



SOFTWARE TOOL ARTICLE

iSEE: Interactive SummarizedExperiment Explorer [version 1; referees: 3 approved]

Kevin Rue-Albrecht ^{1*}, Federico Marini ^{2,3*}, Charlotte Sonesson^{4,5*}, Aaron T.L. Lun^{6*}

¹Kennedy Institute of Rheumatology, University of Oxford, Oxford, OX3 7FY, UK

²Center for Thrombosis and Hemostasis (CTH), University Medical Center of the Johannes Gutenberg University Mainz, Mainz, Germany

³Institute for Medical Biostatistics, Epidemiology and Informatics (IMBEI), University Medical Center of the Johannes Gutenberg University Mainz, Mainz, Germany

⁴SIB Swiss Institute of Bioinformatics, University of Zurich, Zurich, 8057, Switzerland

⁵Institute of Molecular Life Sciences, University of Zurich, Zurich, 8057, Switzerland

⁶Cancer Research UK Cambridge Institute, University of Cambridge, Cambridge, CB2 0RE, UK

* Equal contributors

v1 First published: 14 Jun 2018, 7:741 (doi: [10.12688/f1000research.14966.1](https://doi.org/10.12688/f1000research.14966.1))
 Latest published: 14 Jun 2018, 7:741 (doi: [10.12688/f1000research.14966.1](https://doi.org/10.12688/f1000research.14966.1))

Abstract

Data exploration is critical to the comprehension of large biological data sets generated by high-throughput assays such as sequencing. However, most existing tools for interactive visualisation are limited to specific assays or analyses. Here, we present the iSEE (Interactive SummarizedExperiment Explorer) software package, which provides a general visual interface for exploring data in a SummarizedExperiment object. iSEE is directly compatible with many existing R/Bioconductor packages for analysing high-throughput biological data, and provides useful features such as simultaneous examination of (meta)data and analysis results, dynamic linking between plots and code tracking for reproducibility. We demonstrate the utility and flexibility of iSEE by applying it to explore a range of real transcriptomics and proteomics data sets.

Keywords

visualization, interactive, R, Bioconductor, genomics, transcriptomics, proteomics, shiny



This article is included in the **Bioconductor** gateway.

Open Peer Review

Referee Status:

	Invited Referees		
	1	2	3
version 1			
published 14 Jun 2018	report	report	report

- James W. MacDonald** , University of Washington, USA
- Lorena Pantano** , Harvard T. H. Chan School of Public Health, USA
- Alejandro Reyes** , Dana-Farber Cancer Institute, USA

Discuss this article

Comments (0)

Corresponding author: Aaron T.L. Lun (aaron.lun@cruk.cam.ac.uk)

Author roles: **Rue-Albrecht K:** Conceptualization, Software, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Marini F:** Conceptualization, Methodology, Software, Writing – Original Draft Preparation, Writing – Review & Editing; **Soneson C:** Conceptualization, Software, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Lun ATL:** Conceptualization, Methodology, Software, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

How to cite this article: Rue-Albrecht K, Marini F, Soneson C and Lun ATL. **iSEE: Interactive SummarizedExperiment Explorer [version 1; referees: 3 approved]** *F1000Research* 2018, 7:741 (doi: [10.12688/f1000research.14966.1](https://doi.org/10.12688/f1000research.14966.1))

Copyright: © 2018 Rue-Albrecht K *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution Licence](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Grant information: ATLL was supported by core funding from Cancer Research UK [award no. 17197 to JM]. The work of FM is supported by the German Federal Ministry of Education and Research (BMBF 01EO1003).

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

First published: 14 Jun 2018, 7:741 (doi: [10.12688/f1000research.14966.1](https://doi.org/10.12688/f1000research.14966.1))

Introduction

Interactive data exploration is critical to the analysis and comprehension of data generated by high-throughput biological assays, such as those commonly used in genomics. Exploration drives the formation of novel data-driven hypotheses prior to a more rigorous statistical analysis, and enables diagnosis of potential problems such as batch effects and low-quality samples. To this end, visualisation of the data using an intuitive and interactive interface is crucial for enabling researchers to examine the data from different perspectives across samples (e.g., experimental replicates, patients, single cells) and features (e.g., genes, transcripts, proteins, genomic regions).

Most existing tools for interactive visualisation of biological data are designed for specific assays and analyses, e.g., **pRoloc** for proteomics (Gatto *et al.*, 2014), **shinyMethyl** for methylation (Fortin *et al.*, 2014), **HTSvis** for high-throughput screens (Scheeder *et al.*, 2017). Opportunities for customisation are generally limited, making it difficult to re-use the same visualisation software for new technologies or experimental designs where different aspects of the data are of interest. Moreover, standalone tools such as the **Loupe Cell Browser** from 10x Genomics (Zheng *et al.*, 2017) do not easily integrate into established analysis pipelines such as those based on the R statistical programming language (R Development Core Team, 2008). This complicates any coordinated use of these tools with a reproducible, transparent, and statistically rigorous analysis.

Here, we present the **iSEE** software package for interactive data exploration. **iSEE** is implemented in R using the **Shiny framework** (Chang *et al.*, 2017) and exploits data structures from the open-source **Bioconductor project** (Gentleman *et al.*, 2004), specifically the **SummarizedExperiment** class. **iSEE** allows users to simultaneously visualise multiple aspects of a given data set, including experimental data, metadata and analysis results. Dynamic linking and point selection facilitate the flexible exploration of interactions between different data aspects. Additional functionalities include code tracking, intelligent downsampling of large data sets, custom colour scale specification and tour construction. We demonstrate the capabilities of **iSEE** by applying it to a diverse range of real data sets.

Operation

The **iSEE** software package requires R version 3.5.0 or higher, along with packages from Bioconductor version 3.7 or higher. The interface is initialised with a single call to the **iSEE()** function, accepting a **SummarizedExperiment** object (Huber *et al.*, 2015) as input. Any analysis workflow that generates a **SummarizedExperiment** object is supported.

Motivation for using the SummarizedExperiment class

Each instance of the **SummarizedExperiment** class stores one or more matrices of experimental observations as “assays”, where rows and columns represent genomic features and biological samples, respectively. For instance, individual assays may represent gene expression matrices, either in the form of raw counts or normalised values. In addition, per-feature or per-sample variables are stored in the “rowData” and “colData” slots,

respectively; these may include experimental metadata as well as analysis results.

The flexibility of the **SummarizedExperiment** class is the driving factor behind its broad deployment throughout the Bioconductor ecosystem. **SummarizedExperiment** objects are currently used in analysis pipelines for RNA sequencing (Love *et al.*, 2014), methylation (Aryee *et al.*, 2014) and Hi-C data (Lun *et al.*, 2016), amongst others. Package developers can also easily use the base **SummarizedExperiment** class to derive new bespoke classes for particular applications, such as the **SingleCellExperiment** class for single-cell ‘omics data. By accepting **SummarizedExperiment** objects as input, **iSEE** immediately offers interactive visualisation for a variety of data modalities. This complements the state-of-the-art analysis workflows and methodologies already available in R/Bioconductor packages.

Interface implementation

Using a multi-panel layout

All data aspects stored in a **SummarizedExperiment** can be simultaneously examined in the multi-panel layout of the **iSEE** interface (Figure 1A). The interface layout is built using the **shinydashboard** package (Chang & Borges Ribeiro, 2018), with colour-coded panels to visualise each data aspect. Individual panel types include:

- Column data plots, for visualising sample metadata stored in the **colData** slot of the **SummarizedExperiment** object.
- Feature assay plots, for visualising experimental observations for a particular feature (e.g. gene) across samples from any assay in the **SummarizedExperiment** object.
- Row statistics tables, to present the contents of the **rowData** slot of the **SummarizedExperiment** object.
- Row data plots, for visualising feature metadata stored in the **rowData** slot of the **SummarizedExperiment** object.
- Heatmaps, to visualise assay data for multiple features where samples are ordered by one or more **colData** fields.
- Reduced dimension plots, which display any two dimensions from pre-computed dimensionality reduction results (e.g., from PCA or *t*-SNE). These results are taken from the **reducedDim** slot if this is available in the object supplied to **iSEE**.

Each sample is represented as a point in column data, feature assay and reduced dimension plots. Similarly, each feature is represented by a point in row data plots. For these panel types, a scatter plot is automatically produced if the selected variables on the x- and y-axes are both continuous. If exactly one variable is categorical, points are grouped by the categorical levels and a (vertical or horizontal) violin plot is produced with points scattered within each violin. If both variables are categorical, a “rectangle plot” is produced where each combination of categorical levels is represented by a rectangle with area proportional to the frequency of that combination. Points are scattered randomly within each rectangle. For ease of interpretation, the rectangle

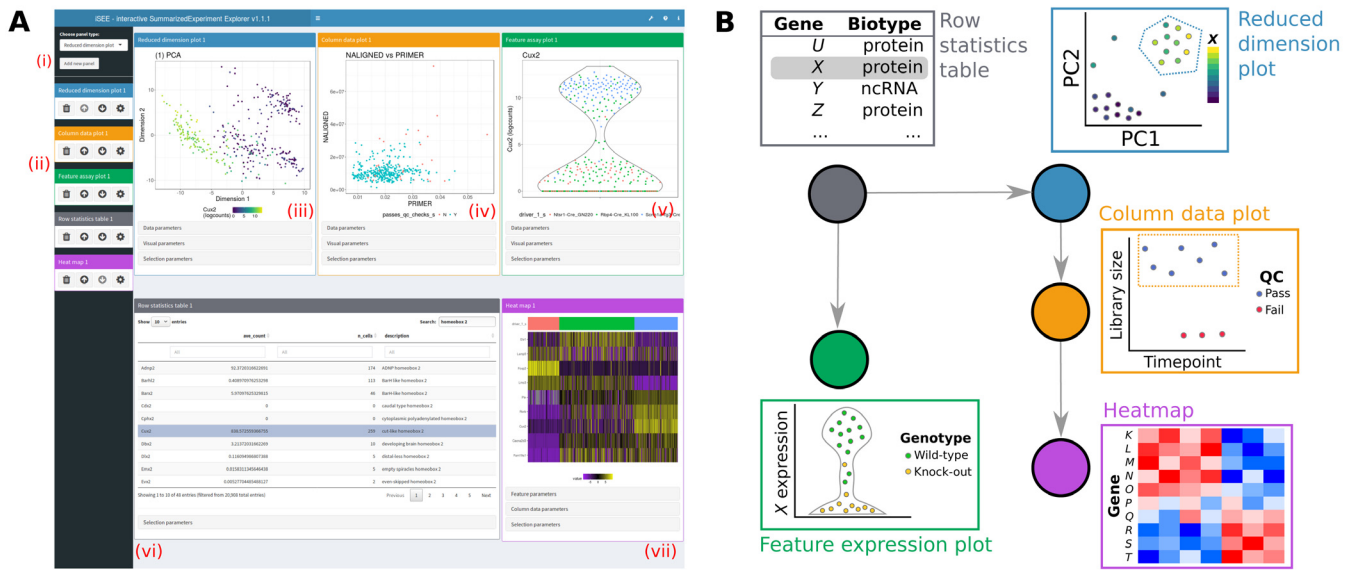


Figure 1. iSEE uses a customisable multi-panel layout (**A**) that simultaneously displays one or more panels of various types, where each panel type visualises a different aspect of the data. New panels of any type can be added (i), and all panels can be removed, reordered or resized (ii). Panel types are available to visualise sample-based reduced dimensionality embeddings (iii), sample-level metadata (iv), and experimental observations across samples for each feature (v). Other panel types include row statistics tables (vi), to facilitate searching across features and their metadata; heatmaps (vii), to visualise experimental observations for multiple features; and feature-level metadata plots. Panels of each type are colour-coded for ease of interpretation. (**B**) Information can be transmitted between panels according to a user-specified scheme. Here, the selection of feature *X* in the row statistics table determines the y-axis of the feature assay plot, and colours the samples in the reduced dimension plot by the expression of *X*. Selection of points in the reduced dimension plot (dotted blue line) also determines the samples that are shown in the column data (i.e., sample metadata) plot; further selection of points in the column data plot determines the samples that are shown in the heatmap.

plot collapses to a mirrored bar plot when one of the categorical variables only has one level.

Custom panel colouring

Sample-based points can be coloured according to the values of any sample-level metadata field in the colData slot or by the assay values of a selected feature. Similarly, feature-based points can be coloured according to any feature-level metadata field in the rowData slot. Heatmaps are coloured according to the expression values of the selected features in the chosen assay, with additional colour annotation for each of the colData fields used to order the samples. In all cases, the variable to use for colouring can be dynamically selected for each plot. This enables users to easily examine relationships between different variables in a single plot.

By default, colour maps for categorical and continuous variables are taken from the *ggplot2* (Wickham, 2009) and *viridis* packages (Garnier, 2018), respectively. However, iSEE also implements the ExperimentColorMap class, which allows users to specify arbitrary colour maps for particular variables. Each colour map is a function that returns a vector of distinct colours of a specified length, and will be called whenever the associated variable is used for point colouring in a particular panel. The returned colours will be mapped to factor levels for categorical variables, or used in colour interpolation for continuous variables. For categorical variables, the function

may also return a constant vector of named colours corresponding to the levels of a known factor. Colour maps can be specified for individual variables; for all assays, all column data variables, or all row data variables (with different functions for continuous or categorical variables); or for all categorical or continuous variables. This provides a convenient yet flexible mechanism for customisation of colouring schemes within the interface.

Dynamic linking between panels

A key feature of iSEE is the ability to dynamically transmit information between panels (Figure 1B). Users can define and reorganise arbitrary links between “transmitting” and “receiving” panels, whereby selections in transmitting panels control the inclusion and appearance of the corresponding data points in receiving panels. This feature facilitates exploration of the relationships between different aspects of the data. For example, users can easily determine co-expression patterns of genes in a particular region of a reduced dimensionality embedding – this is achieved by selecting points in a reduced dimension plot (using the standard rectangular brush or a lasso selection) and transmitting that selection to any number of feature assay plots.

This linking paradigm extends to multiple panels, whereby a panel can transmit to multiple receivers, and a receiving panel can transmit its own selection to another plot. Chains of linked plots allow users to mimic the arbitrarily complex gating strategies often found in analyses of flow cytometry data

(Finak *et al.*, 2014). With iSEE, this concept is extended to any assay data, feature-level or sample-level metadata present in a SummarizedExperiment object, providing a powerful framework for interrogating multiple interactions between data aspects. Row statistics tables can also transmit to various plot types, by selecting a table row to control the colouring of sample-based points; or by defining a subset of features to visualise in a heatmap. Furthermore, row data plots can transmit to row statistics tables, whereby selection of points in the former will subset the latter.

Code tracking and reproducibility

iSEE automatically memorises the exact R code that was used to generate every plot, extending previous work by Marini & Binder (2016). This code is fully accessible to users at any time during the run-time of the interface. By integrating the code reported by iSEE into their own scripts, users can easily reproduce the results of any exploratory analysis. Similarly, the code required to reproduce the current state of the interface can also be reported. This can be used in startup scripts to launch an iSEE instance in any preferred layout, including the panel organisation, variable selection, colouring schemes, links between panels and even individual brushes and lasso selections.

Additional functionalities

Row statistics tables can be augmented with dynamic annotation based on the selected row, linking to online resources such as Ensembl (Zerbino *et al.*, 2018) or Entrez (Coordinators, 2017). For large data sets, points can be downsampled in a density-dependent manner to accelerate rendering of the plots, improving the responsiveness of the interface without compromising the fidelity of the visualisation. Users can also include a bespoke step-by-step “tour” of their data set via the `rintrojs` package (Ganz, 2016), guiding the audience through an examination of the salient features in the data.

Use cases

Plate-based single-cell RNA sequencing

To demonstrate iSEE’s functionality, we used it to explore a plate-based single-cell RNA sequencing (scRNA-seq) data set involving 379 cells from the mouse visual cortex (Tasic *et al.*, 2016). This demonstration guides the user through the main features of the iSEE interface including the multi-panel layout, colouring and dynamic linking.

An interactive tour of this use case can be viewed [here](#).

Droplet-based single-cell RNA sequencing

We applied iSEE to a larger scRNA-seq data set involving 4,000 peripheral blood mononuclear cells (PBMCs), generated by 10x Genomics (Zheng *et al.*, 2017). This demonstration explores the differences between different methods for distinguishing cells from empty droplets in droplet-based scRNA-seq protocols (Lun *et al.*, 2018).

An interactive tour of this use case can be viewed [here](#).

Bulk RNA sequencing from TCGA

We applied iSEE to bulk RNA sequencing data from The Cancer Genome Atlas (TCGA) project, using a subset of expression profiles involving 7,706 tumor samples (Rahman *et al.*, 2015).

This demonstration examines the elevation of *HER2* expression in a subset of breast cancer samples.

An interactive tour of this use case can be viewed [here](#).

Mass cytometry

Finally, we explored a mass cytometry study involving more than 170,000 PBMCs from multiple donors before and after stimulation with BCR/FcR-XL (Bodenmiller *et al.*, 2012). We used iSEE to visualise and refine a gating analysis to obtain B cells, and to investigate differences in expression of the functional marker pS6 after stimulation.

An interactive tour of this use case can be viewed [here](#).

Conclusion

iSEE provides a general interactive interface for visual exploration of high-throughput biological data sets. Any study that can be represented in a SummarizedExperiment object can be used as input, allowing iSEE to accommodate a diverse range of ‘omics data sets. The interface is flexible and can be dynamically customised by the user; supports exploration of interactions between data aspects through colouring and linking between panels; and provides transparency and reproducibility during the interactive analysis, through code tracking and state reporting. The most obvious use of iSEE is that of data exploration for hypothesis generation during the course of a research project. However, we also anticipate that public instances of iSEE will accompany publications to enable authors to showcase important aspects of their data through guided tours.

Software availability

The iSEE package is available at <https://doi.org/doi:10.18129/B9.bioc.iSEE> (Soneson *et al.*, 2018) under an MIT license.

Source code of the development version of the package is available at <https://github.com/csoneson/iSEE>.

Code for the demonstrations and tours is available at <https://github.com/LTLA/iSEE2018>.

Archived source code of the version reported in this article and interactive tours is available from <http://doi.org/10.5281/zenodo.1247374> (Rue-Albrecht *et al.*, 2018)

Data availability

Data used in the described use cases is available from the following articles:

<http://doi.org/10.1038/nm.4216> (Tasic *et al.*, 2016)

<http://doi.org/10.1038/ncomms14049> (Zheng *et al.*, 2017)

<https://doi.org/10.1093/bioinformatics/btv377> (Rahman *et al.*, 2015)

<https://doi.org/10.1038/nbt.2317> (Bodenmiller *et al.*, 2012)

Competing interests

No competing interests were declared.

Grant information

ATLL was supported by core funding from Cancer Research UK [award no. 17197 to JM]. The work of FM is supported by the German Federal Ministry of Education and Research (BMBF 01EO1003).

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Acknowledgements

We thank the organisers and participants of the European Bioconductor Meeting 2017, where the idea for this package was first conceived. We also thank members of the Bioconductor community for their helpful suggestions. Finally, we thank John Marioni and Mark Robinson for their helpful comments on the manuscript.

References

- Aryee MJ, Jaffe AE, Corrada-Bravo H, *et al.*: **Minfi: a flexible and comprehensive Bioconductor package for the analysis of Infinium DNA methylation microarrays.** *Bioinformatics.* 2014; **30**(10): 1363–1369.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Bodenmiller B, Zunder ER, Finck R, *et al.*: **Multiplexed mass cytometry profiling of cellular states perturbed by small-molecule regulators.** *Nat Biotechnol.* 2012; **30**(9): 858–867.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Chang W, Borges Ribeiro B: **shinydashboard: Create Dashboards with 'Shiny'.** R package version 0.7.0. 2018.
[Reference Source](#)
- Chang W, Cheng J, Allaire JJ, *et al.*: **shiny: Web Application Framework for R.** R package version 1.0.5. 2017.
[Reference Source](#)
- Finak G, Frelinger J, Jiang W, *et al.*: **OpenCyto: an open source infrastructure for scalable, robust, reproducible, and automated, end-to-end flow cytometry data analysis.** *PLoS Comput Biol.* 2014; **10**(8): e1003806.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Fortin JP, Fertig E, Hansen K: **shinyMethyl: interactive quality control of Illumina 450k DNA methylation arrays in R [version 2; referees: 2 approved].** *F1000Res.* 2014; **3**: 175.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Ganz C: **rintrojs: A wrapper for the intro.js library.** *J Open Source Softw.* 2016.
[Publisher Full Text](#)
- Garnier S: **viridis: Default Color Maps from 'matplotlib'.** R package version 0.5.1. 2018.
[Reference Source](#)
- Gatto L, Breckels LM, Wiczorek S, *et al.*: **Mass-spectrometry-based spatial proteomics data analysis using pRoloc and pRolocdata.** *Bioinformatics.* 2014; **30**(9): 1322–1324.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Gentleman RC, Carey VJ, Bates DM, *et al.*: **Bioconductor: open software development for computational biology and bioinformatics.** *Genome Biol.* 2004; **5**(10): R80.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Huber W, Carey VJ, Gentleman R, *et al.*: **Orchestrating high-throughput genomic analysis with Bioconductor.** *Nat Methods.* 2015; **12**(2): 115–121.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Love MI, Huber W, Anders S: **Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2.** *Genome Biol.* 2014; **15**(12): 550.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Lun AT, Perry M, Ing-Simmons E: **Infrastructure for genomic interactions: Bioconductor classes for Hi-C, ChIA-PET and related experiments [version 2; referees: 2 approved].** *F1000Res.* 2016; **5**: 950.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Lun AT, Riesenfeld S, Andrews T, *et al.*: **Distinguishing cells from empty droplets in droplet-based single-cell rna sequencing data.** *bioRxiv.* 2018.
[Publisher Full Text](#)
- Marini F, Binder H: **Development of applications for interactive and reproducible research: a case study.** *Genomics Comput Biol.* 2016; **3**(1): e39.
[Publisher Full Text](#)
- NCBI Resource Coordinators: **Database Resources of the National Center for Biotechnology Information.** *Nucleic Acids Res.* 2017; **45**(D1): D12–D17.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- R Development Core Team: **R: A Language and Environment for Statistical Computing.** R Foundation for Statistical Computing, Vienna, Austria, 2008.
[Reference Source](#)
- Rahman M, Jackson LK, Johnson WE, *et al.*: **Alternative preprocessing of RNA-Sequencing data in The Cancer Genome Atlas leads to improved analysis results.** *Bioinformatics.* 2015; **31**(22): 3666–3672.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Rue-Albrecht K, Marini F, Sonesson C, *et al.*: **Interactive SummarizedExperiment Explorer.** *Zenodo.* 2018.
[Data Source](#)
- Scheeder C, Heigwer F, Boutros M: **HTSvis: a web app for exploratory data analysis and visualization of arrayed high-throughput screens.** *Bioinformatics.* 2017; **33**(18): 2960–2962.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Sonesson C, Lun A, Marini F, *et al.*: **iSEE: Interactive SummarizedExperiment Explorer.** R package version 1.0.1. 2018.
[Data Source](#)
- Tasic B, Menon V, Nguyen TN, *et al.*: **Adult mouse cortical cell taxonomy revealed by single cell transcriptomics.** *Nat Neurosci.* 2016; **19**(2): 335–46.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Wickham H: **ggplot2: Elegant Graphics for Data Analysis.** Springer-Verlag New York, 2009.
[Publisher Full Text](#)
- Zerbino DR, Achuthan P, Akanni W, *et al.*: **Ensembl 2018.** *Nucleic Acids Res.* 2018; **46**(D1): D754–D761.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Zheng GX, Terry JM, Belgrader P, *et al.*: **Massively parallel digital transcriptional profiling of single cells.** *Nat Commun.* 2017; **8**: 14049.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)

Open Peer Review

Current Referee Status:



Version 1

Referee Report 27 June 2018

doi:10.5256/f1000research.16293.r35044



Alejandro Reyes 

Department of Biostatistics and Computational Biology, Dana-Farber Cancer Institute, Boston, MA, USA

The authors implement an interactive tool, called iSEE, to perform exploratory analyses for high-throughput experiments. The tool inputs a Bioconductor core structure, the SummarizedExperiment object (coerced into a SingleCellExperiment object) and builds an interactive interphase for data exploration. iSEE provides several tools for data exploration by plotting features of an assay along with sample metadata, feature metadata, and reduced representations of the assays. Furthermore, iSEE enables users to interact with the plots and to dynamically link panels with different representations of the data. The analyses performed using iSEE are reproducible, since the code that was run through the graphic interphase can be downloaded.

Overall, the manuscript presents a very good idea and the code implementation is of great quality. iSEE will be very useful for people without programming background to perform basic analyses. I believe that the success of this tool will depend on whether the authors continue to develop it based on feature requests from users.

I don't have major concerns. However, I do have some recommendations to increase the interest of potential users.

1. Enable users to select more than one group of samples from the dimensionality reduction plots. Furthermore, it would be very useful to enable users to fill new columns of colData based on the interactive grouping of samples.
2. Enable users to retrieve an R data object if the initial input was modified during the analysis.
3. In the context of single-cell or large-scale analyses, it would be helpful to implement tools for differential abundance analyses and gene set enrichment analyses. For instance, one could think of an implementation where users manually define groups of cells from tSNE/PCA plots, retrieve the genes that are differentially expressed between these groups, and extract the pathways that are enriched among the differentially expressed genes.
4. When grouping samples manually on the tSNE/PCA plots, the violin plots of individual features (for example, genes) could be stratified based on these selections (e.g. plot one violin per group of selected points in the "Feature assay plot" panel). In the current implementation, it is only possible to color the points within the violin plot, which makes difficult to compare distributions between groups of samples.

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

I have read this submission. I believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Referee Report 20 June 2018

doi:10.5256/f1000research.16293.r35042



Lorena Pantano 

Department of Biostatistics, Harvard T. H. Chan School of Public Health, Boston, MA, USA

Authors show an interactive visualization tool for a very common data type used for many of the packages in Bioconductors (SummarizedExperiment). It has enough flexibility to explore all kind of information the object can contain, an interactive tool based on Rshiny, is customizable so it can be adapted to each user.

I only have minor some comments:

- Tutorial 2: step 10 gets the text box in the upper left of the windows, but I think it should be at other position since it says to change the y-axis of the plot. I think this happens when the user doesn't follow the instruction to click on to some button that should expand the menu with more options.
- It would be nice the tour re-start from the position it was left, with an option to start over. It happened many times that I click accidentally outside the box and I had to start over.
- In the cases the object doesn't have reducedDim for more than the 2 dimensions shown in the plot. I tried to use 3, and it gave an error. Maybe a more informative error would help the user to understand that there is no that information.

- I am not totally sure how to use the rintrojs package to generate a tool. It would be nice a reference to some documentation on how to do it or clarification if I am not understanding this correctly.
- For the features mentioned like code tracking and additional functionality, it would be nice to have a link to the vignette in the paper so the user can jump into how to get it done.
- I think it would be nice to make available a docker image with all the requirements to run iSEE installed. It would promote the use of the tool a lot among bioinformaticians working with non-computational researchers.
- It is nice to change the color for all the variables. I would add an example on how to change the palette for all categorical since the code would be slightly different than the one for continuous variables. It would make the user quickly using that option and avoid silly errors.
- I don't know if this is possible as it is right now, but it could be an option to load a RDA/RDS file containing the SE object instead of creating an app only for that data? That would open the door to deploy the tool independent of the data. For instance, I can see a scenario where iSEE is installed in a docker container, where the user just starts the image and when opening the browser at localhost:8787, there is an option to load a file with the object.

Congrats on the tools!

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

Referee Expertise: small rna, mirna, visualization, report, integration

I have read this submission. I believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Referee Report 19 June 2018

doi:10.5256/f1000research.16293.r35043



James W. MacDonald 

Department of Environmental and Occupational Health Sciences, University of Washington, Seattle, WA, USA

The iSEE package was developed to allow people to easily perform exploratory data analysis with data that are stored in a Bioconductor SummarizedExperiment object. A SummarizedExperiment container allows researchers to store one or more matrices of data, where the columns represent samples, and the rows represent either genomic positions or genomic features (genes, exons, transcription start sites, etc). In addition to the matrices of data, the SummarizedExperiment also contains two additional objects that describe the samples (the colData) and the rows (the rowData or rowRanges).

iSEE allows users to interactively plot the underlying data from a SummarizedExperiment, and also choose subsets of the data based on either interactive selection of data in a plot, or by selecting samples or genomic regions based on the colData or rowData. The chosen subsets can then be linked to other plots in the Shiny Dashboard. This simplifies what could be a complex process, allowing both experienced R users a quick way to check over their data, and allowing less experienced R users the ability to do things that they otherwise might not have been able to do.

All the underlying code generated while making interactive changes is saved and can be printed out later, in order to make the exploratory data analysis reproducible. This is an excellent feature, particularly for those who want to share observations with colleagues that may not be local.

The only negative for this package is that, being based on the Shiny framework, to allow a colleague to explore the data requires that the colleague either have R, iSEE, and all its dependencies installed, or that you have a server running all necessary packages that you can point the colleague to. This limits sharing with people who are not R savvy, but is a function of how Shiny works, rather than the iSEE package.

This is a high quality package, and given the generalizability of the SummarizedExperiment package, is applicable to a whole range of different data types. Given the ease of use, self documenting features, and applicability to multiple data types, this package will likely become very popular for exploratory data analysis.

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

I have read this submission. I believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research