrated features, `cardinalR` aims to provide researchers to generate more explainable and challenging synthetic datasets focused to the specific needs of evaluating algorithms in high-dimensions. This bridges the gap between geometric foundations and the flexible generation of complex synthetic data.

The paper is organized as follows. In next section, introduces the implementation of `cardinalR` package on CRAN and GitHub, including demonstration of the package's key functions. We illustrate how a clustering data structure affect the Non-linear dimension reductions in **Example** section. Finally, we give a brief conclusion of the paper and discuss potential opportunities for use of our data

Table 1: cardinalR data sets

data	explanation
mobius_clust_data	
mobius_clust_tsne_param1	
mobius_clust_tsne_param2	
mobius_clust_tsne_param3	
mobius_clust_umap_param1	
mobius_clust_umap_param2	
mobius_clust_umap_param3	

collection.

2 Implementation

Installation

The package can be installed from CRAN using

```
install.packages("cardinalR")
```

and the development version can be installed from GitHub

```
devtools::install_github("JayaniLakshika/cardinalR")
```

Web site

More documentation of the package can be found at the web site <https://jayanilakshika.github.io/cardinalR/>.

Data sets

The cardinalR package comes with several data sets that load with the package. These are described in Table 1.

Functions

Main function

Add main function argument list to a table...

```
gen_multicluster(
  n = c(200, 300, 500), p = 4, k = 3,
  loc = matrix(c(
    0, 0, 0, 0,
    5, 9, 0, 0,
    3, 4, 10, 7
  ), nrow = 4, byrow = TRUE),
  dim_weights = dim4_weights,
  scale = c(3, 1, 2),
  shape = c("gaussian", "bluntedcorn", "unifcube"),
  rotation = rotations_4d,
  is_bkg = FALSE
)
```

Table 2: cardinalR branching data generation functions

Function	Explanation
gen_expbranches	
gen_linearbranches	
gen_curvybranches	
gen_orglinearbranches	
gen_orgcurvybranches	

Table 3: cardinalR cube data generation functions

Function	Explanation
gen_gridcube	
gen_unifcube	
gen_cubehole	

Branching**Cone****Cube****Gaussian****Linear****Mobius****Polynomial****Pyramid****S-curve****Sphere****Swiss Roll**

To generalize the Swiss roll structure to arbitrary dimensions, we introduce a function `generate_swiss_roll(n, p)`, which constructs a high-dimensional version of the classic 3D Swiss roll while preserving its core characteristics.

The function generates n points in a p -dimensional space, where the first two dimensions (X_1 , X_2) define the primary Swiss roll shape using a parametric equation:

$$X_1 = t \cos(t), \quad X_2 = t \sin(t), \quad \text{where } t \sim U(0, 3\pi)$$

The third dimension (X_3) introduces variation perpendicular to the roll, sampled uniformly from $[-1, 1]$. Additional dimensions (X_4 to X_p) extend the data structure by applying a **sinusoidal transformation** of the parameter t , ensuring continuity in higher-dimensional spaces:

$$X_i = \frac{\sin(it)}{i}, \quad \text{for } i \geq 4.$$

Table 4: cardinalR polynomial data generation functions

Function	Explanation
gen_quadratic	
gen_cubic	

Table 5: cardinalR pyramid data generation functions

Function	Explanation
gen_pyr	
gen_pyrrect	
gen_pyrtri	
gen_pyrstar	
gen_pyrholes	

Table 6: cardinalR S-curve data generation functions

Function	Explanation
gen_scurve	
gen_scurvehole	

This transformation ensures a gradual decay in variance across dimensions, mimicking real-world high-dimensional structures where later dimensions often capture subtler variations.

Trigonometric

Odd shapes

Add table of odd shapes..

Additional functions

3 Application

- 1) Generate a data structure with different shapes (two non-linear, one gaussian, one linear, one cone in 5-D simplex) and fit different NLDR (Then, demonstrate how this data can be used to assess the ability of different dimensionality reduction techniques (e.g., PCA, t-SNE, UMAP) to preserve the underlying structure in a lower-dimensional embedding.)
- 2) Add noise dimensions to the same data by linking two or three clusters, then discuss how single and complete linkage affect for the solutions (refer Di's book https://dicook.github.io/mulgar_book/8-hierarchical.html)

4 Discussion

The application of our high-dimensional data generation package to evaluate the interplay between dimensionality reduction, nuisance variables, and hierarchical clustering yielded several key insights. The ability to generate synthetic datasets with well-defined underlying structures, coupled with the controlled introduction of nuisance variables, provided a valuable platform for assessing the robustness of downstream unsupervised learning techniques.

Our findings demonstrated that the choice of dimensionality reduction method significantly impacted the ability of hierarchical clustering to recover the true underlying clusters. Methods that

Table 7: cardinalR sphere data generation functions

Function	Explanation
gen_circle	
gen_curvycycle	
gen_unifsphere	
gen_gridedsphere	
gen_clusteredspheres	

Table 8: cardinalR trigonometric data generation functions

Function	Explanation
gen_crescent	
gen_curvycylinder	
gen_sphericals spiral	
gen_helicals spiral	
gen_conics spiral	
gen_nonlinear	

Table 9: cardinalR additional functions

Function	Explanation
gen_noisedims	
gen_bkgnoise	
randomize_rows	
gen_nproduct	
gen_nsum	
gen_wavydims1	
gen_wavydims2	
gen_wavydims3	
gen_rotation	
gen_globalminmax	
scale_data	

effectively preserved the global structure of the data, as defined by our generation process, generally led to more accurate and interpretable hierarchical clustering results. However, the presence of nuisance variables often confounded the low-dimensional embeddings, making it more challenging for hierarchical clustering to separate truly distinct groups. This highlights a critical consideration in real-world data analysis, where unmeasured or latent factors can obscure the signal of interest.

The hierarchical clustering analysis itself, and the choice of linkage criteria, also played a crucial role. Different linkage methods revealed varying degrees of sensitivity to the distorted representations caused by the nuisance variables. For instance, methods that prioritize compact clusters might have been more susceptible to being misled by variance introduced by nuisance factors, while others focusing on the distance between clusters might have shown more resilience. The dendrograms generated by hierarchical clustering provided a visual means to explore the relationships between samples and the potential influence of nuisance, although determining the optimal number of clusters remained a challenge in the presence of these confounding factors.

This application underscores the utility of our data generation package as a powerful tool for controlled experimentation in unsupervised learning. By providing the ground truth cluster assignments and the ability to systematically manipulate data characteristics like dimensionality, geometric structure, and the presence of nuisance variables, researchers can gain a deeper understanding of the strengths and limitations of various DR and clustering algorithms. This controlled environment allows for a more objective evaluation than often possible with real-world datasets where the underlying structure is unknown.

One limitation of this particular application was the specific type and magnitude of the nuisance variables introduced. Future work could explore a wider range of nuisance types (e.g., batch effects, technical noise with specific distributions) and their varying degrees of influence on different DR and clustering methodologies. Furthermore, investigating strategies for mitigating the impact of nuisance variables, either during the DR step or within the clustering process itself, would be a valuable extension.

In conclusion, this example demonstrates the critical role of synthetic data generation in dissecting the complex interactions within unsupervised learning pipelines. Our package provides a flexible and controlled means to create such data, enabling researchers to systematically evaluate the performance and robustness of dimensionality reduction and clustering algorithms under well-defined conditions, including the presence of confounding factors. This capability contributes to a more rigorous and informed approach to high-dimensional data analysis.

5 Code

The code is available at <https://github.com/JayaniLakshika/cardinalR>, and source material for this paper is available at <https://github.com/JayaniLakshika/paper-cardinalR>.

6 Acknowledgements

This article is created using **knitr** (Xie, 2015) and **rmarkdown** (Xie et al., 2018) in R with the `rjtools::rjournal_article` template.

Bibliography

- F. Leisch and E. Dimitriadou. *mlbench: Machine Learning Benchmark Problems*, 2024. URL <https://CRAN.R-project.org/package=mlbench>. R package version 2.1-6. [p1]
- J. Melville. *snedata: SNE Simulation Dataset Functions*, 2025. URL <https://github.com/jlmelville/snedata>. R package version 0.0.0.9001, commit beebcf91c365bf5006be08fb614585b4659c05c5. [p1]
- B. Schloerke. *geozoo: Zoo of Geometric Objects*, 2016. URL <https://CRAN.R-project.org/package=geozoo>. R package version 0.5.1. [p1]
- Y. Xie. *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition, 2015. URL <https://yihui.name/knitr/>. ISBN 978-1498716963. [p6]
- Y. Xie, J. Allaire, and G. Golemund. *R Markdown: The Definitive Guide*. Chapman and Hall/CRC, Boca Raton, Florida, 2018. URL <https://bookdown.org/yihui/rmarkdown>. ISBN 978-1138359338. [p6]
- L. Zappia, B. Phipson, and A. Oshlack. Splatter: simulation of single-cell rna sequencing data. *Genome Biology*, 2017. doi: 10.1186/s13059-017-1305-0. URL <http://dx.doi.org/10.1186/s13059-017-1305-0>. [p1]

Jayani P. Gamage
Monash University
Department of Econometrics and Business Statistics, VIC 3800 Australia
<https://jayanilakshika.netlify.app/>
ORCID: 0000-0002-6265-6481
jayani.piyadigamage@monash.edu

Dianne Cook
Monash University
Department of Econometrics and Business Statistics, VIC 3800 Australia
<http://www.dicook.org/>
ORCID: 0000-0002-3813-7155
dicook@monash.edu

Paul Harrison
Monash University
MGBP, BDInstitute, VIC 3800 Australia
ORCID: 0000-0002-3980-268X
paul.harrison@monash.edu

Michael Lydeamore
Monash University
Department of Econometrics and Business Statistics, VIC 3800 Australia
ORCID: 0000-0001-6515-827X
michael.lydeamore@monash.edu

Thiyanga S. Talagala
University of Sri Jayewardenepura
Department of Statistics, Gangodawila, Nugegoda 10100 Sri Lanka
<https://thiyanga.netlify.app/>

ORCID: 0000-0002-0656-9789
ttalagala@sjp.ac.lk