Published in final edited form as:

Nat Biotechnol. 2012 November ; 30(11): 1036–1039. doi:10.1038/nbt.2404.

NGS analyses by visualization with Trackster

Jeremy Goecks¹, Nate Coraor², The Galaxy Team³, Anton Nekrutenko², and James Taylor¹

¹Departments of Biology and Mathematics & Computer Science, Emory University, Atlanta GA 30322

²Center for Comparative Genomics and Bioinformatics, Penn State University, University Park PA 16802

To The Editor

Visualization is an essential scientific tool, making it possible to view large amounts of data simultaneously, identify patterns and outliers within data, and communicate findings to others. Data analysis and visualization have traditionally been separate: data is first analyzed, and only then visualization is used to present a graphical overview of the results. This approach breaks down for large genomic datasets where visualizing the results of time-consuming, computationally intensive analyses often shows that different analysis settings need to be used. Repeatedly running large analyses and visualizing results is a wasteful and slow way to find good analysis settings. Visualization, then, cannot remain an endpoint for genomic analyses, but instead must be integrated with analysis tools so that visualization can be used to evaluate intermediate results and incrementally improve an analysis.

Here we introduce Trackster, a visual analysis environment for next-generation sequencing data that tightly couples interactive visualization with data analysis. Using Trackster, selected data subsets can be analyzed rather than complete datasets, thereby reducing analysis compute time from days to seconds. Trackster takes advantage of this dramatic reduction in analysis computing time to enable interactive, visual search of analysis settings. Using Trackster, many different analysis settings can be tried quickly and the outputs from different settings visualized together, making it easy to use visual inspection to select the settings that work best—all interactively and in minutes.

As next-generation sequencing tools have been widely adopted for different uses in the life sciences, investigators are spending progressively more time designing and adapting pipelines to meet their needs. One particularly challenging task in the analysis of next-generation sequencing data concerns choosing the right parameters; even small changes in parameter settings can produce a remarkably different result. Trying to understand how changing analysis settings will affect the final result requires a difficult cycle of repeatedly tweaking settings, re-running the analysis many times, and comparing outputs to look for differences. To make things worse, most of today's sequencing analysis tools require multiple parameter settings that change depending on experimental design, type of input data, and many other factors. As a result, it is often impossible to determine a "correct" list of settings a priori without exploring a tool's space of possible parameter values first. Exploring a tool's parameter space can become quite costly as the size of the sequencing datasets is very large and continues to grow. Consider one of the most popular applications of today's high throughput sequencing approaches, RNA-seq¹. A popular approach to analyze RNA-seq data is to use TopHat/Cufflinks package to assemble and quantify

Correspondence should be addressed to J.T. (james.taylor@emory.edu) and A.N. (anton@bx.psu.edu).

³<http://galaxyproject.org>

transcripts². In this approach, the quality of final assembly depends to a large degree on parameter settings chosen. Yet running the analysis requires many hours for most datasets, and hence a simple exploration of parameter settings can take days if not weeks.

Parameter space exploration and similar activities that require iterative analysis and data assessment suggest a new role for visualization. Rather than being an endpoint for genomic analyses, visualization should be integrated with analysis tools to enable investigators to use visualization as they work through an analysis. Integrating visualization tools with biological databases and analysis tools is a growing trend that includes genomic visualizations³. Genomic data visualization builds on the concept of a genome browser pioneered by Artemis⁴, popularized by UCSC Genome Browser⁵, and further extended by dozens of visualization applications and frameworks that are currently in active use or development. The IGV genome browser⁶ includes simple tools for dynamically filtering data, and the Savant browser⁷ includes a framework for extending the browser to perform new visualizations and analyses. In visualization research, approaches for interacting with tool parameter spaces in applications such as computer animation⁸ and image analysis⁹ have been developed. These approaches use visualization as the foundation and build analysis tools on top of this foundation.

An alternative approach is to use an analysis platform as the foundation and build visualization on top of this framework. We have taken this approach by building the Trackster visual analysis environment into the Galaxy tool-integration and analysis platform (<http://usegalaxy.org>)^{10, 11}. Trackster allows dynamic integration of tools incorporated into the Galaxy framework, including many popular tools used for high-throughput analysis. When visualizing tool output in Trackster, investigators can open the tool, modify its parameter settings, and the visualization will immediately update to reflect the new settings. Trackster also provides a parameter space view, where tool parameter spaces can be created and parameter sweeps can be used to systematically sample and visualize output from different parameter settings. This tight coupling of tool settings and visualization enables powerful interactions such as rapid tool parameter space exploration and dynamic data filtering to selectively show only pertinent data. Trackster enables interactive computing by running tools only on selected data, ensuring tool run time is short even for very large high-throughput sequencing datasets. Trackster is entirely Web-based, requiring only a modern Web browser to use all its features.

Galaxy provides a tool integration framework that allows nearly any tool (written in any language) to be integrated into Galaxy. Hundreds of tools have already been integrated in Galaxy, and there is an active community of developers integrating tools into Galaxy (<http://usegalaxy.org/community>). By leveraging the Galaxy tool collection, a variety of tools—including those for interval operations, transcriptome assembly, and SNP finding—can already be used for visual analysis in Trackster. Additional tools can easily be integrated into Trackster through the Galaxy framework as well. Sharing tools between Trackster and Galaxy enables investigators to easily transition between experimentation and mature analyses. After an investigator has used Trackster to experiment and find good tool settings, a single button is used to run the tool with the chosen settings on the complete dataset and its output placed in Galaxy's analysis workspace for further usage. Collaborative visual analysis is possible via shared Trackster visualizations. Shared visualizations are accessible using only a Web browser and provide complete access to all data and to integrated tools. Collaborators can copy a shared visualization, extend it, and reshare the new visualization. In this way, a group can pass around and modify Trackster visualizations to build collective visualizations of group knowledge.

To demonstrate how Trackster can be used for high-throughput sequencing analyses, we have utilized recently generated transcriptome sequencing data to study the expression dynamics of the human XBP1 gene. The unusual features of this locus make it a challenging target for RNA-seq experiments requiring careful parameter calibration. The gene's product—XBP1 transcription factor—is the key effector of eukaryotic unfolded protein response^{12, 13}. It is regulated in a very unusual way through a highly specific cleavage by an endoplasmic reticulum-bound endonuclease. The endonuclease, IRE1, removes a 26 bp spacer from the interior of XBP1 transcript leading to a reading frame-shift (since 26 is not divisible by 3)^{14, 15}. The cleaved mRNA is called XBP1s (where 's' stands for spliced; the unmodified XBP1 mRNA is commonly referred to as XBP1u, where 'u' implies 'unspliced') and is translated into a potent transcription factor activating genes involved in the unfolded protein response. Finally, translation of XBP1u has been shown to pause to retain the transcript in the vicinity of the ER membrane, ensuring that during the unfolded protein response, IRE1 can immediately splice XBP1u to produce XBP1s¹⁶. This pausing allowed us to exercise our approach on a new type of transcriptional data produced from ribosome-profiling experiments allowing precise sequencing of mRNA sites covered by assembled ribosomes¹⁷.

In this experiment, we addressed three questions related to XBP1 expression. First, is the current RNA-seq framework adequate for the detection of "introns" that are only 26 bp long and are not removed by the standard mRNA splicing mechanism. Second, what are the relative amounts of XBP1s and XBP1u mRNAs in various tissues. Third, what can be learned about the translational pausing of human XBP1u.

To address the first two questions, we have used RNA-seq BodyMap 2.0 data generated by Illumina, which provides roughly 160 million reads for each of the 16 human tissues sampled in this study. Using Galaxy, we mapped the reads from each of the 16 tissues against the human reference genome (version hg19) and performed initial transcript assemblies with Cufflinks (see <http://usegalaxy.org/interactive-mnaseq> for exact workflows with all data and parameter settings). Next, using Trackster, Cufflinks was rerun many times for reads mapping to the XBP1 locus to identify a set of parameters that produced the best assemblies across different tissues and allow us to identify the 26-bp intron. Defining suitable parameters was challenging because the two isoforms (XBP1s and XBP1u) are quite similar and expressed at very low levels in some tissues. Fig. 1 shows that by focusing Trackster on the XBP1 locus, we were able to adjust Cufflinks parameters embedded in the track view multiple times to achieve a rough rendering of the isoforms (Fig 1, top) and then use the parameter space view to find precise settings that produced the best rendering (Fig 1, bottom). After finding suitable assembly parameters, a Galaxy workflow was used to map reads, assemble transcripts, and quantify isoform expression for all tissues (Supplementary Table 1).

When transcript assembly and quantification was complete for all tissues, we used Trackster to visualize and filter isoforms. Trackster's dynamic filtering was used to develop filtering criteria to select for high-quality transcripts. Fig. 2 shows how dynamic filtering can be used to simultaneously filter transcript assemblies in multiple tissues. Dynamic filters were used to first find assembly artifacts (Fig. 2A–B) and then to develop criteria to remove them (Fig. 2C).

Using Trackster to assemble and filter transcripts related to XBP1 locus saved many hours of computing time. Assembling transcripts genome-wide takes ~3–8 hours, and filtering filtering genome wide takes ~10–20 minutes. In Trackster, transcript assembly for XBP1 locus can be done in 1–2 minutes and filtering can be done in real time. Hence, hundreds of hours of compute time were saved by using Trackster. Trackster also reduced investigator

time spent on this analysis. When running tools or filtering, the settings and corresponding outputs could be viewed simultaneously rather than switching between the tool and the visualization.

To address the third question related to ribosome pausing while translating XBP1u mRNA, we used data produced by Reid and Niccitta¹⁸. This data includes ribosome footprinting profiles for cytosolic and ER fractions of human embryonic kidney cells. To analyze these data, we mapped ribosomal footprinting reads to the merged set of assembled transcripts for all tissues and visualized the result. Visualization was done by creating a custom genome browser in Trackster for the XBP1u transcript (Fig. S1). One can see that the ribosomal pausing site is clearly identified by piling of reads, and this phenomenon is only observed for reads originating from ER-associated cell fraction.

A Galaxy Page describing details of our analysis of XBP1 isoforms, including datasets, analysis histories, workflows, and Tracker visualizations used, is available here: <http://usegalaxy.org/interactive-rnaseq>. Each Trackster visualization is fully functional: anyone can use a visualization to explore the data and perform visual analysis by running tools. Investigators viewing a shared visualization can create a copy of the visualization, modify or extend it, and then share or publish the new visualization on the Web.

Trackster's use of data subsets to reduce analysis compute time is applicable to a wide set of genomic tools. For instance, genomic interval operations (e.g. intersect, subtract), transcript assembly and quantification, and human variation analysis (e.g. SNP calling) are compatible with Trackster's analysis approach. However, tools (e.g. some peak callers) that use data from many or all genomic regions in order to build a global model require additional support to work with Trackster. These tools must be run once in full to generate the model, and then the model can be stored in Galaxy and reused in Trackster. Transcript quantification in Cufflinks benefits from a global model and Trackster makes use of it when it is available. Alternatively, a tool (e.g. a read mapper) may require all input data because it is not possible to identify, prior to runtime, a subset of input data needed to produce correct output in a particular genomic region. For such tools, dynamic filtering can be used to simulate running a tool using different parameters. In this approach, a tool's parameters are relaxed so that many potential outputs are produced and attribute values are attached to output data. Filtering can then be used to observe the data that would be produced for particular parameter values.

Visualization and data analysis tools are used in nearly all high-throughput sequencing experiments, yet too often they are not well integrated. Coupling visualization and analysis tools into a visual analysis environment where analysis output can be generated and visually assessed in real time is a powerful approach for computational science. Trackster provides an environment for interactive visual analysis that is widely applicable to many different high-throughput sequencing experiments. General visual analysis techniques that can be performed in Trackster include tool parameter space visualization and exploration, systematic sweeps of parameter values, and dynamic filtering. Trackster makes visual analysis possible for a wide variety of tools by leveraging the Galaxy framework, thereby tapping into the large collection of tools already integrated into Galaxy and providing a simple path for integrating additional tools into Trackster. This approach to tool integration enables popular, production-level tools, such as Cufflinks used in our example, to be integrated into Trackster without modification to the tools themselves. In our experiment, Trackster's visual analysis features made it possible to use interactive visualization to improve Cufflinks' transcript assemblies via parameter space exploration and to remove assembly artifacts using dynamic filtering. Trackster also supports collaborative visual

analysis via Web-based, fully-functional shared visualizations that can be modified, extended, reshared, and published.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

Efforts of the Galaxy Team (Enis Afgan, Dannon Baker, Dan Blankenberg, Nate Coraor, Jeremy Goecks, Greg Von Kuster, Ross Lazarus, Kanwei Li) were instrumental for making this work happen. This project was supported by American Recovery and Reinvestment Act (ARRA) funds through grant number HG005542 from the National Human Genome Research Institute, National Institutes of Health as well as grants HG005133, HG004909 and HG006620 and NSF grant DBI 0543285. Additional funding is provided, in part, under a grant with the Pennsylvania Department of Health using Tobacco Settlement Funds. The Department specifically disclaims responsibility for any analyses, interpretations or conclusions.

References

1. Trapnell C, et al. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat Biotech.* 2010; 28:511–515.
2. Trapnell C, et al. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat Protoc.* 2012; 7:562–578. [PubMed: 22383036]
3. Nielsen CB, Cantor M, Dubchak I, Gordon D, Wang T. Visualizing genomes: techniques and challenges. *Nature Methods.* 2010; 7:S5-S15–S15-S15. [PubMed: 20195257]
4. Rutherford K, et al. Artemis: sequence visualization and annotation. *Bioinformatics.* 2000; 16:944–945. [PubMed: 11120685]
5. Kent WJ. The human genome browser at UCSC. *Genome Res.* 2002; 12:996–1006. [PubMed: 12045153]
6. Robinson JT, et al. Integrative genomics viewer. *Nat Biotech.* 2011; 29:24–26.
7. Fiume M, Williams V, Brook A, Brudno M. Savant: genome browser for high-throughput sequencing data. *Bioinformatics.* 2010; 26:1938–1944. [PubMed: 20562449]
8. Jankun-Kelly TJ, Kwan-Liu M. Visualization exploration and encapsulation via a spreadsheet-like interface. *Visualization and Computer Graphics, IEEE Transactions on.* 2001; 7:275–287.
9. Pretorius AJ, Bray MAP, Carpenter AE, Ruddle RA. Visualization of Parameter Space for Image Analysis. *Visualization and Computer Graphics, IEEE Transactions on.* 2011; 17:2402–2411.
10. Goecks J, Nekrutenko A, Taylor J. Galaxy Team T. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology.* 2010; 11:R86–R86. [PubMed: 20738864]
11. Blankenberg, D., et al. Galaxy: a web-based genome analysis tool for experimentalists. In: Ausubel, Frederick M., et al., editors. *Current Protocols in Molecular Biology.* Vol. Chapter 19. 2010.
12. Ron D, Walter P. Signal integration in the endoplasmic reticulum unfolded protein response. *Nat Rev Mol Cell Biol.* 2007; 8:519–529. [PubMed: 17565364]
13. Walter P, Ron D. The unfolded protein response: from stress pathway to homeostatic regulation. *Science.* 2011; 334:1081–1086. [PubMed: 22116877]
14. Mori K. Signalling pathways in the unfolded protein response: development from yeast to mammals. *J Biochem.* 2009; 146:743–750. [PubMed: 19861400]
15. Calton M, et al. IRE1 couples endoplasmic reticulum load to secretory capacity by processing the XBP-1 mRNA. *Nature.* 2002; 415:92–96. [PubMed: 11780124]
16. Yanagitani K, Kimata Y, Kadokura H, Kohno K. Translational pausing ensures membrane targeting and cytoplasmic splicing of XBP1u mRNA. *Science.* 2011; 331:586–589. [PubMed: 21233347]
17. Guo H, Ingolia NT, Weissman JS, Bartel DP. Mammalian microRNAs predominantly act to decrease target mRNA levels. *Nature.* 2010; 466:835–840. [PubMed: 20703300]

18. Reid DW, Nicchitta CV. Genome-scale ribosome footprinting identifies a primary role for endoplasmic reticulum-bound ribosomes in the translation of the mRNA transcriptome. *J Biol Chem*. 2011

Figure 1.

Using Trackster to improve *XBP1* transcript assembly results via interactive tool use and parameter space exploration. **Top:** Interactive tool use in track view: (a) Reference gene annotation for *XBP1* locus; (b) mapped RNA-seq reads to be assembled into transcripts; (c) embedded tool interface for Cufflinks, a transcript assembly tool for high-throughput sequencing data, enabling parameter settings to be quickly changed and Cufflinks rerun on data in the visible region to produce a new assembly; (d) initial assembly using default settings; (e), (f), and (g) repeatedly changing settings and rerunning Cufflinks produces new assemblies that are visualized automatically and can be easily compared. **Bottom:** Tool parameter space exploration: (i) Galaxy's Cufflinks tool form is used to create and modify a

partial tool parameter space tree by setting parameters' minimum, maximum, and number of samples; (ii) partial Cufflinks parameter tree is displayed, and clicking on a node in the tree runs Cufflinks using all combinations of settings defined by the node's subtree; (iii) Cufflinks output in the *XBPI* region is visualized automatically and can be easily compared. In both Trackster views, Cufflinks is run using only data from the visible or selected region, so each assembly can be created in about a minute. Once good parameter settings have been found, Cufflinks can be run on the complete dataset using the good settings.

Figure 2.

Using Trackster to dynamically filtering transcript assemblies for multiple tissues to identify high-quality transcripts and filter out assembly artifacts. Trackster automatically creates filters for attributes values for any genomic interval dataset, including assembled transcripts. Transcripts that are likely to be assembly artifacts are those with a low Score—a relative measure of transcript expression—or those with a low FPKM—an absolute measure of transcript expression. (a) To compare transcript FPKM values, FPKM is encoded as transparency, making transcripts with high expression darker and easy to see; (b) alternatively, Score is encoded as height and a filter is used to hide low scoring transcripts, so transcript height denotes its Score relative to the filter's range (224–1000) and transcripts with a high score can be picked out easily; (c) filtering simultaneously by FPKM and Score yields a set of high-quality transcripts.