RESEARCH ARTICLE

shinyheatmap: Ultra fast low memory heatmap web interface for big data genomics

Bohdan B. Khomtchouk[1,2]*, **James R. Hennessy**[3], **Claes Wahlestedt**[1,2]

1 Center for Therapeutic Innovation, University of Miami Miller School of Medicine, 1501 NW 10th Ave., Miami, FL, 33136, United States of America, **2** Department of Psychiatry and Behavioral Sciences, University of Miami Miller School of Medicine, 1120 NW 14th St., Miami, FL, 33136, United States of America, **3** Department of Mathematics, University of Miami, 1365 Memorial Drive, Coral Gables, FL, 33146, United States of America

* b.khomtchouk@med.miami.edu

Abstract

Background

Transcriptomics, metabolomics, metagenomics, and other various next-generation sequencing (-omics) fields are known for their production of large datasets, especially across single-cell sequencing studies. Visualizing such big data has posed technical challenges in biology, both in terms of available computational resources as well as programming acumen. Since heatmaps are used to depict high-dimensional numerical data as a colored grid of cells, efficiency and speed have often proven to be critical considerations in the process of successfully converting data into graphics. For example, rendering interactive heatmaps from large input datasets (e.g., 100k+ rows) has been computationally infeasible on both desktop computers and web browsers. In addition to memory requirements, programming skills and knowledge have frequently been barriers-to-entry for creating highly customizable heatmaps.

Results

We propose shinyheatmap: an advanced user-friendly heatmap software suite capable of efficiently creating highly customizable static and interactive biological heatmaps in a web browser. shinyheatmap is a low memory footprint program, making it particularly well-suited for the interactive visualization of extremely large datasets that cannot typically be computed in-memory due to size restrictions. Also, shinyheatmap features a built-in high performance web plug-in, fastheatmap, for rapidly plotting interactive heatmaps of datasets as large as 10^5—10^7 rows within seconds, effectively shattering previous performance benchmarks of heatmap rendering speed.

Conclusions

shinyheatmap is hosted online as a freely available web server with an intuitive graphical user interface: http://shinyheatmap.com. The methods are implemented in R, and are available as part of the shinyheatmap project at: https://github.com/Bohdan-Khomtchouk/shinyheatmap. Users can access fastheatmap directly from within the shinyheatmap web

OPEN ACCESS

Citation: Khomtchouk BB, Hennessy JR, Wahlestedt C (2017) shinyheatmap: Ultra fast low memory heatmap web interface for big data genomics. PLoS ONE 12(5): e0176334. https://doi.org/10.1371/journal.pone.0176334

Editor: Chun-Hsi Huang, University of Connecticut, UNITED STATES

Received: December 17, 2016

Accepted: April 10, 2017

Published: May 11, 2017

Data Availability Statement: Availability of Data and Materials All source code has been made publicly available on Github at: https://github.com/Bohdan-Khomtchouk/shinyheatmap and https://github.com/Bohdan-Khomtchouk/fastheatmap.

Funding: Funding for this project was provided to BBK by the United States Department of Defense (DoD) through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program: this research was conducted with Government support under and awarded by DoD, Army Research Office (ARO), National Defense

Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript. Relevant work in CW's laboratory is currently funded by NIH grants DA035592 and AA023781.

Competing interests: The authors have declared that no competing interests exist.

Abbreviations: HPC, high performance computing; PCA, principal component analysis; UI, user interface; URL, Uniform Resource Locator.

interface, and all source code has been made publicly available on Github: https://github.com/Bohdan-Khomtchouk/fastheatmap.

Introduction

Heatmap software can be generally classified into two categories: static heatmap software [1–9] and interactive heatmap software [10–20]. Static heatmaps are pictorially frozen snapshots of genomic activity displayed as colored images generated from the underlying data. Interactive heatmaps are dynamic palettes that allow users to zoom in and out of the contents of a heatmap to investigate a specific region, cluster, or even single gene while, at the same time, being able to hover the mouse pointer over any specific row and column entry in order to glean information about an individual cell's contents (e.g., gene name, expression level, and column name). Interactive heatmaps are especially important for visualizing large gene expression datasets wherein individual gene labels eventually become unreadable due to text overlap, a common drawback seen in static heatmaps of large input data matrices. As such, interactive heatmaps are popular for examining the entire landscape of a large gene expression dataset while, at the same time, allowing users to zoom into specific sectors of the heatmap to visualize them in a magnified manner (i.e., at various resolution levels). Currently, there is a pressing need for modern libraries that are able to visually scale millions of data points at various resolutions [21]. In general, new software infrastructure that facilitates interactive navigation and smooth scaling at different resolution levels is necessary for on-the-fly calculations of both the frontend and backend algorithms in big data visualization software [22].

Even though static heatmaps are still the preferred type of publication figure in many studies, interactive heatmaps are becoming increasingly adopted by the scientific community to emphasize and visualize specific sectors of a dataset, where individual numerical values are rendered as user-specified colors. As a whole, the concept of interactivity is gradually shifting the heatmap visualization field into data analytics territory, for example, by synergizing interactive heatmap software with integrated statistical and genomic analysis suites such as PCA, differential expression, gene ontology, and network analysis [18, 23]. However, currently existing interactive heatmap software are limited by implicit restrictions on file input size, which functionally constrains their range of utility. For example, in Clustviz [23], which employs the pheatmap R package [9] for heatmap generation, input datasets larger than 1000 rows are discouraged [24] for performance reasons. Similarly, in MicroScope, the user is prompted to perform differential expression analysis on the input dataset first, thereby shrinking the number of rows rendered in the interactive heatmap to encompass only statistically significant genes [18]. In general, the standard way of thinking has been to avoid the production of big heatmaps due to a combination of various factors such as poor readability, as static heatmaps are not zoomable; computational infeasibility, since large interactive heatmaps require supercomputer-level memory resources to perform efficient, lag-free zooming and panning [25–31]; and unclear interpretation, since large heatmaps contain so much information that the standard recommended approach has been to preemptively subset the input data matrix into a smaller size [32].

Nevertheless, NGS-driven research studies often produce datasets on the order of 10^4 rows (e.g., transcriptome studies such as the HTA 2.0 array [33] that have up to 400,000 rows, each representing individual exons). Likewise, single-cell RNA-seq studies often produce datasets ranging from several thousand to several hundred thousand cells [34, 35], posing significant computational challenges to efficient data visualization. Currently, interactively visualizing such big data is not possible using existing state-of-the-art methodologies, despite existing efforts in this direction [36, 37]. Unlocking the computational ability to visualize interactive heatmaps on such unprecedented size scales would allow researchers to investigate high-dimensional numerical data as a colored grid of cells that is easily zoomable to any desired resolution, thereby aiding the exploratory data analysis process.

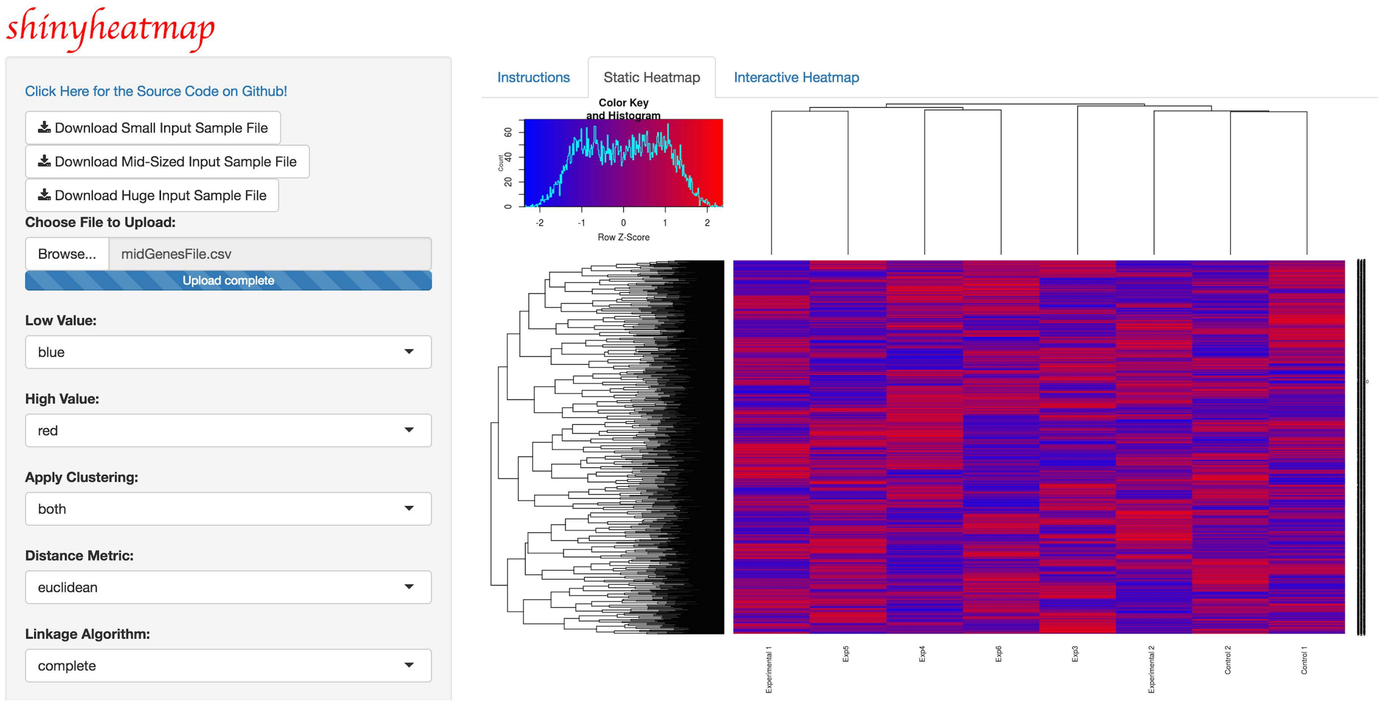
With the advent of increasingly sophisticated interactive heatmap software and the rise of big data coupled with a growing community interest to examine it interactively, there has arisen an unmet and pressing need to address the computational limitations that hinder the production of large, interactive heatmaps. Examining such heatmaps would be valuable for visualizing the landscape of both global gene expression patterns as well as individual genes. Motivated to address these objectives, we propose an ultra fast and low memory user-friendly heatmap software suite capable of efficiently creating highly customizable static and interactive heatmaps in a web browser.

Materials and methods

shinyheatmap is hosted online as an R Shiny web server application. shinyheatmap may also be run locally from within R Studio, as shown here: https://github.com/Bohdan-Khomtchouk/shinyheatmap. shinyheatmap leverages the cumulative utility of R's heatmaply [36], shiny [38], data.table [39], and gplots [40] libraries to create a cohesive web browser-based software experience requiring absolutely no programming experience from the user, or even the need to download R on a local computer. This kind of user-friendliness is geared towards the broader biological community, but will also appeal to the bioinformatics and computational biology communities. In contrast to most existing state-of-the-art heatmap software, shinyheatmap provides users with an extensive array of user-friendly hierarchical clustering methods, both in the form of multiple distance metrics as well as various linkage algorithms. This is especially useful for exploratory data analysis, particularly when the underlying data structure is unknown [41]. Since the choice of distance measure and linkage algorithm will directly influence the hierarchical clustering results, it is recommended to try different hierarchical clustering settings during analysis [41]. Agglomerative hierarchical clustering algorithms and their properties are described in detail at [42–46].

For the static heatmap generation, shinyheatmap employs the heatmap.2 function of the gplots library. For the interactive heatmap generation, shinyheatmap employs the heatmaply R package, which directly calls the plotly.js engine, in order to create fast, interactive heatmaps from large input datasets. The heatmaply R package is a descendent of the d3heatmap R package [47], which successfully creates advanced interactive heatmaps but is incapable of handling large inputs (e.g., 2000+ rows) due to memory considerations. As such, heatmaply constitutes a much-needed performance upgrade to d3heatmap, one that is made possible by the plotly R package [48], which itself relies on the sophisticated and complex plotly.js engine [49]. Therefore, it is the technical innovations of the plotly.js source code that make drawing extremely large heatmaps both a fast and efficient process. However, heatmaply also adds certain features not present in either the plotly.js engine nor the plotly R package, namely the ability to perform advanced hierarchical clustering and dendrogram-side zooming.

Despite these advantages, heatmaply is inadequate for plotting large datasets beyond a certain size limit, even with computationally expensive operations like hierarchical clustering disabled; for instance in certain cases, simple input matrices as small as 5000 × 5 may pose users with severe efficiency problems during heatmap rendering and zooming, even with no clustering present [37]. Due to this limitation, we developed a high performance web plug-in to



Choose File to Upload: midGenesFile.csv

Low Value: blue

High Value: red

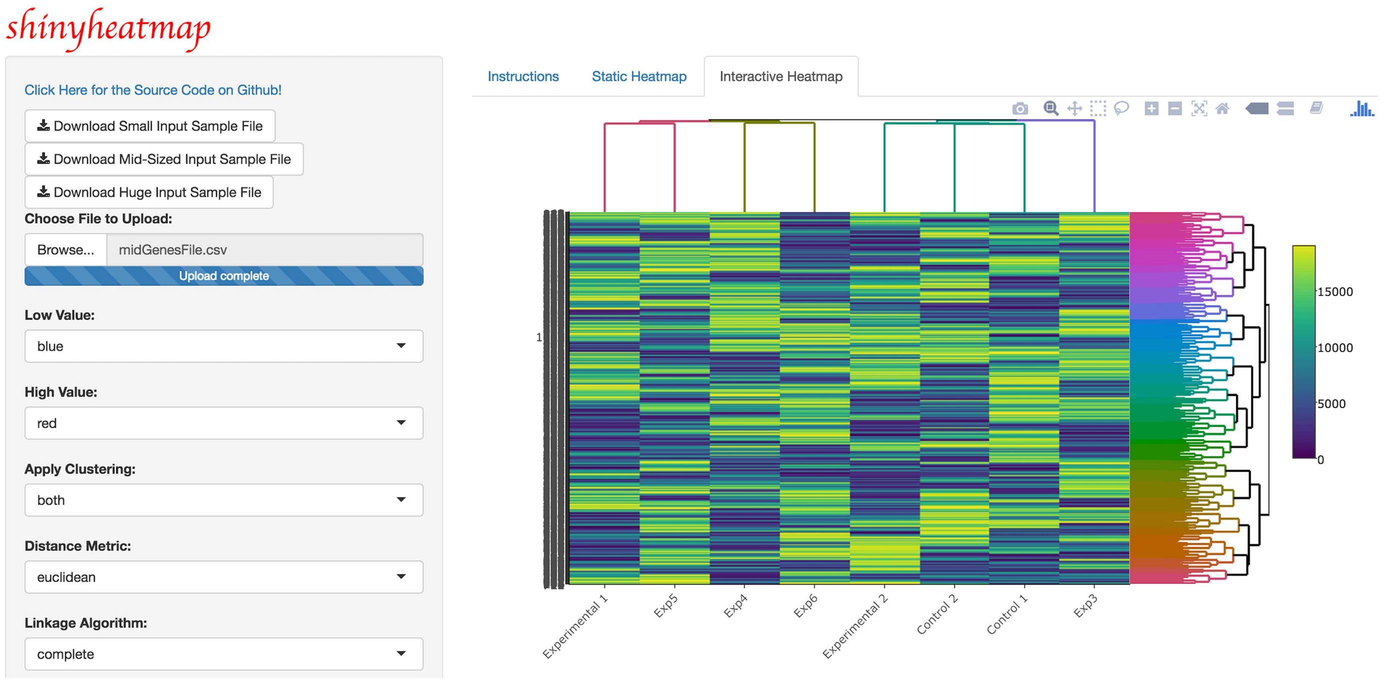
Apply Clustering: both

Distance Metric: euclidean

Linkage Algorithm: complete

Fig 1. shinyheatmap static heatmap. shinyheatmap UI showcasing the visualization of a static heatmap generated from a large input dataset. Parameters such as hierarchical clustering (including options for distance metrics and linkage algorithms), color schemes, scaling, color keys, trace, and font size can all be set by the user. Progress bars appear during the heatmap rendering process to alert the user if any technical issues may arise. Sample input files of various sizes are provided as part of the web application, whose source code can be viewed on Github.

https://doi.org/10.1371/journal.pone.0176334.g001



Choose File to Upload: midGenesFile.csv

Low Value: blue

High Value: red

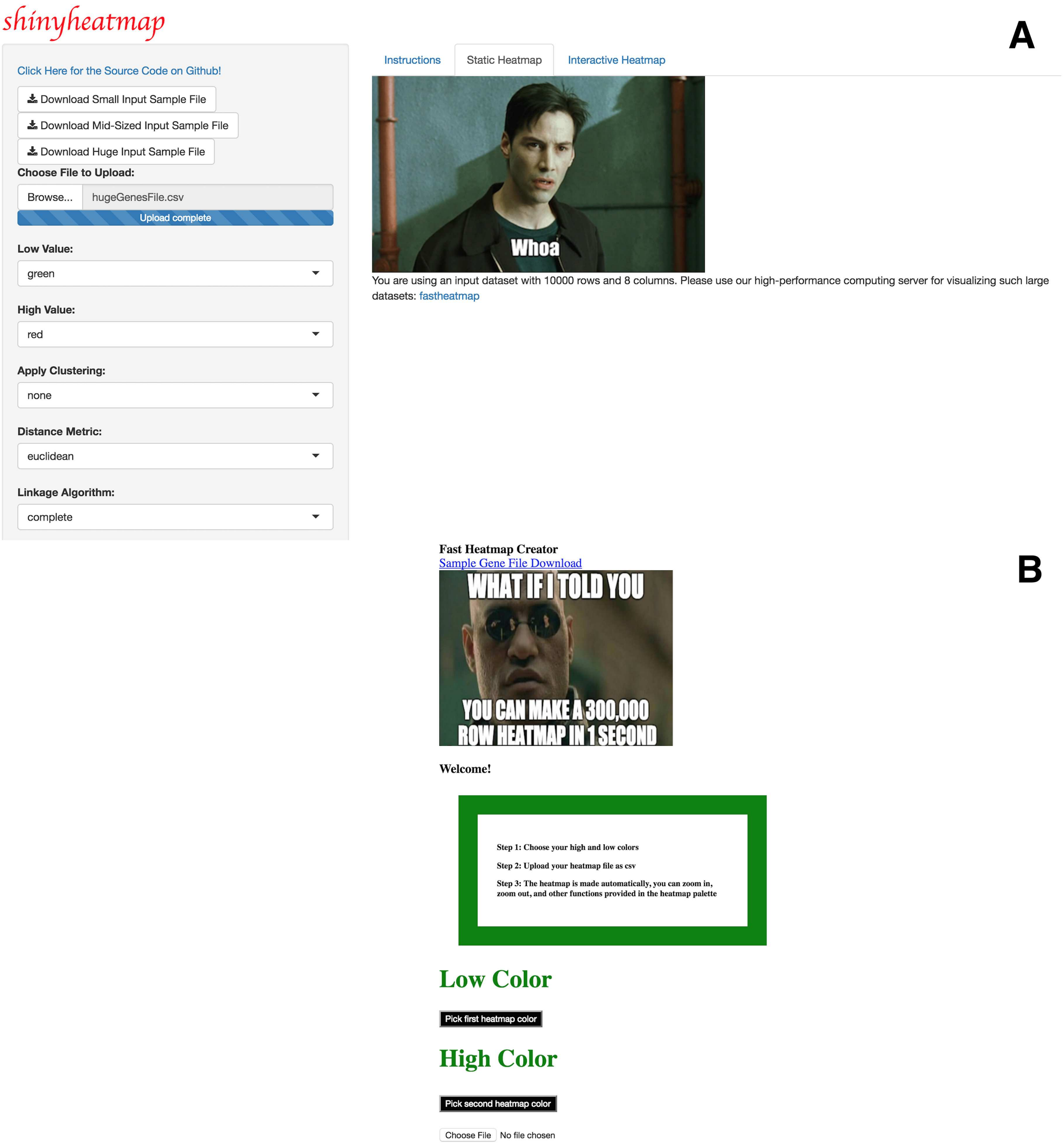
Apply Clustering: both

Distance Metric: euclidean

Linkage Algorithm: complete

Fig 2. shinyheatmap interactive heatmap. shinyheatmap UI showcasing the visualization of an interactive heatmap generated from a large input dataset. An embedded panel that appears top right on-hover provides extensive download, zoom, pan, lasso and box select, autoscale, reset, and other features for interacting with the heatmap.

https://doi.org/10.1371/journal.pone.0176334.g002


shinyheatmap
A
Click Here for the Source Code on Github!
Download Small Input Sample File
Download Mid-Sized Input Sample File
Download Huge Input Sample File
Instructions
Static Heatmap
Interactive Heatmap
Whoa
Browse...
Upload complete
B
WHAT IF I TOLD YOU
YOU CAN MAKE A 300,000 ROW HEATMAP IN 1 SECOND
Pick first heatmap color
Pick second heatmap color
Choose File

Choose File to Upload: hugeGenesFile.csv

Low Value: green

High Value: red

Apply Clustering: none

Distance Metric: euclidean

Linkage Algorithm: complete

You are using an input dataset with 10000 rows and 8 columns. Please use our high-performance computing server for visualizing such large datasets: fastheatmap

Fast Heatmap Creator
Sample Gene File Download

Welcome!

Step 1: Choose your high and low colors

Step 2: Upload your heatmap file as csv

Step 3: The heatmap is made automatically, you can zoom in, zoom out, and other functions provided in the heatmap palette

Low Color

High Color

No file chosen

Fig 3. fastheatmap & shinyheatmap are linked together. A) shinyheatmap contains an auto-detector that detects the size of a user's input matrix and, if the input matrix is too large, the user will be provided with a direct link to access shinyheatmap's high performance computing server: fastheatmap. B) fastheatmap UI upon clicking on the URL link shown in Panel A.

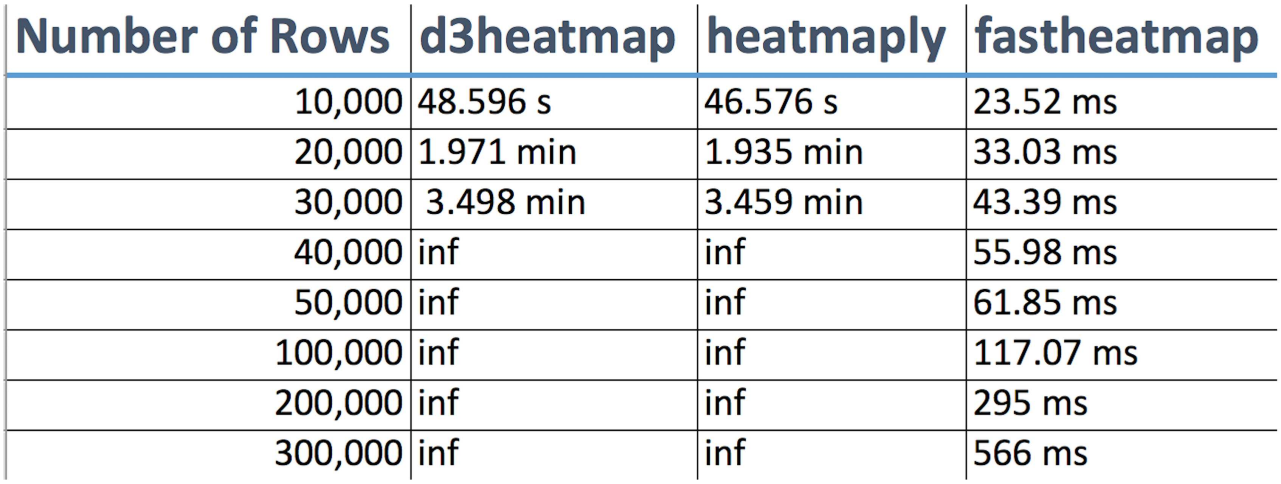
https://doi.org/10.1371/journal.pone.0176334.g003

shinyheatmap, called fastheatmap [50], which can rapidly plot interactive heatmaps of datasets as large as 10^5—10^7 rows within seconds directly in a web browser. Zooming in and out of such extremely large heatmaps is achievable in milliseconds, in contrast to d3heatmap or heatmaply, which takes minutes or even hours, if it is possible at all (due to memory limitations). This constitutes an unprecedented performance benchmark that dominantly positions shinyheatmap and its high performance computing server, fastheatmap, at the leading forefront of big data genomics heatmap visualization technology. In fact, to the best of our knowledge, the shinyheatmap/fastheatmap duo is the first big data software to appear on the biological heatmap visualization scene. All source code from the fastheatmap project is made publicly available at: https://github.com/Bohdan-Khomtchouk/fastheatmap.

Results

To use shinyheatmap, input data must be in the form of a matrix of integer values. The value in the *i*-th row and the *j*-th column of the matrix denotes how many reads (or fragments, for paired-end RNA-seq) have been unambiguously assigned to gene *i* in sample *j* [51]. Analogously, for other types of assays, the rows of the matrix might correspond e.g., to binding regions (with ChIP-seq), species of bacteria (with metagenomic datasets), or peptide sequences (with quantitative mass spectrometry). For detailed usage considerations, shinyheatmap provides a convenient Instructions tab panel upon login.

Upon uploading the input dataset, both static and interactive heatmaps are automatically created, each in their own respective tab panel. The user can then proceed to customize the static heatmap through a suite of available parameter settings located in the sidebar panel (Fig 1). For example, hierarchical clustering, color schemes, scaling, color keys, trace, and font size can all be set to the specifications of the user. In addition, a download button is provided for users to save publication quality heatmap figures. Likewise, the user can customize the interactive heatmap through its own respective hoverable toolbar panel located at the upper right corner of the heatmap (Fig 2). This toolbar provides extensive download, zoom, pan, lasso and box select, autoscale, reset, and hover features for interacting with the heatmap. Users with large input datasets will be directed by shinyheatmap to its fastheatmap plug-in by way of a user-friendly message that automatically recognizes the dimensions of the input data



Number of Rows	d3heatmap	heatmaply	fastheatmap
10,000	48.596 s	46.576 s	23.52 ms
20,000	1.971 min	1.935 min	33.03 ms
30,000	3.498 min	3.459 min	43.39 ms
40,000	inf	inf	55.98 ms
50,000	inf	inf	61.85 ms
100,000	inf	inf	117.07 ms
200,000	inf	inf	295 ms
300,000	inf	inf	566 ms

Fig 4. shinyheatmap performance benchmarks. shinyheatmap's HPC plug-in, fastheatmap, performs >100000 faster than other state-of-the-art interactive heatmap software. "Number of Rows" denotes the number of rows in the input file, "inf" (infinity) denotes a system crash due to memory overload, "s" denotes seconds, "min" denotes minutes, and "ms" denotes milliseconds.

https://doi.org/10.1371/journal.pone.0176334.g004

matrix (Fig 3). Performance benchmarks indicate (Fig 4) that fastheatmap significantly outperforms the latest state-of-the-art interactive heatmap software by several orders of magnitude. All benchmarks were tested on a 64-bit Windows 10 Pro desktop machine with 16.0 GB of RAM and an Intel(R) Core(TM) i7-5820K CPU at 3.30 GHz.

Conclusion

We provide access to a user-friendly web application designed to quickly and efficiently create static and interactive heatmaps within the R programming environment, without any prerequisite programming skills required of the user. Our software tool aims to enrich the genomic data exploration experience by providing a variety of customization options to investigate large input datasets.

Acknowledgments

BBK dedicates this work to the memory of his uncle, Taras Khomchuk. BBK wishes to acknowledge the financial support of the United States Department of Defense (DoD) through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program: this research was conducted with Government support under and awarded by DoD, Army Research Office (ARO), National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a. Relevant work in CW's laboratory is currently funded by NIH grants DA035592 and AA023781.

Author Contributions

Conceptualization: BBK.

Data curation: BBK.

Formal analysis: BBK.

Funding acquisition: BBK CW.

Investigation: BBK JRH.

Methodology: BBK.

Project administration: BBK CW.

Resources: CW.

Software: BBK JRH.

Supervision: BBK CW.

Validation: BBK JRH.

Visualization: BBK.

Writing – original draft: BBK.

Writing – review & editing: BBK.

References

1. Saeed AI, Sharov V, White J, Li J, Liang W, Bhagabati N, et al.: TM4: a free, open-source system for microarray data management and analysis. Biotechniques. 2003, 34(2): 374–378. PMID: 12613259

2. Reich M, Liefeld T, Gould J, Lerner J, Tamayo P, Mesirov JP: GenePattern 2.0. Nat Genet. 2006, 38(5): 500–501. https://doi.org/10.1038/ng0506-500 PMID: 16642009

3. Verhaak RGW, Sanders MA, Bijl MA, Delwel R, Horsman S, Moorhouse MJ, et al.: HeatMapper: powerful combined visualization of gene expression profile correlations, genotypes, phenotypes and sample characteristics. BMC Bioinformatics. 2006, 7:337. https://doi.org/10.1186/1471-2105-7-337 PMID: 16836741

4. Qlucore Omics Explorer: The D.I.Y Bioinformatics Software. http://www.qlucore.com.

5. Gould J: GENE-E software hosted at the Broad Institute. http://www.broadinstitute.org/cancer/software/GENE-E/.

6. Chu VT, Gottardo R, Raftery AE, Bumgarner RE, Yeung KY: MeV+R: using MeV as a graphical user interface for Bioconductor applications in microarray analysis. Genome Biology. 2008, 9: R118. https://doi.org/10.1186/gb-2008-9-7-r118 PMID: 18652698

7. Howe EA, Sinha R, Schlauch D, Quackenbush J: RNA-Seq analysis in MeV. Bioinformatics. 2011, 27(22): 3209–3210. https://doi.org/10.1093/bioinformatics/btr490 PMID: 21976420

8. Khomtchouk BB, Van Booven DJ, Wahlestedt C: HeatmapGenerator: high performance RNAseq and microarray visualization software suite to examine differential gene expression levels using an R and C++ hybrid computational pipeline. Source Code for Biology and Medicine. 2014, 9(1): 1–6.

9. Kolde R: *pheatmap: Pretty Heatmaps*. 2015. R package version 1.0.8. https://CRAN.R-project.org/package=pheatmap.

10. Saldanha AJ: Java Treeview—extensive visualization of microarray data. Bioinformatics. 2004, 20(17): 3246–3248. https://doi.org/10.1093/bioinformatics/bth349 PMID: 15180930

11. Caraux G, Pinloche S: Permutmatrix: A Graphical Environment to Arrange Gene Expression rofiles in Optimal Linear Order. Bioinformatics. 2005, 21: 1280–1281. https://doi.org/10.1093/bioinformatics/bti141 PMID: 15546938

12. Kibbey C, Calvet A: Molecular Property eXplorer: a novel approach to visualizing SAR using tree-maps and heatmaps. J Chem Inf Model. 2005, 45(2): 523–532. https://doi.org/10.1021/ci0496954 PMID: 15807518

13. Wu HM, Tien YJ, Chen CH: GAP: A Graphical Environment for Matrix Visualization and Cluster Analysis. Computational Statistics and Data Analysis. 2010, 54: 767–778.

14. Perez-Llamas C, Lopez-Bigas N: Gitools: analysis and visualisation of genomic data using interactive heat-maps. PLoS One. 2011, 6: e19541. https://doi.org/10.1371/journal.pone.0019541 PMID: 21602921

15. Škuta C, Bartůněk P, Svozil D: InCHlib—interactive cluster heatmap for web applications Journal of Cheminformatics. 2014, 6(44): 1–9.

16. Turkay C, Lex A, Streit M, Pfister H, Hauser H: Characterizing cancer subtypes using dual analysis in Caleydo StratomeX. IEEE Comput Graph Appl. 2014, 34(2): 38–47. https://doi.org/10.1109/MCG.2014.1 PMID: 24808198

17. Babicki S, Arndt D, Marcu A, Liang Y, Grant JR, Maciejewski A, et al.: Heatmapper: web-enabled heat mapping for all. Nucleic Acids Research 2016, pii: gkw419. [Epub ahead of print].

18. Khomtchouk BB, Hennessy JR, Wahlestedt C: MicroScope: ChIP-seq and RNA-seq software analysis suite for gene expression heatmaps. BMC Bioinformatics. 2016; 17(390). https://doi.org/10.1186/s12859-016-1260-x PMID: 27659774

19. Deu-Pons J, Schroeder MP, Lopez-Bigas N. jHeatmap: an interactive heatmap viewer for the web. Bioinformatics. 2014. https://doi.org/10.1093/bioinformatics/btu094 PMID: 24567544

20. Yachdav G, Hecht M, Pasmanik-Chor M, Yeheskel A, Rost B: HeatMapViewer: interactive display of 2D data in biology [v1; ref status: indexed, http://f1000r.es/2u6] F1000Research 2014, 3:48. https://doi.org/10.12688/f1000research.3-48.v1 PMID: 24860644

21. Pavlopoulos GA, Oulas A, Iacucci E, Sifrim A, Moreau Y, Schneider R, et al.: Unraveling genomic variation from next generation sequencing data. BioData Mining. 2013, 6:13. https://doi.org/10.1186/1756-0381-6-13 PMID: 23885890

22. Pavlopoulos GA, Malliarakis D, Papanikolaou N, Theodosiou T, Enright AJ, Iliopoulos I: Visualizing genome and systems biology: technologies, tools, implementation techniques and trends, past, present and future. GigaScience 2015, 4:38. https://doi.org/10.1186/s13742-015-0077-2 PMID: 26309733

23. Metsalu T, Vilo J: ClustVis: a web tool for visualizing clustering of multivariate data using Principal Component Analysis and heatmap. Nucleic Acids Research. 2015, 43(W1): W566–570. https://doi.org/10.1093/nar/gkv468 PMID: 25969447

24. Kolde R: pheatmap: Pretty Heatmaps. 2015. Page 4:ftp://cran.r-project.org/pub/R/web/packages/pheatmap/pheatmap.pdf.

25. SO, 2011. "How can I make a heatmap with a large matrix?" http://stackoverflow.com/questions/5667107/how-can-i-make-a-heatmap-with-a-large-matrix.

26. SO, 2013. "D3: How to show large dataset." http://stackoverflow.com/questions/18244995/d3-how-to-show-large-dataset.

27. SO, 2014. "How to draw heatmap with huge data." http://stackoverflow.com/questions/23297616/how-to-draw-heatmap-with-huge-data.

28. SO, 2014. "clustering very large dataset in R." http://stackoverflow.com/questions/21984940/clustering-very-large-dataset-in-r.

29. Google Groups, 2012. "Heat map with 500*300 nodes." https://groups.google.com/forum/m/#!topic/d3-js/wVWvwa-YkFE.

30. Mango Information Systems, 2013. "Pre-render d3.js charts at server side." https://mango-is.com/blog/engineering/pre-render-d3-js-charts-at-server-side/.

31. vida.io, 2014. "BigQuery Big Data Visualization With D3.js." http://blog.vida.io/2014/07/06/bigquery-big-data-visualization-with-d3-dot-js/.

32. Biostars 2014. "How to plot the heatmap of gene expression for very large data set?" https://www.biostars.org/p/104976/

33. Sood S, Szkop KJ, Nakhuda A, Gallagher IJ, Murie C, Brogan RJ, et al.: iGEMS: an integrated model for identification of alternative exon usage events Nucleic Acids Research. 2016, 44(11): e109. https://doi.org/10.1093/nar/gkw263 PMID: 27095197

34. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al.: Massively parallel digital transcriptional profiling of single cells. Nature Communications. 2017, 8:14049. https://doi.org/10.1038/ncomms14049 PMID: 28091601

35. 10x Genomics Inc., 2017. "Single cell datasets." https://support.10xgenomics.com/single-cell/datasets.

36. Galili T: *heatmaply: Interactive Heat Maps Using'plotly'*. 2016. R package version 0.6.0. https://CRAN.R-project.org/package=heatmaply.

37. Galili T: *heatmaply: Interactive Heat Maps Using'plotly'*. 2016. R package version 0.6.0. https://github.com/talgalili/heatmaply/issues/20.

38. Chang W, Cheng J, Allaire JJ, Xie Y, McPherson J, RStudio, et al.: *shiny: Web Application Framework for R*. 2015. R package version 0.12.2.

39. Dowle M, Srinivasan A, Short T, Lianoglou S, Saporta R, Antonyan E: *data.table: Extension of Data.frame*. 2015. R package version 1.9.6.

40. Warnes GR, Bolker B, Bonebakker L, Gentleman R, Huber W, Liaw A, et al.: *gplots: Various R Programming Tools for Plotting Data*. 2016. R package version 3.0.1. https://CRAN.R-project.org/package=gplots.

41. Sakai R, Winand R, Verbeiren T, Moere AV, Aerts J: dendsort: modular leaf ordering methods for dendrogram representations in R. [version 1; referees: 2 approved] F1000 Research. 2014, 3(177): 3246–3248.

42. R Core Team: *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. 2016. https://rweb.stat.umn.edu/R/library/stats/html/hclust.html.

43. R Core Team: *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. 2016. https://rweb.stat.umn.edu/R/library/stats/html/hclust.html.

44. Quackenbush J: Computational analysis of microarray data. Nature Reviews Genetics. 2001, 2(6): 418–27. https://doi.org/10.1038/35076576 PMID: 11389458

45. Tan P, Kumar V, Steinbach M: *Introduction to data mining*. Boston: Pearson Addison Wesley, 1st ed edition. 2005.

46. Hastie T, Tibshirani R, Friedman J: *The Elements of Statistical Learning*. Springer Series Statistics. 2009.

47. Cheng J, Galili T, RStudio Inc, Bostock M, Palmer J: *d3heatmap: Interactive Heat Maps Using 'htmlwidgets' and 'D3.js'*. 2015. R package version 0.6.1.

48. Sievert C, Parmer C, Hocking T, Chamberlain S, Ram K, Corvellec M, et al.: *plotly: Create Interactive Web Graphics via'plotly.js'*. 2016. R package version 3.6.0. https://CRAN.R-project.org/package=plotly.

49. Plotly Technologies Inc.: *Collaborative data science*. Plotly Technologies Inc. Montreal, QC. 2015, https://plot.ly.

50. Khomtchouk BB: *fastheatmap: high performance interactive heatmap software*. 2016-2017. https://github.com/Bohdan-Khomtchouk/fastheatmap.

51. Love M, Anders S, Kim V, Huber W: *RNA-seq workflow: gene-level exploratory analysis and differential expression*. 2016, http://www.bioconductor.org/help/workflows/rnaseqGene/.