series in their previous course, now find them interesting and valuable.

John Holmes

Credits

Photo of John Holmes is courtesy of John Holmes.

DONUT: Creation, Development, and Opportunities of a Database

Barbara Giunti, Jānis Lazovskis, and Bastian Rieck

1. Origin

DONUT[1] [GLR22] is a database of papers about practical, real-world uses of topological data analysis (TDA). Its original seed was planted in a group chat formed during the HIM Spring School on Applied and Computational Algebraic Topology in April 2017.

In 2019, Barbara Giunti, at the time a PhD student at the University of Pavia, asked the group chat whether anyone had heard of applications of topology in a specific area. Jānis Lazovskis, then also a PhD student, at the University of Illinois at Chicago, had been collecting such papers during the spring school and later events, and shared a list of about 10 papers demonstrating TDA applications. The format of an online spreadsheet soon proved too restrictive, and in 2020, they moved to Zotero [GLR20], an application specifically designed to handle bibliographic databases. The number of applications had increased by then to around 30, and Jānis and Barbara started to feel

Figure 1. Participants of the 2017 Spring School on Applied and Computational Algebraic Topology.

the need for smart planning: what if this project grows as we dream, with hundreds of entries and to be used by many? How can we make it searchable and versatile? They came up with a *tags* and *flavors* system compatible with the Zotero infrastructure and classified all the papers accordingly (more information about the system can be found in Section 3.2). In the meantime, they started advertising the database, and received immediate positive feedback from the community. Among the backers, they got the great help of Professor Mikael Vejdemo-Johansson, who provided more than one hundred papers from his personal database. Moreover, the year after, Professor Vejdemo-Johansson covered the cost of the annual Zotero subscription since the volume of papers had by then exceeded the threshold of free storage (the fee has since been footed by Barbara).

Figure 2. Participants of the 2022 workshop Topology of Data in Rome.

In 2022, the Zotero database had reached over 300 entries, all classified according to the tags and flavors system, and it began to show its limitations: one needed to be familiar with the app to successfully find the desired references. Luckily, in Salzburg during the biannual TDA Austrian Meeting and then at the workshop "Topology of Data in Rome," Barbara met Bastian Rieck, who fell in love with the project and revitalized it with his contribution: the web frontend search engine DONUT. This acronym stands for "Database of Original & Non-Theoretical Uses of Topology," and references one of the most basic shapes with nontrivial topology, the donut.

Barbara Giunti was a postdoctoral researcher at Graz University of Technology and now is an assistant professor at University at Albany. Her email address is bgiunti@albany.edu.

Jānis Lazovskis is a docent at the Riga Technical University and a faculty member at RTU Riga Business School. His email address is janis.lazovskis_1@rtu.lv.

Bastian Rieck is a principal investigator at Helmholtz Munich and a faculty member of the School of Computation, Information, and Technology at the Technical University of Munich. His email address is bastian.rieck@tum.de.

DOI: https://doi.org/10.1090/noti2797

[1] https://donut.topology.rocks

2. Motivation

The original goal was to have a tool to retrieve all applications of TDA in a specific domain, to provide an overview not only at the specific application level but also of all the areas of applications at a higher level. This tool is useful, for example, to promote the research field, showcasing its richness and power. Having such a tool is particularly handy in preparing introductions of papers and theses and, crucially, when writing grant applications, to reference relevant uses of a particular method. It also serves as a way to attract more researchers to the field; TDA being a highly intersectional and interdisciplinary field, it is open to new contributions from different domains. DONUT is also useful to create or extend projects, for example, by applying TDA in novel domains or overcoming limitations of previous approaches.

Another goal in creating this database was to organize existing knowledge, a burning necessity in an age of information overload. For this reason, the tags and flavors include not only the area of applications but also which mathematical tools are used, how the data are retrieved and pre-processed, and how novel the results are in the specific domain of application. Having the information in such a structured format not only helps the practitioners achieve their research goals, but also allows for literature and cross-sectional studies. In the absence of a structured bibliographic format for reporting such details, DONUT serves the important purpose of providing an ever-evolving, dynamic taxonomy of the field.

3. How the Cataloging Works

As of May 2023, the database contains over 430 entries.

3.1. Admissibility criteria.

To be included in DONUT, an entry must use a TDA technique to analyze data. We therefore exclude applications to other areas of mathematics or computer science, or employing mathematical (even topological) tools that are not part of the TDA toolbox.

The entries we index must be either published or available as a preprint on a preprints server (such as arXiv or bioRxiv). We prefer open-access (OA) publications and items with a DOI. Preprints that are later published are replaced with their published version. If the later publication is not open access, a link to the public preprint is kept. Conference submissions are allowed only if published in proceedings; conference submissions that only consist of an abstract are not included.

3.2. Tags & flavors.

There are three classes of tags (*area of applications*, *mathematical tools used*, and *input type*) and two flavor labels (**innovate** and **confirm**). Every indexed entry must have at least one tag for each class; this is a hard requirement to ensure the utility of DONUT.

Area of applications. This is the most difficult tag to add. We hope to harness feedback from the community to continuously refine rules on adding this tag. Ideally, this tag should involve subtags, to ensure optimal search results. For example, an entry about epilepsy should have as area-of-applications tag `medicine`, as the general field, `neurology` as the specification of it, and, finally, the most precise tag `epilepsy`. Because of how DONUT works (see Section 4), searching for "`epilepsy`" and not for "`tag:epilepsy`" will result in all entries that mention the word and thus in an imprecise search output.

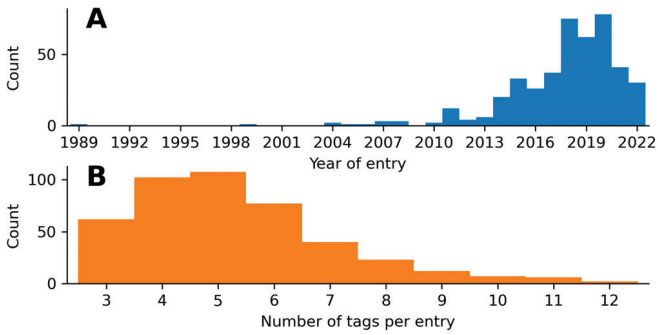

Figure 3. Histograms of (A) the year of publication and (B) the number of tags for each entry.

Mathematical tools used. This class is easiest to tag, as authors are usually clear about the technical description of the analysis process and state the used tools explicitly. We aim to tag all employed tools, not just the ones from TDA, to provide a faithful summary of the context in which TDA is applied.

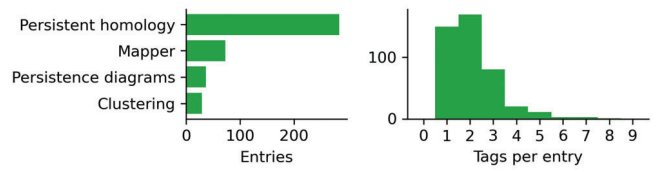

Figure 4. The most popular tags (left) for the type of tool used, and a histogram (right) of how many tags of this type each entry has.

Input type. The third class of tag is conceptually very simple and denotes the data type(s) used in the TDA pipeline, such as grayscale image, point cloud, time series, etc. However, in practice, we find that this tag is difficult to apply as input data usage is often not explicitly stated by the authors, or stated indirectly. Analysis should always be reproducible, and authors' failing to provide how the raw data were preprocessed to become suitable TDA input severely hinders reproducibility.

Flavors. Flavors are not mandatory, both because their classification is more delicate and because not all entries fall clearly in one or the other type. The label **confirm** states that the findings of the entry are aligned with findings of already-published methods. These entries perform the crucial job of reproducing results, and show that TDA

can be used as an alternative method. The label **innovate** encodes all entries whose results are novel to the specific area of application. A result can be novel, for example, if it comes from a larger dataset that was too big to be handled by other methods, or because TDA extracts more information from the same data, or also because TDA can analyze data that other methods could not. Since this tag is critical in promoting TDA, we decided to be strict about it: if a result is not unquestionably novel, the flavor is *not* added. As of May 2023, 12 entries are labeled **confirm** and 59 entries are labeled **innovate**.

4. Technical Details

DONUT is based on a database of bibliography entries that is maintained via Zotero. The advantages of using Zotero are

(i) it provides a simple way of searching for publications and indexing them, and
(ii) all bibliographical entries are stored as BibTeX entries.

This means that DONUT remains flexible and can be easily switched to another data source in the future, while at the same time not having to worry about issues with data entry. Thus, DONUT consists of three independent components:

1. An *importer* for one-way synchronization between Zotero and the database of entries.
2. A fulltext search engine for handling queries and maintaining the entries.
3. A web frontend for interaction with the fulltext search engine.

The importer is realized as a stand-alone program, making use of the Zotero API via `Pyzotero` [Hüg19]. The result of the parse process is a sequence of BibTeX entries. Each of these entries are then inserted into `Xapian`, an open-source full-text search engine. `Xapian` indexes bibliographic information of documents and makes them accessible via a well-defined API. Finally, a web interface based on `Flask`, a Python frontend for web development, interacts with the database, depicts the results, and renders all queries. We briefly comment on the choice behind the search engine and the frontend.

4.1. **A full-text search engine.** The benefit of a full-text search engine like `Xapian` is that the indexing process of structured document data is full of hidden complexities. For instance, are "high-dimensional" and "high dimensional" the same? How are simple spelling mistakes such as "simplical" instead of "simplicial" handled? How are transliterated spellings ("Pawel" instead of Paweł) or approximate spellings ("Pavel") treated? The frontend by Zotero ignores such questions and only permits simple queries that match a given string perfectly. `Xapian`, by contrast, is language-aware and can be set up to permit alternative spelling suggestions for queries. Since the utility of DONUT hinges on the quality of its results for a given query string, we opted to index as much information about a bibliographic entry as possible. As a result, DONUT is able to find documents more quickly than Zotero (with query times ranging in the lower millisecond range) and provide more depth to queries. Currently, only the content of BibTeX entries is used when searching, which includes the abstract, but not the full text.

4.2. **Frontend.** Users interact with databases typically through specialized query interfaces that are, ideally, as easy to use as Google. Using `Flask`, a Python-based web framework, we provide such an interface (in some sense, end users might perceive DONUT to *be* the web interface, but as outlined above, DONUT actually consists of multiple parts). The design choices behind the interface are first and foremost driven by speed and simplicity, following a minimalist design philosophy. The search interface will work well on big screens and small screens alike, and care has been taken to follow accessibility guidelines.

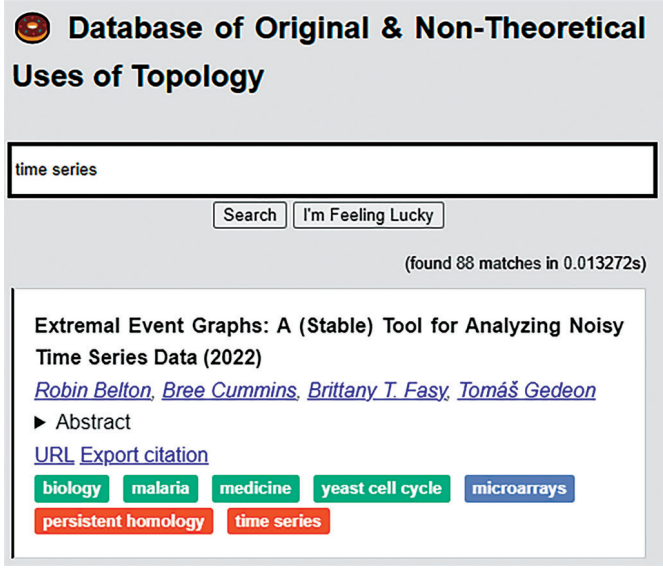

Figure 5. User interface with a search term (only top result shown).

DONUT does not track users by means of cookies or related technologies. General web server logs are stored in anonymized form, making it impossible to identify users. These logs are used for diagnosing problems and providing summary statistics about accesses to the database. Logs are stored encrypted and are automatically deleted on a rolling basis. DONUT is thus fully compliant with GDPR and goes well beyond the "best practices" of contemporary websites.

4.3. **Code.** We make the code for DONUT available using GitHub, using a BSD 3-Clause License.[2] This license essentially permits anyone to use the code and modify it,

[2] `https://github.com/Pseudomanifold/DONUT`

provided our original copyright notice remains intact. By making the code publicly available, any missing functionalities of the frontend or backend can be raised easily and addressed by the community or us. We also hope that DONUT will inspire similar initiatives, since the code is not specific to applications in topology and could easily accommodate other scientific domains as well.

4.4. **Example queries.** The frontend of DONUT can be used just like a regular search engine would be used. Any queries are searched for in any of the fields of the database. Thus, searching for `general` will find an entry called "The Euler Characteristic: A General Topological Descriptor for Complex Data," but also all documents that have `general` somewhere in their abstract, for instance. This mimics the default behavior of search engines, which do not care about the location of a search term within a website. To accommodate the needs of researchers, the DONUT frontend supports more refined queries and various operators: searching for `title:"euler"` will return all entries that have the word "Euler" somewhere in their title. Similarly, one can search for document tags and authors, via `tag:` and `author:`, respectively.

Automated normalization of queries. Concerning the aforementioned issues of different spellings, one prominent feature of the query interface is that it supports transliterated spellings of names. Hence, to stay with the original example, the search term `author:pawel` will show results that include both the spelling "Paweł," as well as the spelling "Pawel." Similar rules apply to names with umlauts and other special characters. This functionality is *unique* to DONUT and not provided by the Zotero query interface.

Spelling suggestions. Moreover, DONUT is capable (in contrast to Zotero) of suggesting other search terms to users based on similarity. DONUT will *never* change the search query on its own. It will, however, suggest alternative concepts or spellings. For instance, searching for `homotopy` brings up `homology` as a potentially related query. Searching for `homollogy`, on the other hand, will bring up no results, prompting DONUT to suggest "Did you mean '`homology`'?" We expect to further improve this functionality over time.

4.5. **Experimental features.** We also use DONUT as a platform to experiment with various ways of making application papers more accessible and queryable. For instance, we provide a "landscape visualization" that shows all indexed documents, following a natural landscape metaphor [FMM10]. This provides a way to interact with documents and potentially find similar papers.

Another ongoing improvement involves the indexing of the text of open-access publications. This is considerably more complicated than integrating overall bibliographic information (authors, abstracts, ...) since it requires being able to process PDF files. For entries whose text we can successfully process, we will incorporate the text in the search engine, meaning that keywords or phrases that appear only in the text of the entry will be made accessible to readers. To ensure compliance with copyright, we will only do this for open-access publications.

5. Future Opportunities

We want DONUT first and foremost to be a useful tool *by* the community *for* the community. As such, we believe that the most useful opportunities consist in expanding the taxonomy, that is, expanding the tagging system. Going from "more general" to "more specific" leads to a natural hierarchy of tags. For instance, an entry whose data tag is `graphs` could be assigned a more specific tag of the form `graphs:directed` if that is its context. Over time, we expect that such a hierarchy will become more refined, allowing both unspecific and highly specific queries. To aid users in their interactions with the hierarchy, we plan on implementing a "tree visualization" of tags. When viewing an individual entry, we will make excerpts of the hierarchy visible, making it possible for users to navigate within the tree.

We hope that DONUT continues to be a useful tool for our community. Everyone is warmly invited to contribute to DONUT in various ways. We are open to additional suggestions for inclusion, updates to the web interface, as well as suggestions for new functionality.

ACKNOWLEDGMENTS. The authors would like to thank the Hausdorff Research Institute for Mathematics for organizing excellent mathematical events, including the one at which the idea for this database was born, and Professor Mikael Vejdemo-Johansson and Professor Nina Otter for their contributions to the database. B.R. is grateful for discussions with Lukas Hahn, Maximilian Schmahl, and Daniel Spitz. B.G. was supported by the Austrian Science Fund (FWF) P 33765-N.

References

[FMM10] Sara Irina Fabrikant, Daniel R. Montello, and David M. Mark, *The natural landscape metaphor in information visualization: The role of commonsense geomorphology*, Journal of the American Society for Information Science and Technology **61** (2010), no. 2, 253–270.

[GLR20] Barbara Giunti, Jānis Lazovskis, and Bastian Rieck, *Zotero database of real-world applications of Toplogical Data Analysis*, 2020. `https://www.zotero.org/groups/tda-applications`.

[GLR22] Barbara Giunti, Jānis Lazovskis, and Bastian Rieck, *DONUT: Database of Original & Non-Theoretical Uses of Topology*, 2022. `https://donut.topology.rocks`.

[Hüg19] Stephan Hügel, *Pyzotero*, Zenodo, 2019. `https://doi.org/10.5281/zenodo.7057503`.


Barbara Giunti


Jānis Lazovskis


Bastian Rieck

Credits

Figure 1 is courtesy of ©Hausdorff Research Institute for Mathematics (HIM), Bonn.
Figure 2 is courtesy of Ryan Budney.
Figures 3–5 and photo of Jānis Lazovskis are courtesy of Jānis Lazovskis.
Photo of Barbara Giunti is courtesy of Barbara Giunti.
Photo of Bastian Rieck is courtesy of Andreas Heddergott.

Thinking About Failure in Data Analysis and Beyond

Roger D. Peng

Data science is a career that has expanded greatly over the past 10 years, with data science touching almost every aspect of our daily lives. Early career statisticians and mathematicians occasionally come to me and ask about the job of the data scientist, particularly in an academic setting. This can be a challenging question to answer as the job itself is continuously evolving both inside and outside of academia. My goal in this article is to touch on a concept that I think is common to the work of all data scientists, which is the question of what it means for a data analysis to fail.

As a statistician working in academia, I have a variety of jobs, including analyzing data and teaching about analyzing data. In the context of data analysis, I often think about what does it mean for a data analysis to fail? Usually, this is amongst the first questions students ask me because they want to do well in the class and outside of it. And yet answering this question is not necessarily straightforward. You can imagine how much the students love hearing that!

My general thinking about failure in any context is that the word "failure" is a bit overused and arguably has too many meanings. I like to split those meanings into three categories: funnies, anomalies, and (genuine) failures. What separates these three categories of outcomes are people's expectations and the consequences.

Funnies are unexpected outcomes that are interesting but don't necessarily change what you would do in the present or future. It's often useful to understand how these outcomes occur, but it's not necessarily an urgent matter. For example, if I'm analyzing a large dataset and one data point appears far different from what would consider a typical value, I might continue with the analysis anyway. But I want to eventually find out what caused that data point to be corrupted.

In the context of data analysis, anomalies are larger deviations from what we expect and can change how we do the analysis or how we collect future data. If in the previous example I had loaded the dataset and saw that half the observations were corrupted, I might pause and try to figure out what is going on. But that is only because my *expectation* was that all of the data points would be clean. Another, perhaps more experienced analyst, might know that this is what the data always look like.

A person's expectations are critical to defining what is anomalous or unexpected. In other words, one person's anomaly can be another person's expected outcome. When I was in graduate school, I submitted what I thought was a strong paper to a journal but the journal sent it back asking for a revision. I was so disappointed and frustrated at this outcome. But a senior professor later told me that in his entire career he had never had a paper accepted on the first submission. In this situation, the only thing that differed between me and this senior professor was our expectations.

Failures are the most severe category of outcome and distinguish themselves from anomalies in that people's expectations for what *should* happen are largely in agreement. If I am driving my car and notice that the brakes are no longer working, that is a failure. I would be hard-pressed to find a person that doesn't expect a car's brake to work all the time. The interesting thing about failures is that although they are easy to observe, their root causes may not be immediately obvious. I once had a student fail my class because he didn't hand in any work. Later, I discovered that he thought he had dropped the class early in the

Roger D. Peng is a professor of statistics and data science at the University of Texas, Austin. His email address is roger.peng@austin.utexas.edu.

DOI: https://doi.org/10.1090/noti2798