GigaScience, 2020, 1–9

doi: xx.xxxx/xxxx
Manuscript in Preparation
Paper

PAPER

A single-cell RNA-seq Training and Analysis Suite using the Galaxy Framework

Mehmet Tekman[1*†], Bérénice Batut[1*†], Alexander Ostrovsky[2], Christophe Antoniewski[3], Dave Clements[2], Fidel Ramirez[4], Graham J Etherington[5], Hans-Rudolf Hotz[6], Jelle Scholtalbers[7], Jonathan R Manning[8], Lea Bellenger[3], Maria A Doyle[9], Mohammad Heydarian[2], Ni Huang[8,10], Nicola Soranzo[5], Pablo Moreno[8], Irene Papatheodorou[8], Anton Nekrutenko[11], James Taylor[2], Daniel Blankenberg[12], Rolf Backofen[1] and Björn Grüning[1*]

[1]Chair of Bioinformatics, University of Freiburg, Freiburg, Germany, and [2]Department of Biology, Johns Hopkins University, Baltimore, Maryland, USA and [3]ARTbio, Sorbonne Université, CNRS FR 3631, Inserm US 037, Paris, France and Institut de Biologie Paris Seine, Paris, France and [4]Boehringer Ingelheim, Biberach, Germany and [5]Earlham Institute, Norwich Research Park, Norwich, NR4 7UZ, United Kingdom and [6]Friedrich Miescher Institute for Biomedical Research, Basel, Switzerland and Swiss Institute of Bioinformatics, Basel, Switzerland and [7]Helmholtz Institute for Translational Oncology, Mainz, Germany and [8]European Molecular Biology Laboratory, European Bioinformatics Institute, EMBL-EBI, Hinxton, United Kingdom and [9]Research Computing Facility, Peter MacCallum Cancer Centre, Melbourne, Victoria 3000, Australia and Sir Peter MacCallum Department of Oncology, The University of Melbourne, Victoria 3010, Australia and [10]Wellcome Sanger Institute, Cambridge, United Kingdom and [11]Department of Biochemistry and Molecular Biology, The Pennsylvania State University, University Park, Pennsylvania, USA and [12]Genomic Medicine Institute, Lerner Research Institute, Cleveland Clinic, Cleveland, Ohio, USA

[*]tekman@informatik.uni-freiburg.de; bebatut@informatik.uni-freiburg.de; gruening@informatik.uni-freiburg.de
[†] These authors contributed equally to this work.

Abstract

Background The vast ecosystem of single-cell RNA-seq tools has until recently been plagued by an excess of diverging analysis strategies, inconsistent file formats, and compatibility issues between different software suites. The uptake of 10x Genomics datasets has begun to calm this diversity, and the bioinformatics community leans once more towards the large computing requirements and the statistically-driven methods needed to process and understand these ever-growing datasets.
Results Here we outline several Galaxy workflows and learning resources for scRNA-seq, with the aim of providing a comprehensive analysis environment paired with a thorough user learning experience that bridges the knowledge gap between the computational methods and the underlying cell biology. The Galaxy reproducible bioinformatics framework provides tools, workflows and trainings that not only enable users to perform one-click 10x preprocessing, but also empowers them to demultiplex raw sequencing data manually. The downstream analysis supports a wide range of high-quality interoperable suites separated into common stages of analysis: inspection, filtering, normalization, confounder removal and clustering. The teaching resources cover an assortment of different concepts from computer science to cell biology. Access to all resources is provided at the `singlecell.usegalaxy.eu` portal.
Conclusions The reproducible and training-oriented Galaxy framework provides a sustainable HPC environment for users to run flexible analyses on both 10x and alternatively derived datasets. The tutorials from the Galaxy Training Network along with the frequent training workshops hosted by the Galaxy Community provide a means for users to learn, publish and teach scRNA-seq analysis.

Key words: scRNA; Galaxy; resources; HPC; single-cell; 10x; Training; Web;

Key Points

- Single-cell RNA-seq has stabilised towards 10x Genomics datasets.
- Galaxy provides rich and reproducible scRNA-seq workflows with a wide range of robust tools.
- The Galaxy Training Network provides tutorials for the processing of both 10x and non-10x datasets.

Background

Single-Cell RNA-seq and cellular heterogeneity. The continuing rise in single cell technologies has led to previously unprecedented levels of analysis into cell heterogeneity within tissue samples, providing new insights into developmental and differentiation pathways for a wide range of disciplines. Gene expression studies are now performed at a cellular level of resolution, which compared to bulk RNA-seq methods, allows researchers to model their tissue samples as distributions of different expressions instead of as an average.

Pathways from Single-cell data. The various expression profiles uncovered within tissue samples infer discrete cell types which are related to one another across an "expression landscape". The relationships between the more distinct profiles are inferred via distance-metrics or manifold learning techniques. Ultimately, the aim is to model the continuous biological process of cell differentiation from multipotent stem cells to distinct mature cell types, and infer lineage and differentiation pathways between transient cell types [1].

Elucidating Cell Identity. Trajectory analysis which integrates the up or down regulation of significant genes along lineage branches can then be performed in order to uncover the factors and extracellular triggers that can coerce a pluripotent cell to become biased towards one cell fate outcome compared to another. This undertaking has created a new frontier of exploration in cell biology, where researchers have assembled reference maps for different cell lines for the purpose of fully recording these cell dynamics and their characteristics in which to create a global "atlas" of cells [2, 3].

Pitfalls and Technical Challenges

Sequencing sensitivity and Normalization. With each new protocol comes a host of new technical problems to overcome. The first wave of software utilities to deal with the analysis of single cell datasets were statistical packages, aimed at tackling the issue of "dropout events" during sequencing, which would manifest as a high prevalence of zero-entries in over 80% of the feature-count matrix. These zeroes were problematic, since they could not be trivially ignored as their presence stated that either the cell did not produce any molecules for that transcript, or that the sequencer simply did not detect them. Normalisation techniques originally developed for bulk RNA-seq had to be adapted to accommodate for this uncertainty, and new ones were created that harnessed hurdle models, data imputation via manifold learning techniques, or by pooling subsets of cells together and building general linear models [4].

Improvements in sequencing. With the downstream analysis packages attempting to solve the dropouts via stochastic methods, the upstream sequencing technologies also aspired to solve the capture efficiency via new well, droplet, and flow cytometry based protocols, all of which lend importance to the process of barcoding sequencing reads.

In each protocol, cells are tagged with cell barcodes such that any reads derived from them can be unambiguously assigned to the cell of origin. The inclusion of unique molecular identifiers (UMIs) are also employed to mitigate the effects of amplification bias of transcripts within the same cell. The detection, extraction, and (de-)multiplexing of cell barcodes and UMIs is therefore one of the first hurdles researchers encounter when receiving raw `FASTQ` data from a sequencing facility.

The Burgeoning Software Ecosystem

Since its conception, several different packages and many pipelines have been developed to assist researchers in the analysis of scRNA-seq [5, 6]. The vast majority of these packages were written for the R programming language since many of the novel normalisation methods developed to handle the dropout events depended on statistical packages that were primarily R-based [7]. Standalone analysis suites emerged as the different authors of these packages rapidly expanded their methods to encapsulate all facets of the single-cell analysis, often creating compatibility issues with previous package versions. The Bioconductor repository provided some much-needed stability in this regard by hosting stable releases, but researchers were still prone to building directly from repository sources in order to reap the benefits of new features in the upstream versions [8, 9].

Nonexchangeable Data Formats. Another issue was the proliferation of the many different and quickly evolving R-based file formats for processing and storing the data, such as `SingleCellExperiment` as used by the *Scater* suite, `SCSeq` from *RaceID*, and `SeuratObject` from *Seurat* [10, 11]. Many new packages would cater only towards one format or suite, leading to the common problem that data processed in one suite could not be reliably processed by methods in another. This incompatibility between packages fuelled a choice of one analysis suite over another, or conversely required researchers to dig deeper into the internal semantics of R `S4` objects in order to manually slot data components together [12]. These problems only accelerated the rapid development of these suites, leading to further version instability. As a result of this analysis diversity, there are many tutorials on how to perform scRNA-seq analysis each oriented around one of these pipelines [13].

Error propagation and Analysis Uncertainty. Different pipelines produce different results, where the stochastic nature of the analyses means that any uncertainty in a crucial quality control step upstream, such as filtering or the removal of unwanted variability, can propagate forward into the downstream sections to yield wildly different results on the same data. This uncertainty, and the statistically-driven methods to overcome

Compiled on: June 6, 2020.
Draft manuscript prepared by the author.

them, leaves a wide knowledge gap for researchers simply trying to understand the underlying dynamics of cell identity.

Rise of 10x Genomics

10x Launch. In 2015, 10x Genomics released their *GemCode* product, which was a droplet-seq based protocol capable of sequencing tens of thousands of cells with an average cell quality higher than other facilities [14]. This unprecedented level of throughput steadily gained traction amongst researchers and startups seeking to perform single-cell analysis, and thus 10x datasets began to prevail in the field.

10x Analysis Software. 10x Genomics provided software that was able to perform much of the pre-processing, and provided feature-count matrices in a transparent HDF5-based format which provided a means of efficient matrix storage and exchange, and conclusively removed the restriction for downstream analysis modules to be written in R.

ScanPy, a popular alternative. The *ScanPy* suite [15], written in Python, using its own HDF5-based `AnnData` format became a valid alternative for analysing 10x datasets. The Seurat developers had similar aspirations and soon adopted the `LOOM` format, another `HDF5` variant. However, the popularity of ScanPy rose as it began to integrate the methods of other standalone packages into its codebase, making it the natural choice for users who wanted to achieve more without compromising on compatibility between different suites [9].

Solutions in the Cloud

Accessible Science. As the size of datasets scaled, so did the computing resources required to analyse them, both in terms of the processing power and in storage. Galaxy is an open-source biocomputing infrastructure that exemplifies the three main tenets of science: reproducibility, peer review, and open-access - all freely accessible within the web-browser [16]. It hosts a wide range of highly-cited bioinformatic tools with many different versions, and enables users to freely create their own workflows via a seamless drag-and-drop interface.

Reproducible Workflows. Galaxy can make use of *Conda* or *Containers* to setup tool environments in order to ensure that the bioinformatic tools will always be able to run, even when the library dependencies for a tool have changed, by building tools under locked version dependencies and bundling them together in a self-contained environment [17]. These environments provide a concise solution for the package version instability that plagues scRNA-seq analysis notebooks, both in terms of reproducibility and analysis flexibility. A user could keep the quality control results obtained from an older version of ScanPy, whilst running a newer ScanPy version at the clustering stage to reap the benefits of the later improvements in that algorithm. By allowing the user to select from multiple versions of the same tool, and by further permitting different versions of the tools within a workflow, Galaxy enables an unprecedented level of free-flow analysis by letting researchers pick and choose the best aspects of a tool without worrying about the underlying software libraries [18]. The burden of software incompatibility and choice of programming language that plagued the scRNA-seq analysis ecosystem before, are now completely alleviated from the user.

User-driven Custom Workflows. Analyses are not limited to one pipeline either, as the datasets which are passed between tools can easily be interpreted by a different tool that is capable of reading that dataset. In the case of scRNA-seq, Galaxy can convert between `CSV`, `MTX`, `LOOM` and `AnnData` formats. This interexchange of modules from different tools further extends the flexibility of the analysis by again letting the user decide which component of a tool would be best suited for a specific part of an analysis.

Training Resources. Galaxy also provides a wide range of learning resources, with the aim of guiding users step-by-step through an analysis, often reproducing the results of published works. The teaching and training materials are part of the Galaxy Training Network (GTN), which is a worldwide collaborative effort to produce high-quality teaching material in order to educate users in how to analyse their data, and in turn to train others of the same materials via easily deployable workshops backed by monthly stable releases of the GTN materials [19]. Training materials are provided on a wide variety of different topics, and workshops are hosted regularly, as advertised on the Galaxy Events web portal. The GTN has grown rapidly since its conception and gains new volunteers every year who each contribute and coordinate training and teaching events, maintain topic and subtopics, translate tutorials into multiple languages, and provide peer review on new material [20].

Methods

Stable Workflows in Galaxy. The analysis of scRNA-seq within Galaxy was a two-pronged effort concentrated on bringing high quality single-cell tools into Galaxy, and providing the necessary workflows and training to accompany them. As mentioned in the previous section, this effort needed to overcome incompatible file format issues, unstable packages due to rapid development, and needed to establish a standardised basis for the analysis.

Tutorials. The tutorials are split into two main parts as outlined in Figure 1: first, the *pre-processing* stage which constructs a count matrix from the initial sequencing data; second, the cluster-based *downstream analysis* on the count matrix. These stages are very different from one another in terms of their information content, since the pre-processing stage requires the researcher to be more familiar with wetlab sequencing protocols than your average bioinformatician would normally know, and the downstream analysis stage requires the researcher to be familiar with statistics concepts that a wetlab scientist might not be too familiar with. The tutorials are designed to broadly appeal to both the biologist and the statistician, as well as complete beginners to the entire topic.

Pre-processing Workflows

The pre-processing scRNA-seq materials tackle the two most common use-cases that researchers will encounter when they first begin the field: processing scRNA-seq data from 10x Genomics, and processing data generated from alternative protocols. For instance, microwell-based protocols have been known to yield more features and display lower levels of dropouts compared to 10x, and so we accommodate for them by providing a more customizable path through the pre-processing stage [21].

Barcode Extraction. Before the era of 10x Genomics, scRNA-seq data had to be demultiplexed, mapped, and quantified. The demultiplexing stage entails an intimate knowledge of cell barcodes and Unique Molecular Identifiers (UMIs) which are protocol dependent, and expects that the bioinformatician knows

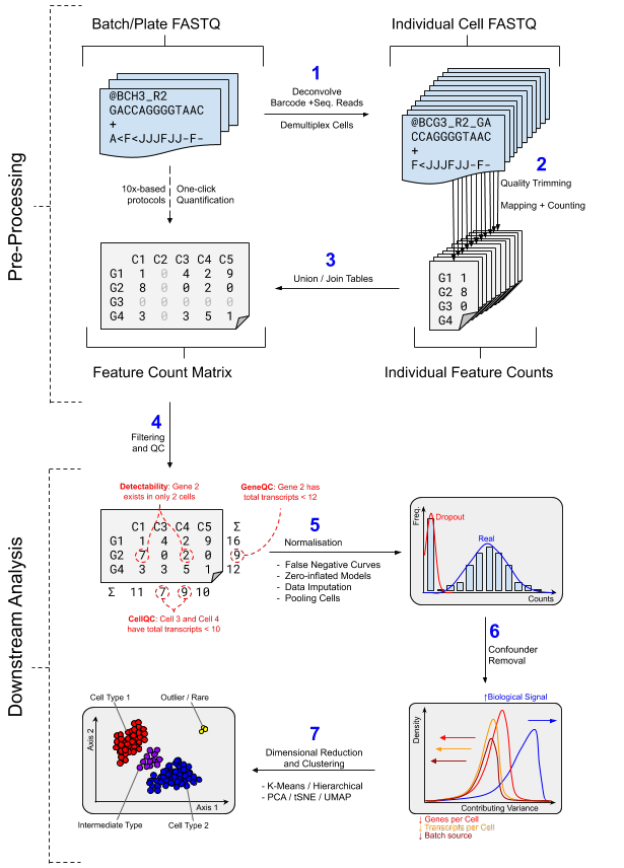


Figure 1. The main stages of single-cell analysis, separated broadly into the upper and lower stages of pre-processing and downstream analysis, respectively. The upper part illustrates the two main routes to generating a count-matrix from sequencing data; via one-click quantification solutions, or through manual demultiplexing. The lower part describes the four main stages required to perform cluster-based analysis from the count-matrix, through filtering, normalisation, confounder removal, and embedding.

exactly where and how the data was generated. One common pitfall at this very first stage is estimating how many cells to expect from the FASTQ input data, and this requires three crucial pieces of information: which reads contain the barcodes (or precisely, which subset of both the forward and reverse reads contains the barcodes); of these barcodes, which specific ones were actually used for the analysis; and how to resolve barcode mismatches/errors.

Barcode Estimation. Naive strategies involve using a known barcode template and querying against the FASTQ data to profile the number of reads that align to a specific barcode, often employing 'knee' methods to estimate this amount [22]. However, this approach is not robust, since certain cells are more likely to be over-represented compared to others, and some cell barcodes may contain more unmappable reads compared to others, meaning that the metric of higher library read depth is not necessarily correlated with a better-defined cell. Ultimately, the bioinformatician must inquire directly with the sequencing lab as to which cell barcodes were used, as these are often not specific to the protocol but to the technician who designed them, with the idea that they should not align to a specific reference genome or transcriptome.

One-click Pre-processing

Quantification with Cell Ranger. 10x Genomics simplified the scRNA package ecosystem by using a language independent file format, and streamlining much of the barcode particularities with their *Cell Ranger* pipeline, allowing researchers to focus more on the internal biological variability of their datasets [23].

Quantification with STARsolo. The 10x pre-processing workflow given in Galaxy uses *RNA STARsolo* utility as a drop-in replacement for *Cell Ranger*, because not only is it a feature update of the already existing *RNA STAR* tool in Galaxy, but because it boasts a ten-fold speedup in comparison to *Cell Ranger* and does not require Illumina lane-read information to perform the processing [24, 25].

Other Approaches. The pre-processing workflow for these "one-click" solutions consume the same datasets and yield approximately the same count matrices by following similar modes of barcode discovery and quantification. Within Galaxy, there is also *Alevin* (paired with *Salmon*) and *scPipe* which can both also perform the necessary demultiplexing, (alignment-free) mapping, and quantification stages in a single step [26, 27, 28].

Flexible Pre-processing

CELSeq2 Barcoding. The custom pre-processing workflow follows the CELSeq2 protocol using the barcoding strategies of the Freiburg Max-Planck Institute laboratory as its main template, but the training is flexible to accommodate any droplet or well-based protocol [29].

Manual Demultiplexing and Quantification. The training pictographically guides users through the concepts of extracting cell barcodes from the protocol, explains the significance of UMIs in the process of read deduplication with illustrative examples, and instructs the user in the process of performing further quality controls on their data during the post-mapping process via *RNA STAR* and other tools that are native to Galaxy.

Training the User. At each stage, the user's knowledge is queried via question prompts and expandable answer box dialogs, as well as helpful hints for future processing in comment boxes, all written in the transparent Markdown specification developed for contributing to the GTN.

Downstream Workflows

Common Stages of Analysis. The downstream modules are defined by the five main stages of downstream scRNA-seq analysis: filtering, normalisation, confounder removal, clustering, and trajectory inference. There are three workflows to aid in this process (two of which are shown in Figure 2, each sporting a different well-established scRNA-seq pipeline tool.

Scater-based Quality Control. The Scater pipeline follows a visualise-filter-visualise paradigm which provides an intuitive means to perform quality control on a count matrix by use of repeated incremental changes on a dataset through the use of PCA and library size based metrics [30]. Once this pre-analysis stage is complete, the full downstream analysis can be performed by workflows based on the following suites: RaceID and ScanPy.

RaceID Suite. RaceID was developed initially to analyse rare cell transcriptomes whilst being robust against noise, and thus is ideal for working with smaller datasets in the range of 300 to 1000 cells. Due to its impressive cell lineage and fate predictions models, it can also be used on larger datasets with some scaling costs.

ScanPy Suite. ScanPy was developed as the Python alternative to the innumerable R-based packages for scRNA-seq which was

the dominant language for such analyses, and it was one of the first packages with native 10x genomics support. Since then it has grown substantially, and has been re-implementing much of the newer R-based methods released in BioConductor as "recipe" modules, thereby providing a single source to perform many different types of the same analysis.

The workflows derived from both these suites emulate the five main stages of analysis mentioned previously, where filtering, normalisation, and confounder removal are typically separated into distinct stages, though sometimes merged into one step depending on the tool.

Filtering

Cell and Gene Removal. During the filtering stage, the initial count matrix removes low-quality or unwanted cells using commonly used parameters such as minimum gene detection sensitivity and minimum library size, and low-quality genes are also removed under similar metrics, where the minimum number of cells for a gene to be included is decided. The Scater pre-analysis workflow can also be used here to provide a PCA-based method of feature selection so that only the highly variable genes are left in the analysis.

Disadvantages of Filtering. There is always the danger of over-filtering a dataset, whereby setting overzealous lower-bound thresholds on gene variability, can have the undesired effect of removing essential housekeeping genes. These relatively uniformly expressed genes are often required for setting a baseline to which the more desired differentially expressed genes can be selected from. It is therefore important that the user first performs a naive analysis and only later refine their filtering thresholds to boost the biological signal.

Normalisation

Library Size Normalisation. The normalisation step aims to remove any technical factors that are not relevant to the analysis, such as the library size, where cells sharing the same identity are likely to differ from one another more by the number of transcripts they exhibit, than due to more relevant biological factors.

Intrinsic Cell Factors. The first and foremost is cell capture efficiency, where different cells produce more or less transcripts based on the amplification and coverage conditions they are sequenced in. The second is the presence of dropout events which manifest as a prevalence of "zeroes" in the final count matrix. Whether a "zero" is imputable to the lack of detection of an existing molecule or to the absence of the molecule in the cell is uncertain. This uncertainty alone has led to a wide selection of different normalisation techniques that try to model this expression either via hurdle models, or imputing the data via manifold learning techniques, or working around the issue by pooling subsets of cells together [31].

In this regard, both the RaceID and ScanPy workflows offer many different normalisation techniques, and users are encouraged to take advantage of the branching workflow model of Galaxy to explore all possible options.

Confounder Removal

Regression of Cell Cycle Effects. Other sources of variability stem from unwanted biological contributions known as confounder effects, such as cell cycle effects and transcription. Depending on what stage of the cell cycle a cell was sequenced at, two cells of the same type might cluster differently because one might have more transcripts due to it being in the M-phase of the cell cycle. Library sizes notwithstanding, it is the variability in specific cell cycle genes that can be the main driving factor in the overall variability. Thankfully, these effects are easy to regress

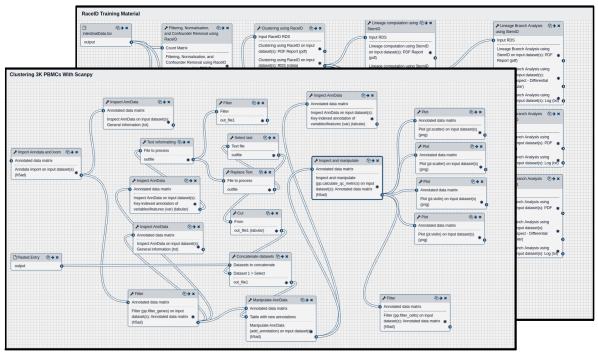


Figure 2. Downstream analysis workflows as shown in the Galaxy Workflow Editor for (top) RaceID and (bottom) ScanPy, each displaying modules symbolizing the five main stages of analysis.

out, and we replicate an entire standalone ScanPy workflow dedicated to detecting and visualising the effects based on the original notebook [32].

Transcriptional Bursting. The transcription effects are harder to model, as these are semi-stochastic and are as of yet still not well understood. In bulk RNA-seq the expression of genes undergoing transcription are averaged to give "high" or "low" signals producing a global effect that gives the false impression that transcription is a continuous process. The reality is more complex, where cells undergo transcription in "bursts" of activity followed by periods of no activity, at irregular intervals [33]. At the bulk level these discrete processes are smoothed to give a continuous effect, but at the cell level it could mean that even two directly adjacent cells of the same type normalised to the same number of transcripts can still have different levels of expression for a gene due to this process. This is not something that can be countered for, but it does educate the users in which factors they can or cannot control in an analysis, and how much variability they can expect to see.

Clustering and Projection

Dimension Reduction and Clustering. Once a user has obtained a count matrix they are confident with, they are then guided through the process of dimension reduction (with choice of different distance metrics), choosing a suitable low-dimensional embedding, and performing clustering through commonly-used techniques such as k-means, hierarchical, and neighbourhood community detection. These complex techniques are illustrated in layman's terms through the use of helpful images and community examples. For example, the GTN ScanPy tutorial explains the Louvain clustering approach[34] via a standalone slide deck to assist in the workflow [35].

Commonly-used Embeddings. The clustering and the cluster inspection stages are notably separated into distinct utilities here, with the understanding that the same initial clustering can appear dissimilar under different projections, e.g. t-distributed Stochastic Network Embedding (tSNE) against Uniform Manifold Approximation and Projection (UMAP) [36, 37]. Ultimately the user is encouraged to play with the plotting parameters to yield the best looking clusters.

Static Plots or Interactive Environments. Cluster inspection tools are available that allow users to easily generate static plots tailored to pipeline-specific information as originally defined by the software package authors. However, the `AnnData` and `LOOM` specifications store this map projection data separately, enabling the use of a plethora of possible plotting tools, in-

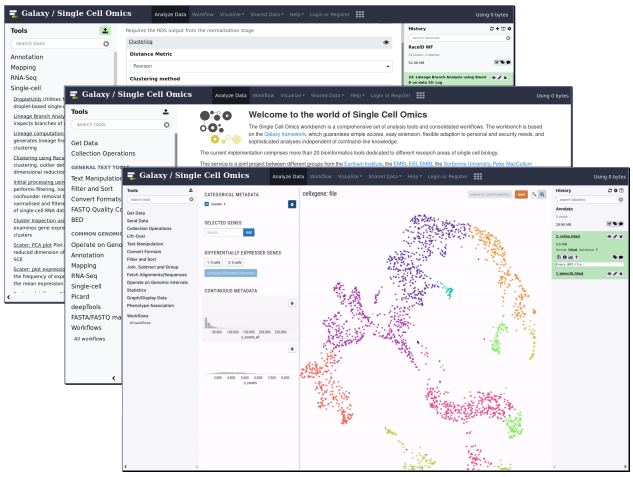


Figure 3. Galaxy Training Network hosting a comprehensive suite of tools, trainings, and workflows to perform scRNA-seq analysis.

cluding HTML5-based interactive visualisations, such as *cellxgene*[38], that permit on-demand querying and rendering of individual cell features without having to generate static images. A collection of these Galaxy interactive tools can be accessed at the website `live.usegalaxy.eu`. Though these tools are excellent at dynamically displaying map projections, especially 3-dimensional ones, further computation must be performed to complete a full pseudotime analysis.

Pseudotime Trajectory Analysis

Inferring Developmental Pathways. The cell pseudotime series analysis is often referred to as the trajectory inference stage, since cells are ordered along a trajectory to reflect the continuous changes of gene expression along a developmental pathway under the assumption that the cells are transitioning from one pluripotent type to another less-potent type.

Pseudotime Techniques. For the trajectory inference stage, there is the Partition-based Graph Abstraction (PAGA) technique championed by ScanPy [39], and there is also the FateID and StemID packages for the RaceID workflow [40]. The former provides a level of graph abstraction to the datasets in order to infer a community graph structure, which it can use to learn the shape of the data and infer pathways between neighbourhoods. The latter is more intuitive, in that it constructs a minimal spanning tree of related clusters that infer lineage, and cell fate decisions that can be explored by querying branches in the tree, as a function of the genes which are up or down regulated along the currently explored pathway. The statistical strength and significance of each pathway guides the user along more valid trajectories that would more accurately reflect the biological variation occurring within transitioning cells.

Sharing Reference Maps

The insights and novel cell types discovered in these analyses can also be integrated into the Human Cell Atlas portal [41], which is an initiative that aims to classify unique or rare cell types as well as their transitive properties in order to build a comprehensive map of cells that can be used to investigate the various differentiation pathways of multipotent stem cells in the human body.

Galaxy Training Network

Tutorial Hierarchy. Tutorials in the GTN are grouped by topic, e.g. Variant Calling, Transcriptomics, Assembly, etc. These tutorials can also declare prerequisites, so that users can review required concepts from previous tutorials, e.g. quality control checks from bulk RNA-seq still being used in scRNA-seq. Not only does this allow users to derive a clear route through the range of training materials, but it also empowers them to choose their own learning path through the network of topics. In particular for scRNA-seq, users can start their training from pre-processing tutorials and continue till downstream analysis.

Tutorial Structure. Tutorials usually consist of a hands-on workflow that guides the user through an analysis with Galaxy utilising a step-by-step approach, and is often accompanied with a slide deck that either serves to explain standalone concepts more concisely, or is used during workshops and trainings as a way to introduce the user to the topic. In an effort to maintain reproducibility in science, all tutorials require example workflows, and all materials needed to run the workflows and tutorials are hosted for free with open access at Zenodo with a permanent DOI tag.

User-driven Contribution. The user contributions are the heart of the GTN community, and options are given to appeal to different levels of contribution. At the casual level, each tutorial has at the bottom an anonymous feedback form that rates the quality of the tutorial and asks for hints on what could be improved, which the tutorial authors can then act on. At the more eager level, users can contribute directly to the material hosted at the GitHub repository using the approachable GTN Markdown format, which further empowers contributors to not only adapt existing material, but to also write tutorials in their own specialist topics. The GitHub code reviews paired with the plain-text GTN Markdown format, facilitate easy peer-review of tutorial topics by using standard diff utilities.

Galaxy Subdomains and Environments

Subdomains Encapsulate Relevant Tools. The Galaxy tools and the GTN are further tied together by Galaxy subdomains, that better serve the various topics within their own self-contained environments. These complement the training materials by providing only the necessary Galaxy tools in order to not trouble the user with unrelated tools that might not be so relevant to the material, e.g. Variant Analysis tools are not included in an scRNA-seq environment. This also has the benefit that smaller more specialised Galaxy instances can be packaged and deployed, avoiding the overhead of presenting the entire Galaxy tool repertoire.

Single Cell and Human Cell Atlas. In this light, the `singlecell.usegalaxy.eu` subdomain hosts the entirety of the single-cell materials, tools, workflows, and single-cell related events. A table containing the full list of tools in the subdomain, as well as their application to the previously mentioned stages of scRNA-seq analysis is given in Supplemental Table 1. Human Cell Atlas community members, led by the European Bioinformatics Institute and the Wellcome Sanger Institute have their own subdomain at `humancellatlas.usegalaxy.eu` [42], providing access to widely applicable tools including ScanPy, Seurat and *Monocle3* [43], but also specialist tools such as those for cell type prediction (including *scmap* [44], *scPred* [45] and *Garnett* [46]).

Analysis in Galaxy workflows. The tools outlined in the *Downstream Workflows* subsection expose the full set of parameters of their underlying program suites, in order to serve the same level of customisation that the users would expect when run-

ning a notebook-based analysis. This suits the needs of most researchers, but some are more used to processing the data directly in a language-driven notebook environment.

Galaxy Interactive Environments (GIE). For the more computer programming-oriented users, Galaxy hosts interactive environments at `live.usegalaxy.eu` which allows users to spin up their own Jupyter [47] or RStudio [48] notebooks whilst harnessing the same cloud compute infrastructure. Here, users can import their Galaxy datasets, process them in their own desired manner, and export them back into their histories in a similar way to how datasets are treated in workflows.

List of GIEs. In addition to interactive notebooks, the GIE also boasts a selection of other interactive tools such as the previously mentioned *cellxgene* featured in Figure 3, as well as *SPARQL* a query language interface, *BAM/VCF IOBIO* a file format analysis viewer [49], *EtherCalc* a web spreadsheet [50], *PHINCH* a metagenomes visualiser [51], *Wallace* a species modelling platform [52], *WILSON* an omics visualiser [53], *IDE* for materials science, *Panoply* a netCDF viewer [54], *HiGlass* a Hi-C data visualiser [55], and even an XFCE Virtual Desktop environment [56].

Discussion

Growth of scRNA Training Materials. The single-cell materials on the GTN are growing substantially every year, with at first only one pre-processing tutorial in 2018, one downstream tutorial at the start of 2019, and at the current time of writing three pre-processing tutorials and three downstream analysis workflows, further accompanied by slide decks and interactive visualisations. Single-cell Galaxy workshops based on these materials have been given at the Single-Cell RNAseq Training Course 2018 at the Earlham Institute, the 2019 Galaxy Community Conference (GCC2019), within the Freiburg MeInBio consortium, and at the Association of Biomolecular Resource Facilities (ABRF). The trainings also lend themselves seamlessly to online Webinars which have proved useful during the COVID-19 lockdown period.

Reproducible Cloud-based Analysis. The advent of scRNA-seq analysis within the Galaxy framework re-echoes the efforts to standardise the analysis of scRNA-seq with the promise of presenting reproducible research. The burden of computation on the ever-growing size of the datasets is shifted to the cloud computing resources, and as scRNA sequencing technology scales, more researchers are likely to migrate towards cloud-based solutions in order to reap the benefits of superior computing abilities and storage capabilities. Ultimately, the Galaxy framework abstracts the user from the many non-trivial technicalities of the analysis, and exposes them to a legible interface of tools that they can pick and choose from.

Longevity and Accessibility. The community regularly comes together during scheduled code festivals (CoFests) or hackathons to review, contribute, and actively maintain the training materials. The number of community contributions have steadily increased over the last four years [16], and this growing trend ensures that the Galaxy resources will stay current and adapt to changes in scRNA sequencing technology and analysis methods if necessary. The GTN also makes use of language translation tools to provide international interpretations of the training materials in order to reach a wider more internationally diverse audience.

Future of scRNA-seq in Galaxy. The capacity for growth of scRNA-seq in Galaxy is limitless, with the continuing acquisition of new single-cell tools being incorporated into Galaxy workflows, and the expanding GTN community bringing more expert-level contributions to the training material. The vestiges of incompatible libraries and in-exchangeable file formats are unburdened from the user as the epoch of web-based tools and strong biocomputing frameworks become more dominant. From the first data upload to the final finishing touches of a customized workflow, the single cell Galaxy portal upholds the ideals of open science by supporting the user all the way from the initial training to the final publication, where they can export and share their results with a single click.

Availability of source code and requirements (optional, if code is present)

Lists the following:

- Project name: Single-Cell RNA-seq Analysis in Galaxy
- Project home page: `singlecell.usegalaxy.eu`
- Operating system(s): Web-based, Platform independent
- License: GNU GPL v3 Any restrictions to use by non-academics: e.g. licence needed

Availability of supporting data and materials

All datasets used in the GTN are independently hosted at Zenodo and are easily findable under the tag "Galaxy Training Network", as well as being directly hosted within the Galaxy Data Libraries on the UseGalaxy.eu server.

The tool wrappers which serve as the functional components of the many different single-cell analysis tools are hosted at the *GitHub Tools-IUC* repository, as well as at the Galaxy Toolshed under the category of "Transcriptomics".

Declarations

List of abbreviations

DOI Digital Object Identifier
GTN Galaxy Training Network
HDF5 Hierarchical Data Format 5
HPC High Performance Computing
PAGA Partition-based Graph Abstraction
PCA Principal Component Analysis
scRNA Single-Cell RNA
tSNE t-distributed Stochastic Network Embeddings
UMAP Uniform Manifold Approximation and Projection
UMI Unique Molecular Identifier

Consent for publication

Not applicable.

Competing Interests

The author(s) declare that they have no competing interests.

Funding

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) -

22977937/GRK2344, the BBSRC Core Strategic Programme Grants BBS/E/T/000PR9814, BBS/E/T/000PR9817, BBS/E/T/000PR9818, and BBS/E/T/000PR9819, Core Capability Grant BBS/E/T/000PR9816 at the Earlham Institute, and the National Institutes of Health grant U41HG006620.

The European Galaxy project is in part funded by Collaborative Research Centre 992 Medical Epigenetics (DFG grant SFB 992/1 2012) and German Federal Ministry of Education and Research (BMBF grants 031 A538A/A538C RBC, 031L0101B/031L0101C de.NBI-epi).

Author's Contributions

B.G. conceived the project, and created the singlecell subdomain. M.T. and B.B. wrapped RaceID and ScanPy respectively, and created all initial workflows and trainings. A.O., B.G., C.A., D.C., D.B., G.J.E, H.R.H., J.R.M., L.B., M.D., M.H., N.H., F.R., J.S., N.S., and P.M.. developed tools or made functional contributions to the tools and training materials. D.B., I.P., A.N., J.T. and R.B. supported the development of the project. M.T. wrote the original draft manuscript. All authors have read, made suggestions, and ultimately approved the final manuscript.

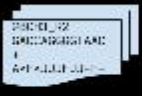
Acknowledgements

We thank the bioinformatics group at the University of Freiburg for the development and hosting of the European Galaxy server, Monika Degen-Hellmuth at the Backofen Lab for her assistance in the organization of the project, the Institut Français de Bioinformatique (IFB) for its support of the ARTbio team, and we also thank the worldwide contributions from users and developers towards the Galaxy Project and all upstream authors and contributors of the software ecosystem that we use and rely on.

References

1. Wagner A, Regev A, Yosef N. Revealing the vectors of cellular identity with single-cell genomics. Nat Biotechnol 2016;34(11):1145.
2. Rozenblatt-Rosen O, Stubbington MJ, Regev A, Teichmann SA. The Human Cell Atlas: from vision to reality. Nature 2017;550(7677):451–453.
3. Briggs, James A W, Caleb W, Daniel E M, Sean P, Leonid K, et al. The dynamics of gene expression in vertebrate embryogenesis at single-cell resolution. Science 2018;360(6392):eaar5780.
4. Camara PG. Methods and challenges in the analysis of single-cell RNA-sequencing data. Curr Opin Syst Biol 2018;7:47–53.
5. McCarthy DJ, Campbell KR, Lun AT, Wills QF. Scater: preprocessing, quality control, normalization and visualization of single-cell RNA-seq data in R. Bioinformatics 2017;33(8):1179–1186.
6. Satija R, Farrell JA, Gennert D, Schier AF, Regev A. Spatial reconstruction of single-cell gene expression data. Nat Biotechnol 2015;33(5):495.
7. Amezquita RA, Lun AT, Becht E, Carey VJ, Carpp LN, Geistlinger L, et al. Orchestrating single-cell analysis with Bioconductor. Nat Methods 2019;p. 1–9.
8. Satija R, Farrell JA, Gennert D, Schier AF, Regev A, List of Seurat Releases;. Accessed: 2020-01-10. https://github.com/satijalab/seurat/releases.
9. Wolf APTF F, ScanPy Release Notes;. Accessed: 2020-01-10. https://scanpy.readthedocs.io/en/stable/release-notes.html.
10. Lun A, Risso D, Korthauer K. SingleCellExperiment: S4 classes for single cell data. R package version 2018;1(0).
11. Grün D, Lyubimova A, Kester L, Wiebrands K, Basak O, Sasaki N, et al. Single-cell messenger RNA sequencing reveals rare intestinal cell types. Nature 2015;525(7568):251.
12. Ruckdeschel P, Kohl M, Stabla T, Camphausen F. S4 classes for distributions. The Newsletter of the R Project Volume 6/2, May 2006 2006;6:2.
13. Luecken MD, Theis FJ. Current best practices in single-cell RNA-seq analysis: a tutorial. Mol Syst Biol 2019;15(6).
14. Vickovic S, Ståhl PL, Salmén F, Giatrellis S, Westholm JO, Mollbrink A, et al. Massive and parallel expression profiling using microarrayed single-cell sequencing. Nat Commun 2016;7:13182.
15. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. Genome Biol 2018;19(1):15.
16. Afgan E, Baker D, Batut B, Van Den Beek M, Bouvier D, Čech M, et al. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic Acids Res 2018;46(W1):W537–W544.
17. Grüning B, Chilton J, Köster J, Dale R, Soranzo N, van den Beek M, et al. Practical computational reproducibility in the life sciences. Cell systems 2018;6(6):631–635.
18. Grüning B, Dale R, Sjödin A, Chapman BA, Rowe J, Tomkins-Tinch CH, et al. Bioconda: sustainable and comprehensive software distribution for the life sciences. Nat Methods 2018;15(7):475.
19. Batut B, Hiltemann S, Bagnacani A, Baker D, Bhardwaj V, Blank C, et al., List of Galaxy Training Network Releases;. Accessed: 2020-01-10. https://github.com/galaxyproject/training-material/releases.
20. Batut B, Hiltemann S, Bagnacani A, Baker D, Bhardwaj V, Blank C, et al. Community-Driven Data Analysis Training for Biology. Cell Systems 2018;6(6):752 – 758.e1. http://www.sciencedirect.com/science/article/pii/S2405471218302308.
21. Wang X, Yao H, Zhang Q, Ren X, Zhang Z. Direct Comparative Analysis of 10X Genomics Chromium and Smart-seq2. bioRxiv 2019;p. 615013.
22. Smith T, Heger A, Sudbery I. UMI-tools: modeling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy. Genome research 2017;27(3):491–499.
23. Zheng GX, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun 2017;8(1):1–12.
24. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, et al. STAR: ultrafast universal RNA-seq aligner. Bioinformatics 2013;29(1):15–21.
25. Dobin A, STARsolo Release Page;. Accessed: 2020-01-10. https://github.com/alexdobin/STAR/blob/master/docs/STARsolo.md.
26. Srivastava A, Malik L, Smith T, Sudbery I, Patro R. Alevin efficiently estimates accurate gene abundances from dscRNA-seq data. Genome Biol 2019 Mar;20(1). http://dx.doi.org/10.1186/s13059-019-1670-y.
27. Patro R, Duggal G, Love MI, Irizarry RA, Kingsford C. Salmon provides fast and bias-aware quantification of transcript expression. Nat Methods 2017 Mar;14(4):417–419. http://dx.doi.org/10.1038/nmeth.4197.
28. Tian L, Su S, Dong X, Amann-Zalcenstein D, Biben C, Seidi A, et al. scPipe: A flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data. PLOS Comput Biol 2018 Aug;14(8):e1006361. http://dx.doi.org/10.1371/journal.pcbi.1006361.
29. Hashimshony T, Senderovich N, Avital G, Klochendler A, de Leeuw Y, Anavy L, et al. CEL-Seq2: sensitive

highly-multiplexed single-cell RNA-Seq. Genome Biol 2016;17(1):77.
30. Etherington GJ, Soranzo N, Mohammed S, Haerty W, Davey RP, Di Palma F. A Galaxy-based training resource for single-cell RNA-sequencing quality control and analyses. GigaScience 2019;8(12):giz144.
31. Lun AT, Bach K, Marioni JC. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. Genome Biol 2016;17(1):75.
32. Wolf APRS F, ScanPy Preprocessing and Clustering 3k PBMCs Tutorial;. Accessed: 2020-01-10. https://scanpy-tutorials.readthedocs.io/en/latest/pbmc3k.html.
33. Raj A, van Oudenaarden A. Nature, nurture, or chance: stochastic gene expression and its consequences. Cell 2008;135(2):216–226.
34. Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment 2008;2008(10):P10008.
35. Tekman M, Accompanying Slide Deck for ScanPy PBMC Workflow;. Accessed: 2020-01-10. https://training.galaxyproject.org/training-material/topics/transcriptomics/tutorials/scrna-scanpy-pbmc3k/slides.html.
36. Maaten Lvd, Hinton G. Visualizing data using t-SNE. Journal of machine learning research 2008;9(Nov):2579–2605.
37. McInnes L, Healy J, Melville J. Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:180203426 2018;.
38. Megill C, Weaver C, Martin B, Kiggins J, Badajoz S, Bell S, et al., chanzuckerberg/cellxgene: Release 0.11.2; 2019. https://doi.org/10.5281/zenodo.3368662.
39. Wolf FA, Hamey FK, Plass M, Solana J, Dahlin JS, Göttgens B, et al. PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. Genome Biol 2019;20(1):59.
40. Herman JS, Grün D, et al. FateID infers cell fate bias in multipotent progenitors from single-cell RNA-seq data. Nat Methods 2018;15(5):379.
41. Regev A, Teichmann SA, Lander ES, Amit I, Benoist C, Birney E, et al. Science Forum: The Human Cell Atlas. Elife 2017;6:e27041.
42. Moreno P, Huang N, Manning JR, Mohammed S, Solovyev A, Polanski K, et al. User-friendly, scalable tools and workflows for single-cell analysis. bioRxiv 2020;.
43. Cao J, Spielmann M, Qiu X, Huang X, Ibrahim DM, Hill AJ, et al. The single-cell transcriptional landscape of mammalian organogenesis. Nature 2019;566(7745):496–502.
44. Kiselev VY, Yiu A, Hemberg M. scmap: projection of single-cell RNA-seq data across data sets. Nat Methods 2018 05;15(5):359–362.
45. Alquicira-Hernandez J, Sathe A, Ji HP, Nguyen Q, Powell JE. scPred: accurate supervised method for cell-type classification from single-cell RNA-seq data. Genome Biol 2019 12;20(1):264.
46. Pliner HA, Shendure J, Trapnell C. Supervised classification enables rapid annotation of cell atlases. Nat Methods 2019 10;16(10):983–986.
47. Kluyver T, Ragan-Kelley B, Pérez F, Granger BE, Bussonnier M, Frederic J, et al. Jupyter Notebooks-a publishing format for reproducible computational workflows. In: ELPUB; 2016. p. 87–90.
48. Allaire J. RStudio: integrated development environment for R. Boston, MA 2012;770.
49. Miller C, Qiao Y, DiSera T, D'Astous B, Marth G. Bam. Iobio: a Web-based, real-time, sequence alignment file inspector. Nat Methods 2014 11;11:1189.
50. Tang A, EtherCalc Github Repository;. Accessed: 2020-01-10. https://github.com/audreyt/ethercalc.
51. Bik HM, Interactive P. Phinch: an interactive, exploratory data visualization framework for -Omic datasets. bioRxiv 2014;p. 009944.
52. Kass JM, Vilela B, Aiello-Lammens ME, Muscarella R, Merow C, Anderson RP. Wallace: A flexible platform for reproducible modeling of species niches and distributions built for community expansion. Methods Ecol Evol 2018;9(4):1151–1156.
53. Schultheis H, Kuenne C, Preussner J, Wiegandt R, Fust A, Bentsen M, et al. WIlsON: Web-based Interactive Omics VisualizatioN. Bioinformatics 2019;35(6):1055–1057.
54. Schmunk RB. Panoply netcdf, hdf and grib data viewer. National Aeronautics and Space Administration-Goddard Institute for Space Studies 2018;.
55. Kerpedjiev P, Abdennur N, Lekschas F, McCallum C, Dinkla K, Strobelt H, et al. HiGlass: web-based visual exploration and analysis of genome interaction maps. Genome Biol 2018;19(1):1–12.
56. Fourdan O. Xfce: A Lightweight Desktop Environment. In: Annual Linux Showcase & Conference; 2000. .



1

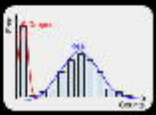
2

	C1	C2	C3	C4	C5
O1	1	0	4	2	9
O2	5	0	8	2	3
O3	0	0	0	0	0
O4	3	0	3	0	1

3

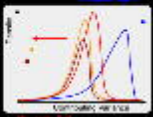
4

5



6

7



Galaxy / Single Cell Omics

Analyze Data | Workflow | Visualize | Shared Data | Help | Login or Register | Using 0 bytes

Tools

search tools

Annotation
Mapping
RNA-Seq
Single-cell

Requires the RDS output from the normalisation stage

Clustering

Distance Metric

Pearson

Clustering method

History

search datasets

RaceID WF

52.08 MB

33: Lineage Branch Analysis using Stemi D on data 28: Log

Galaxy / Single Cell Omics

Analyze Data | Workflow | Visualize | Shared Data | Help | Login or Register | Using 0 bytes

Tools

search tools

Get Data
Collection Operations

GENERAL TEXT TOOL

Text Manipulation
Filter and Sort
Convert Formats
FASTQ Quality Co
BED

COMMON GENOMIC

Operate on Geno
Annotation
Mapping
RNA-Seq
Single-cell
Picard
deepTools
FASTA/FASTQ ma
Workflows

All workflows

Welcome to the world of Single Cell Omics

The Single Cell Omics workbench is a comprehensive set of analysis tools and consolidated workflows. The workbench is based on the Galaxy framework, which guarantees simple access, easy extension, flexible adaption to personal and security needs, and sophisticated analyses independent of command-line knowledge.

The current implementation comprises more than 20 bioinformatics tools dedicated to different research areas of single cell biology.

This service is a joint project between different groups from the Earlham Institute, the EMBL-EBI, EMBL the Sorbonne University, Peter MacCallum

Galaxy / Single Cell Omics

Analyze Data | Workflow | Visualize | Shared Data | Help | Login or Register | Using 0 bytes

Tools

search tools

Get Data
Send Data
Collection Operations
Lift-Over
Text Manipulation
Convert Formats
Filter and Sort
Join, Subtract and Group
Fetch Alignments/Sequences
Operate on Genomic Intervals
Statistics
Graph/Display Data
Phenotype Association

Workflows

All workflows

CATEGORICAL METADATA

louvain

SELECTED GENES

Add

DIFFERENTIALLY EXPRESSED GENES

1: 0 cells | 2: 0 cells

Compute Differential Expression

CONTINUOUS METADATA

50,000 100,000 150,000 200,000 250,000

n_counts_all

3,000 4,000 5,000 6,000 7,000 8,000

n_counts

cellxgene: file

reset

History

search datasets

Anndata

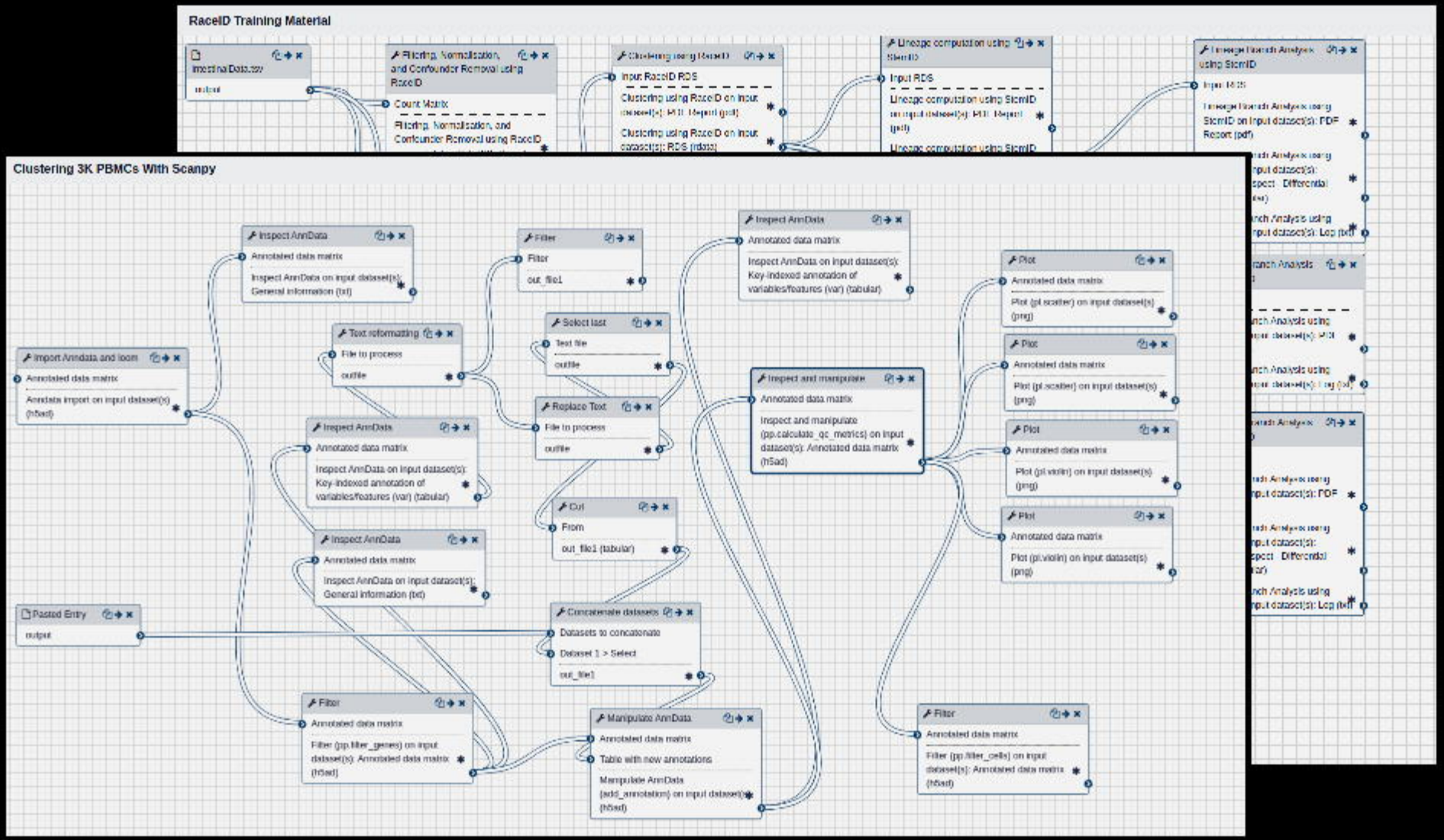
28.99 MB

2: retina.h5ad

5.5 MB

format: h5ad, database: ?

1: pbmc3k.h5ad


RaceID Training Material
Clustering 3K PBMCs With Scanpy
Import Anndata and loom
Annotated data matrix
Anndata import on input dataset(s) (h5ad)
Inspect AnnData
Annotated data matrix
Inspect AnnData on input dataset(s): General information (txt)
Text reformatting
File to process
outfile
Inspect AnnData
Annotated data matrix
Inspect AnnData on input dataset(s): Key-indexed annotation of variables/features (var) (tabular)
Inspect AnnData
Annotated data matrix
Inspect AnnData on input dataset(s): General information (txt)
Pasted Entry
output
Filter
Annotated data matrix
Filter (pp.filter_genes) on input dataset(s): Annotated data matrix (h5ad)
Filter
Filter
out_file1
Select last
Text file
outfile
Replace Text
File to process
outfile
Cut
From
out_file1 (tabular)
Concatenate datasets
Datasets to concatenate
Dataset 1 > Select
out_file1
Manipulate AnnData
Annotated data matrix
Table with new annotations
Manipulate AnnData (add_annotation) on input dataset(s) (h5ad)
Inspect AnnData
Annotated data matrix
Inspect AnnData on input dataset(s): Key-indexed annotation of variables/features (var) (tabular)
Inspect and manipulate
Annotated data matrix
Inspect and manipulate (pp.calculate_qc_metrics) on input dataset(s): Annotated data matrix (h5ad)
Filter
Annotated data matrix
Filter (pp.filter_cells) on input dataset(s): Annotated data matrix (h5ad)
Plot
Annotated data matrix
Plot (pl.scatter) on input dataset(s) (png)
Plot
Annotated data matrix
Plot (pl.scatter) on input dataset(s) (png)
Plot
Annotated data matrix
Plot (pl.violin) on input dataset(s) (png)
Plot
Annotated data matrix
Plot (pl.violin) on input dataset(s) (png)