Semi-automated Ti*k*Z directed acyclic graphs in R

Travis Stenborg

Abstract

Directed acyclic graphs (DAGs) are a key visualisation tool in graph theory. Semi-automated generation of Ti*k*Z code for rendering DAGs is introduced. Automatic Ti*k*Z generation via the `causalDisco` package of the R statistical programming language is proposed. Such easy, rapid DAG generation for LaTeX environments alleviates the need for tedious manual layout of DAG vertices and edges.

1 Directed acyclic graphs

Directed acyclic graphs (DAGs) are a type of mathematical graph structure consisting of *vertices* connected by *edges*. DAGs have two properties that distinguish them from general graphs. Firstly, DAGs have edges with an associated direction defining an order to the vertices (hence, *directed*). Secondly, the edges never define a path wherein the starting vertex of a path is also its ending vertex (hence, *acyclic*).

DAGs have applications in fields such as causal data science [4], computational optimisation [5] and even TeX paragraph aesthetics [7]. Ti*k*Z rendering allows fine tuning of graph presentation, and easy font matching with underlying LaTeX documents.

2 The `causalDisco` R package

The "causal discovery" R package, `causalDisco`, can autogenerate Ti*k*Z code to render DAGs from a concise vertex and edge specification. A version (0.9.1) is available from CRAN, but the more recent version (0.9.3) from GitHub addresses rendering bugs. Additionally, `causalDisco` has Bioconductor package dependencies. A combined download call in R is:

```
BiocManager::install(c("graph", "RBGL"))
github_repo <- "annennenne/causalDisco"
devtools::install_github(github_repo)
```

3 Automated Ti*k*Z from R

Example R code to render a DAG in Ti*k*Z is given below. A seven-vertex graph derived from coral reef ecology [1] was used here.

```
dag_matrix = matrix(
  c(0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0,
    1, 1, 0, 0, 0, 0, 0,
    1, 0, 1, 1, 0, 0, 0,
    0, 1, 1, 0, 1, 0, 0,
    0, 0, 0, 1, 0, 1, 0),
  nrow = 7, ncol = 7, byrow = TRUE)

# Specify matching matrix row and column names.
rownames(dag_matrix) <- c(
  "a_nd1", "a_nd2", "a_nd3",
  "b_nd4", "b_nd5", "b_nd6", "c_nd7")
colnames(dag_matrix) = rownames(dag_matrix)

# Create a temporal adjacency matrix.
model <- causalDisco::tamat(
  dag_matrix, c("a", "b", "c"))

# Render TikZ and copy to clipboard.
causalDisco::maketikz(model, xjit = 0,
  markperiods = FALSE, addAxis = FALSE,
  varLabels = list(
    a_nd1 = "Depth",
    a_nd2 = "\\footnotesize Structural\\\\
             \\footnotesize Complexity",
    a_nd3 = "\\footnotesize Human\\\\
             \\footnotesize Gravity",
    b_nd4 = "MPA",
    b_nd5 = "\\footnotesize Fishing\\\\
             \\footnotesize Pressure",
    b_nd6 = "\\footnotesize Reef Fish\\\\
             \\footnotesize Biomass",
    c_nd7 = "\\footnotesize Coral\\\\
             \\footnotesize Cover")
  )
```

By default, `causalDisco` generates `\small` vertex labels. To better balance the size of graph vertices with multiline vs. single line labels, judicious label adjustment via `\footnotesize` was made.

4 Finishing touches

There is no shortage of learning material for Ti*k*Z beginners [2, 6, 8, 10, 11, 12, 13, 14, 15, 16, 17]. Assuming Ti*k*Z basics are familiar, the `causalDisco` manual recommends the following Ti*k*Z preamble.

```
\usepackage{tikz}
\usetikzlibrary{arrows, arrows.meta,
  automata, backgrounds, shapes, snakes,
  petri}
\usepackage{pgfplots}
```

The example `causalDisco` code copies Ti*k*Z commands to the system clipboard. They should be pasted into a Ti*k*Z `\begin{tikzpicture}` and `\end{tikzpicture}` block. The compiled results generate *naked nodes*, i.e. nodes without any encapsulating boundary. Here however, additional Ti*k*Z `shape` calls, shown below, were manually added to nodes to encircle them (hence, *semi-automated*).

```
\node (1) at (0,1) [shape=circle,draw]
      {Depth};
```

The end result is given in Figure 1.

doi.org/10.47397/tb/45-1/tb139stenborg-dags

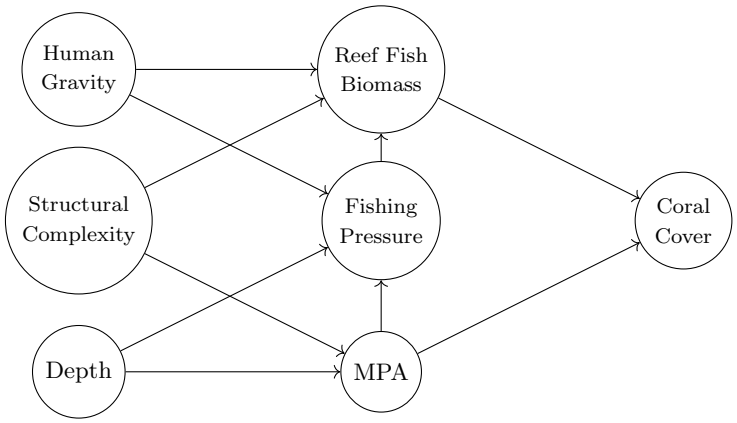

Figure 1: Directed acyclic graph visualising the causal structure of the influence of marine protected areas (MPAs [9]) on reef fish biomass. Adapted from an example in coral reef ecology [1]. *Human gravity* measures the human population near a reef, divided by the square of the time it takes to travel to that reef [3].

Acknowledgements

This work was supported by the Australian Research Council Training Centre in Data Analytics for Resources and Environments (project ICI9010031).

References

[1] S. Arif, M.A. MacNeil. Utilizing causal diagrams across quasi-experimental approaches. *Ecosphere*, 13(4):e4009, 2022.

[2] I. Borja. An introduction to automata design with TikZ's automata library. *TUGboat* 44(1):102–107, 2023. doi.org/10.47397/tb/44-1/tb136prado-automata

[3] J.E. Cinner, E. Maire, et al. Gravity of human impacts mediates coral reef conservation gains. *Proc. Natl. Acad. Sci. USA*, 115(27):E6116–E6125, 2018.

[4] G. Gao, B. Mishra, D. Ramazzotti. Causal data science for financial stress testing. *J. Comput. Sci.*, 26:294–304, 2018.

[5] J.L. Gross, J. Yellen, P. Zhang. *Handbook of Graph Theory, 2nd Edition*. Chapman and Hall/CRC, 2013.

[6] G. Grätzer. *More Math Into LaTeX*. Springer, Cham, 5th ed., 2016.

[7] Y. Haralambous. TeX as a path, a talk given at Donald Knuth's 80th birthday celebration symposium. *TUGboat* 39(1):8–15, 2018. tug.org/TUGboat/tb39-1/tb121haralambous-knuth80.pdf

[8] S. Kottwitz. *LaTeX graphics with TikZ*. Packt, Birmingham, 2023.

[9] D. Laffoley, J.M. Baxter, et al. Ch. 29: Marine protected areas. In *World Seas: An Environmental Evaluation*, pp. 549–569. Elsevier, second ed., 2019.

[10] C. Maggi. The DuckBoat: The Morse code of Ti*k*Z. *TUGboat* 39(1):21–26, 2018. tug.org/TUGboat/tb39-1/tb121duck-tikz.pdf

[11] C. Maggi. The DuckBoat: You do not need to be Neo to cope with a Ti*k*Z matrix. *TUGboat* 41(1):20–25, 2020. tug.org/TUGboat/tb41-1/tb127duck-matrix.pdf

[12] A. Mertz, W. Slough. Graphics with PGF and Ti*k*Z. *TUGboat* 28(1):91–99, 2007. tug.org/TUGboat/tb28-1/tb88mertz.pdf

[13] A. Mertz, W. Slough. A Ti*k*Z tutorial: Generating graphics in the spirit of TeX. *TUGboat* 30(2):214–226, 2009. tug.org/TUGboat/tb30-2/tb95mertz.pdf

[14] T. Stenborg. A TikZ rendering of the Arecibo message. *TUGboat* 44(3):375–377, 2023. doi.org/10.47397/tb/44-3/tb138stenborg-arecibo

[15] M.R.C. van Dongen. *LaTeX and Friends*. Springer, Berlin, 2012.

[16] Z. Walczak. Graphics in LaTeX using Ti*k*Z. *TUGboat* 29(1):176–179, 2008. tug.org/TUGboat/tb29-1/tb91walczak.pdf

[17] K. Wolcott. Three-dimensional graphics with PGF/Ti*k*Z. *TUGboat* 33(1):102–113, 2012. tug.org/TUGboat/tb33-1/tb103wolcott.pdf

◇ Travis Stenborg
Sydney, Australia
ORCID 0000-0002-2693-9628