Using Galaxy to Perform Large-Scale Interactive Data Analyses

Jennifer Hillman-Jackson,[1] Dave Clements,[2] Daniel Blankenberg,[1] James Taylor,[2] Anton Nekrutenko,[1] and Galaxy Team[1,2]

[1]Penn State University, University Park, Pennsylvania
[2]Emory University, Atlanta, Georgia

ABSTRACT

Innovations in biomedical research technologies continue to provide experimental biologists with novel and increasingly large genomic and high-throughput data resources to be analyzed. As creating and obtaining data has become easier, the key decision faced by many researchers is a practical one: where and how should an analysis be performed? Datasets are large and analysis tool set-up and use is riddled with complexities outside of the scope of core research activities. The authors believe that Galaxy provides a powerful solution that simplifies data acquisition and analysis in an intuitive Web application, granting all researchers access to key informatics tools previously only available to computational specialists working in Unix-based environments. We will demonstrate through a series of biomedically relevant protocols how Galaxy specifically brings together (1) data retrieval from public and private sources, for example, UCSC's Eukaryote and Microbial Genome Browsers, (2) custom tools (wrapped Unix functions, format standardization/conversions, interval operations), and 3rd-party analysis tools. *Curr. Protoc. Bioinform.* 38:10.5.1-10.5.47. © 2012 by John Wiley & Sons, Inc.

Keywords: Galaxy • comparative genomics • genomic alignments • Web application • genome variation

INTRODUCTION

Most experimental biologists cannot fully take advantage of genomic data due to a formidable wall of countless and unnecessary computational issues. The goal of Galaxy (Blankenberg et al., 2010; Goecks et al., 2010) is to solve these issues. Consider the following example: a researcher wants to identify protein-coding exons containing the highest density of SNPs. Most biologists know three primary sources of genome-wide data for vertebrates: Entrez at the National Center for Biotechnology Information (NCBI; *UNIT 1.3*; Maglott et al., 2005), the Genome Browser at the University of California at Santa Cruz (*UNIT 1.4*; Karolchik et al., 2003; Schneider et al., 2006; Rosenbloom et al., 2009), and Ensembl (*UNIT 1.15*; Birney et al., 2004) at the EBI/Wellcome Trust Sanger Institute (U.K.). Although these three sources offer extensive information about genes, including genomic structure, gene expression profiles, and SNPs, the end user must still perform this task elsewhere—the listed resources do not provide functionality necessary to perform this analysis. Typically, the project ends up in the hands of a graduate student who might initially try to achieve the analysis using popular desktop applications. Unfortunately, Excel (like many other desktop applications) cannot handle that much data. As a result, this relatively simple task becomes a complex endeavor that may easily take weeks or months. In the authors' view, this does not have to be complicated. Galaxy bridges the gap between data and analyses by allowing experimental biologists without programming experience to easily perform large-scale studies from within their Web browsers.

In this unit, the authors describe the functionality of Galaxy using a series of examples that correspond to the following protocols. Basic Protocol 1 covers the most

fundamental features of Galaxy. Basic Protocol 2 elaborates on different types of data accepted by Galaxy. It also shows the user how to upload data and set data attributes. Basic Protocol 3 demonstrates analysis with ChIP-seq high throughput sequencing data. Basic Protocol 4 shows that manipulation of genomic intervals is one of Galaxy's greatest strengths. Basic Protocol 5 explains how Galaxy enables users to manipulate multiple alignments.

In addition, a fully interactive supplement titled "Using Galaxy to Perform Large-Scale Interactive Data Analysis: A live supplement" is available on the main public Galaxy instance under Shared Data: Published Pages: Using Galaxy 2012, at *http://usegalaxy.org/u/galaxyproject/p/using-galaxy-2012*. For each protocol, the input datasets, a complete history, and any workflows are included along with the exact methods and a screencast (video tutorial). These items can be examined, copied, rerun, and modified at the main public Galaxy instance (*http://usegalaxy.org*) and downloaded for use in a local or cloud instance (*http://getgalaxy.org*).

BASIC PROTOCOL 1

FINDING HUMAN CODING EXONS WITH HIGHEST SNP DENSITY

Suppose one wants to find the top hundred protein-coding exons in the human genome with the highest density of single nucleotide polymorphisms (SNPs). Answering this question is not trivial. To do so, one needs to compare all human exons to all human SNPs. To put this into perspective, the current version of the human genome at UCSC for hg19 includes over 350,000 known coding exons and dbSNP build 134 (Sherry et al., 2001) contains nearly 49 million SNPs. Galaxy is specifically designed to make such large-scale analyses fast and user-friendly. Galaxy's interface is accessible from *http://usegalaxy.org*. In the following protocol, the authors will use RefSeq (Karolchik et al., 2004; Pruitt et al., 2005) exons and dbSNP annotations on chromosome 22 extracted from the UCSC Table Browser (Fujita et al., 2011).

Necessary Resources

Hardware

An Internet-connected computer

Software

Internet browser that supports JavaScript (e.g., most current browsers such as Mozilla Firefox, Safari, Opera, Chrome, or Microsoft Internet Explorer)

Files

None

1. Open the Galaxy Project's homepage by pointing your Web browser to *http://galaxyproject.org.*

   The project homepage features four prominent sections: Use Galaxy, Get Galaxy, Learn Galaxy, and Get Involved.

2. Click on the Use Galaxy link (*http://usegalaxy.org*), which will bring up the free public Galaxy server.

   The Galaxy interface, populated with sample data, is shown in Figure 10.5.1.

3. Hover over User in the top bar, then click on Register in the submenu.

   The center panel of the Galaxy interface will change into a form asking you to provide user account details.

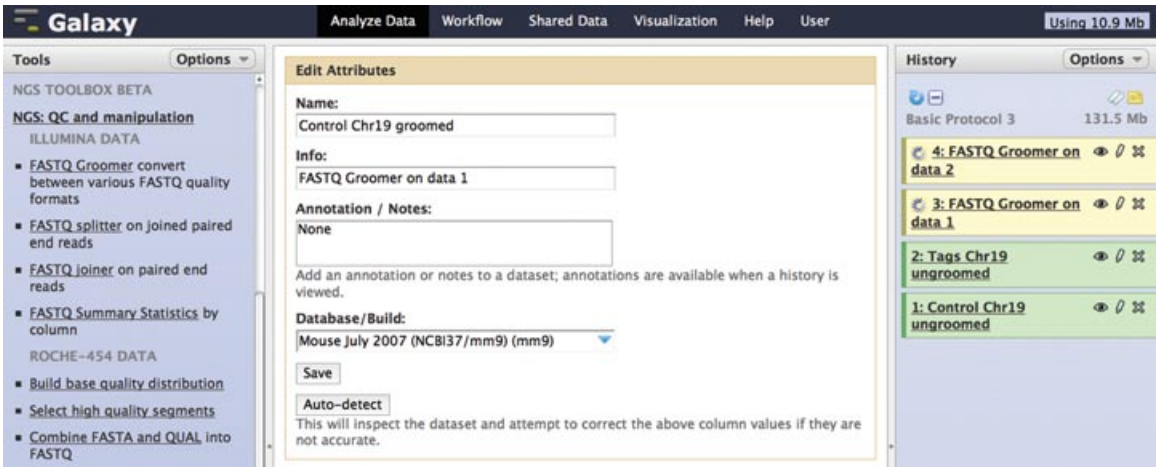


Figure 10.5.1 Galaxy interface contains four areas: the top bar, Tools panel (left column), detail panel (middle column), and History panel (right column). The top bar contains user account controls as well as help and contact links. The left panel lists the analysis tools and data sources available to the user. The middle panel displays interfaces for tools selected by the user. The right panel (the History panel) shows datasets and the results of analyses performed by the user. Pictured here are four history items in two different stages of completion: The two "FASTQ Groomer" items are yellow, meaning they are in progress, while the two "ungroomed" items are shown in green, meaning they have completed successfully. Every action by the user generates one or more new history items, which can then be used in subsequent analyses, downloaded, or visualized. For the color version of this figure go to *http://www.currentprotocols.com/protocol/bi1005*.

4. Fill in the Create Account form and click "submit."

   Although Galaxy can be used without creating an account, the authors highly recommend registering. First, having an account allows you to access your data from any machine connected to the Internet. Second, having an account safeguards data stored in the history against deletion. Anonymous histories and datasets are not reusable from one session to the next. You also cannot do all the protocols in this unit without an account.

   When registering for the account, note the mailing list subscription checkbox. By checking it, a new user will be subscribed to the "galaxy-announce" mailing list. This list is a moderated, low-volume list for announcements of interest to the Galaxy community.

5. After registering, you will be automatically logged in. For subsequent sessions, log in using your e-mail and password. Hover over User in the top menu and click Login in the submenu.

6. Name this history.

   a. Click on Unnamed History in the History panel.

   b. Enter Basic Protocol 1 and hit Return.

      You can only do this step if you are logged in.

7. Click the Get Data link at the top of the Tools panel.

8. Click the UCSC Main link.

   The UCSC Table Browser interface will appear in the middle panel of the Galaxy screen. The History panel on the right will disappear until you leave UCSC Main.

9. Import coordinates of protein-coding exons of known human genes from the UCSC Table Browser to Galaxy. Make sure the following parameters are set as shown in Figure 10.5.2A:

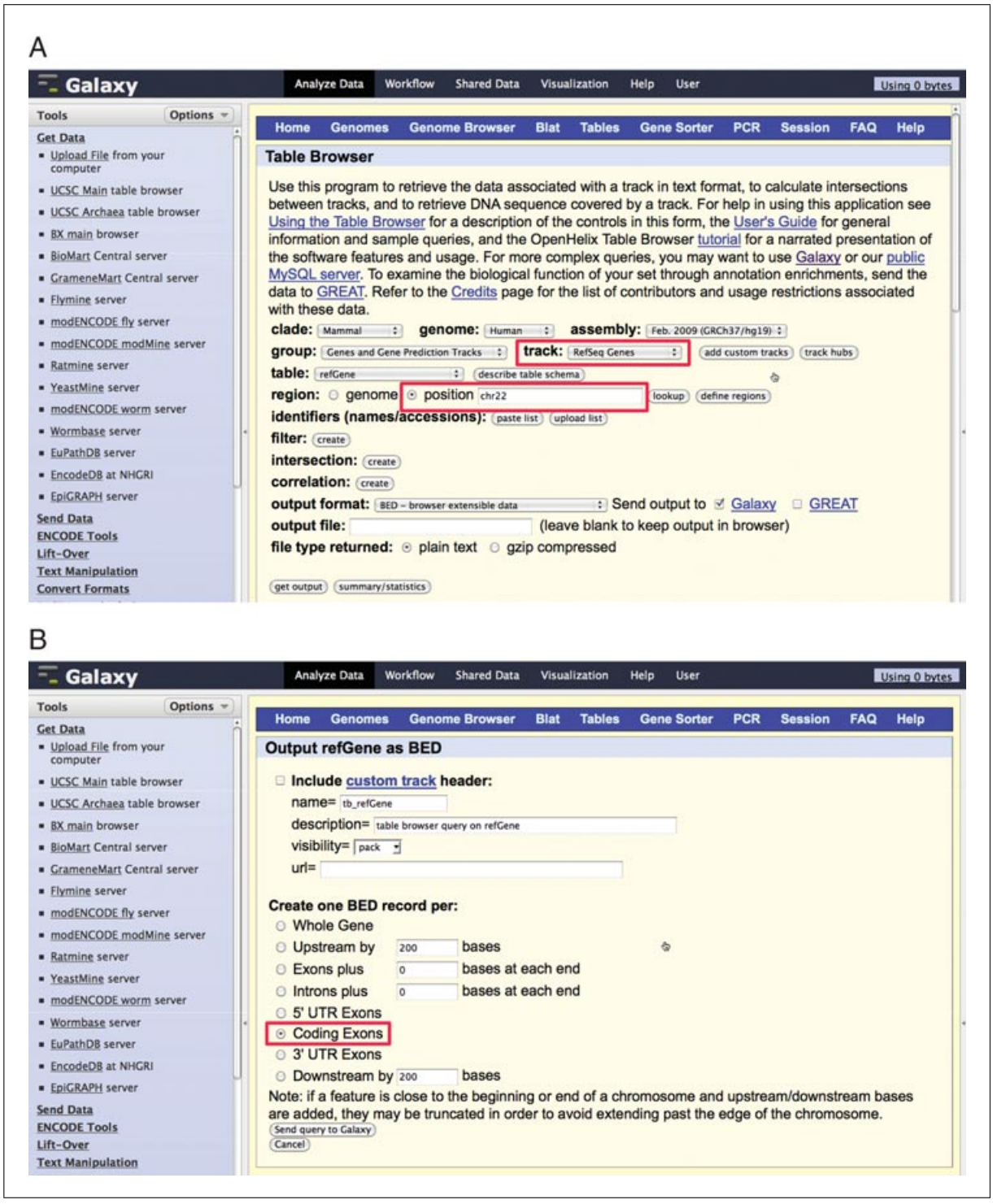


Figure 10.5.2 Uploading a list of protein-coding exons (in BED format) of known human genes from the UCSC Table browser involves two steps (**A** and **B**) described in the text.

clade: Mammal
genome: Human
assembly: Feb 2009 (GRCh37/hg19)
group: Genes and Gene Predictions Tracks
track: RefSeq Genes
region: position
position: chr22

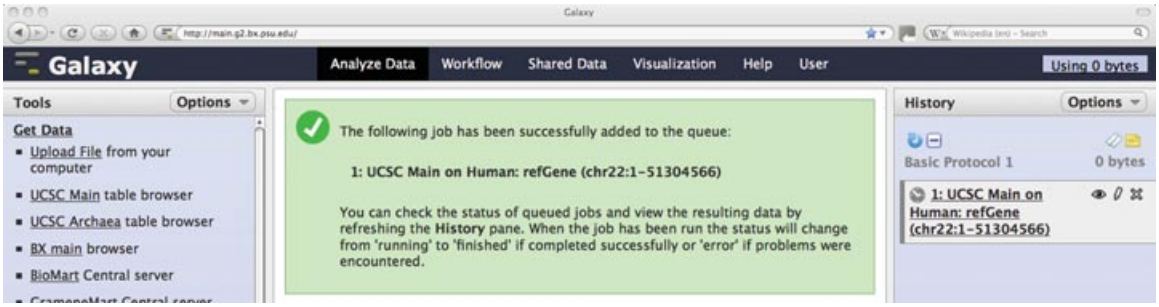


Figure 10.5.3 When a job is queued, a history item is initially gray. When a job is running, a history item is yellow. When a job is complete, a history item is green (successful) or red (error). For the color version of this figure go to *http://www.currentprotocols.com/protocol/bi1005*.

output format: BED – browser extensible data
Send output to: Galaxy

a. Click the "get output" button.

This brings up the next screen of the Table Browser interface as shown in Figure 10.5.2B.

b. Select the Coding Exons radio button.

c. Click the "Send query to Galaxy" button.

This will return you to Galaxy and create the first item called "UCSC Main on Human: refGene (chr22:1-51304566)" in your History panel, and place a large green box in the center panel showing that the upload has been successfully added to the Galaxy job queue. The history item is initially gray, showing it is queued (Fig. 10.5.3). The history item becomes yellow when the job is running, and green once it is complete. If a task fails for any reason the history item will turn red.

This dataset contains ~ 7,100 exons.

d. Click the dataset's name (underlined text, upper left corner) to expand the box.

Icons in the upper right corner (e.g., eye, pencil, and ×) as well as links in the expanded history item allow one to perform tasks described in the legend for Figure 10.5.4.

10. Rename the dataset to something more memorable.

a. In the history panel, click the pencil icon next to the "UCSC Main on Human: refGene (chr22:1-51304566)" dataset.

This opens the Edit Attributes panel in the center as shown in Figure 10.5.5.

b. Copy and paste the contents of the Name field into the Info field.

This is step is not necessary, but it does keep this somewhat useful information ("UCSC Main on Human: refGene (chr22:1-51304566)") associated with the dataset.

c. Type `Exons hg19 chr22` into the Name text box.

d. Click the Save button.

The item is now renamed in the History panel. To see the new Info value (which was the old name), click on the new name in the History panel.

Giving the dataset a meaningful name makes it easier to keep track of multiple datasets when working in large histories.

11. In the Tools panel, click Get Data, and then UCSC Main under it.

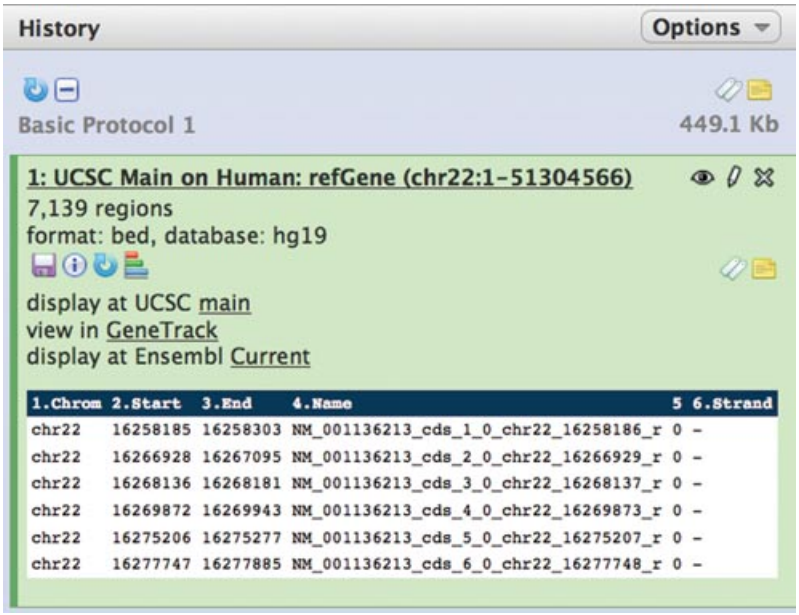


Figure 10.5.4 Close up of Galaxy history item. Clicking on links and icons triggers the following events: eye = shows a preview of the dataset in Galaxy's middle panel; pencil = open metadata editor. This brings up an interface in the middle panel of the Galaxy screen that allows one to edit the attributes of the current history item. For example, one may wish to give the history item a more descriptive name or change column assignments (see Basic Protocol 2); × = delete item from the history (To undelete or permanently delete, use the history's Options menu and select "View deleted datasets".); "save" = copy dataset to your computer; "i" = view details about this dataset in center panel, including the dataset(s), if any, that it was generated from; "rerun" = display this tool in center panel with the same settings it was run with, allowing this step to be exactly rerun or to be modified and rerun; "tags" = add free text tags to this dataset; "sticky note" = add free text annotation. Finally, if the dataset can be visualized in a browser, links to the Galaxy Track Browser (stacked bars icon) and to UCSC, GeneTrack, Ensembl, and others will also be displayed.

12. Import coordinates of SNPs from the UCSC Table Browser to Galaxy. Make sure the following parameters are set:

    clade: Mammal
    genome: Human
    assembly: Feb 2009 (GRCh37/hg19)
    group: Variation and Repeats
    track: Common SNPs(132)
    region: position
    position: chr22
    output format: BED – browser extensible data
    Send output to: Galaxy

    a. Click the "get output" button.

       This brings up the next screen of the Table Browser interface.

    b. Select the Whole Gene radio button.

    c. Click the "Send query to Galaxy" button.

       This will create the second history item named "UCSC Main on Human: snp132Common (chr22:1-51304566)". This dataset is much larger than the Exons dataset, with ~170,000 SNPs in it.

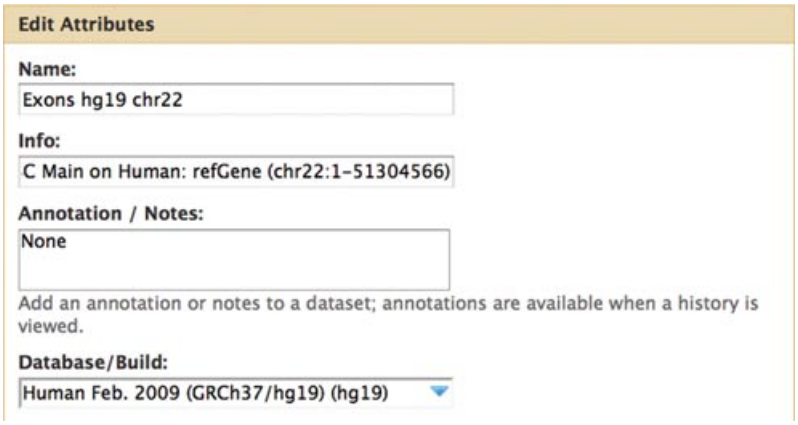


Figure 10.5.5 The “Edit Attributes" form in the center panel. Each attribute can be modified and saved. In this figure, the system-generated name has been copied to the Info field, and a short descriptive name entered in the Name field.

d. Rename the new dataset. Click on the new dataset’s pencil icon, copy the old name to the Info text box, and type `SNPs hg19 chr22` in the Name text box. Finish by clicking the Save button.

13. Click Operate on Genomic Intervals in the Tools panel.

14. Click Join to perform a Join operation.

a. Set:
Join: Exons hg19 chr22
with: SNPs hg19 chr22
with min overlap: 1
Return: Only records that are joined (INNER JOIN)

This will join any exon and SNP records that overlap by one or more base pairs. For explanation of various join options, see Basic Protocol 4.

b. Click Execute.

This will take a few minutes to compute.

The join tool allows the user to find intersections between two sets of genomic intervals. In our case, we are joining protein-coding exons and SNPs as shown in Figure 10.5.6A.

The result of this operation, a dataset with ~4,800 overlapping exon-SNP pairs, is shown in Figure 10.5.7. The first six columns represent protein-coding exons, while the last six represent SNPs. The six columns are: (1) chromosome, (2) start position, (3) end position, (4) description, (5) score (always 0 in this example), and (6) strand (+ or –). Figure 10.5.7 highlights a single exon (located on chromosome 22 between positions 17,264,508 and 17,265,299), which contains (overlaps with) 4 SNPs. One can see that coordinates of SNPs (columns eight and nine) are always within the start and end positions of the exon (columns two and three).

c. Rename the join dataset. Click on the new dataset’s pencil icon, copy the old name to the Info text box, and type `Exon-SNP Pairings` in the Name text box. Finish by clicking Save.

15. Click Statistics in the Tools panel.

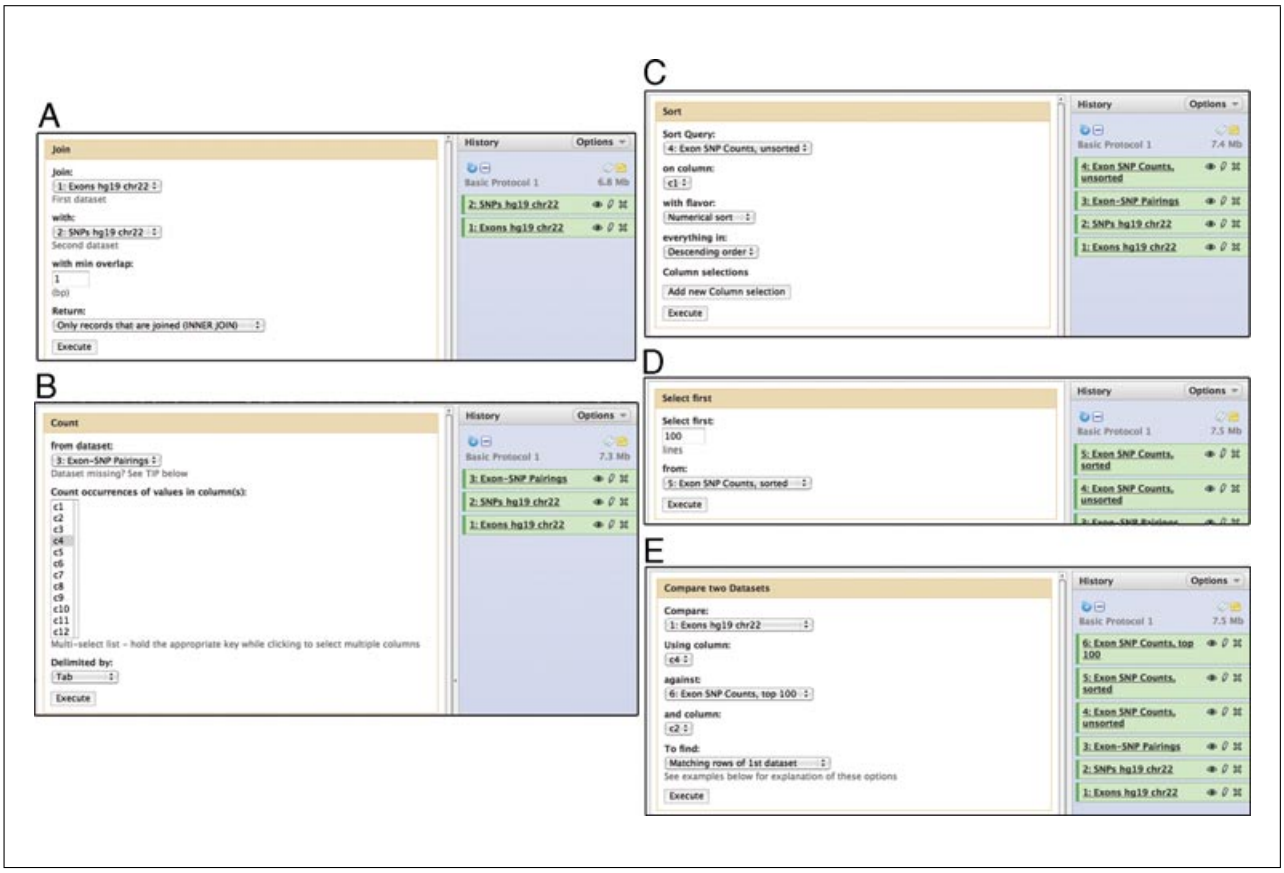


Figure 10.5.6 Data manipulation tools: Join (**A**), Count (**B**), Sort (**C**), Select first lines (**D**), and Compare two datasets (**E**).

16. Click Count to count the number of SNPs per exon as shown in Figure 10.5.6B.

    a. Set:
    from dataset: Exon-SNP Pairings
    Count occurrences of values in columns(s): c4
    delimited by: Tab

    Column 4 contains the exon name.

    b. Click Execute.

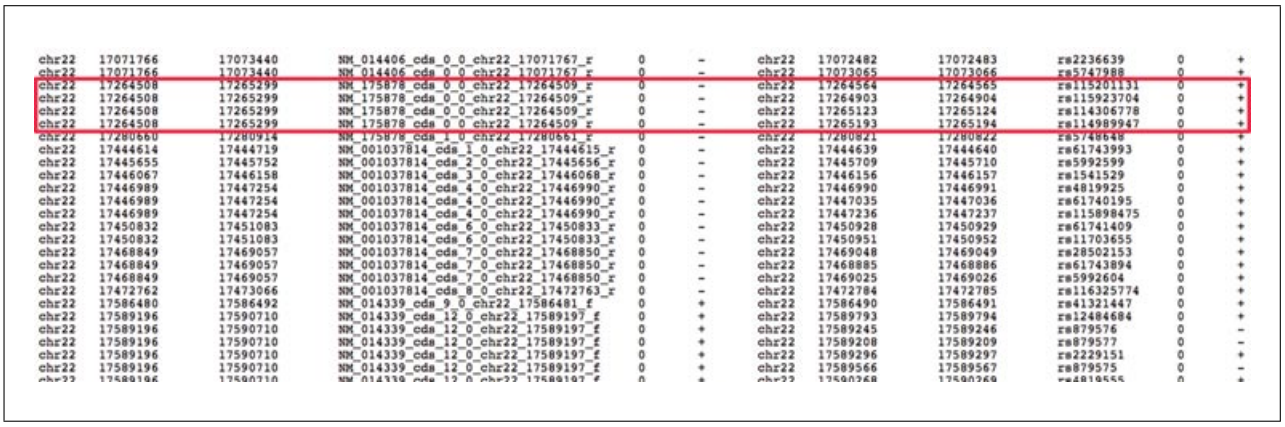
    Figure 10.5.7 shows that the number of times each exon is listed equals the number of SNPs that exon overlaps with. Thus, by counting the number of occurrences of every exon in this dataset, one can compute how many SNPs each exon overlaps with. The resulting dataset contains ~2,600 lines, one for each exon that overlaps with one or more SNPs.

    c. Rename the new dataset. Click the new dataset's pencil icon, set the Info text box to "Count on data 3; Count of unique values in c4" and type `Exon SNP Counts, unsorted` in the Name text box. Finish by clicking Save.

17. Sort results by the number of SNPs per exon as shown in Figure 10.5.6C.

    a. In the Tools panel, click Filter and Sort and then "Sort."

    b. Set:
    Sort Query: Exon SNP Counts, unsorted
    on column: c1
    with flavor: Numerical sort
    everything in: Descending order



chr22	17071766	17073440	NM_014406_cds_0_0_chr22_17071767_r	0	-	chr22	17072482	17072483	rs2236639	0	+
chr22	17071766	17073440	NM_014406_cds_0_0_chr22_17071767_r	0	-	chr22	17073065	17073066	rs5747988	0	+
chr22	17264508	17265299	NM_175878_cds_0_0_chr22_17264509_r	0	-	chr22	17264564	17264565	rs115201131	0	+
chr22	17264508	17265299	NM_175878_cds_0_0_chr22_17264509_r	0	-	chr22	17264903	17264904	rs115923704	0	+
chr22	17264508	17265299	NM_175878_cds_0_0_chr22_17264509_r	0	-	chr22	17265123	17265124	rs114306778	0	+
chr22	17264508	17265299	NM_175878_cds_0_0_chr22_17264509_r	0	-	chr22	17265193	17265194	rs114989947	0	+
chr22	17280660	17280914	NM_175878_cds_1_0_chr22_17280661_r	0	-	chr22	17280821	17280822	rs5748648	0	+
chr22	17444614	17444719	NM_001037814_cds_1_0_chr22_17444615_r	0	-	chr22	17444639	17444640	rs61743993	0	+
chr22	17445655	17445752	NM_001037814_cds_2_0_chr22_17445656_r	0	-	chr22	17445709	17445710	rs5992599	0	+
chr22	17446067	17446158	NM_001037814_cds_3_0_chr22_17446068_r	0	-	chr22	17446156	17446157	rs1541529	0	+
chr22	17446989	17447254	NM_001037814_cds_4_0_chr22_17446990_r	0	-	chr22	17446990	17446991	rs4819925	0	+
chr22	17446989	17447254	NM_001037814_cds_4_0_chr22_17446990_r	0	-	chr22	17447035	17447036	rs61740195	0	+
chr22	17446989	17447254	NM_001037814_cds_4_0_chr22_17446990_r	0	-	chr22	17447236	17447237	rs115898475	0	+
chr22	17450832	17451083	NM_001037814_cds_6_0_chr22_17450833_r	0	-	chr22	17450928	17450929	rs61741409	0	+
chr22	17450832	17451083	NM_001037814_cds_6_0_chr22_17450833_r	0	-	chr22	17450951	17450952	rs11703655	0	+
chr22	17468849	17469057	NM_001037814_cds_7_0_chr22_17468850_r	0	-	chr22	17469048	17469049	rs28502153	0	+
chr22	17468849	17469057	NM_001037814_cds_7_0_chr22_17468850_r	0	-	chr22	17468885	17468886	rs61743894	0	+
chr22	17468849	17469057	NM_001037814_cds_7_0_chr22_17468850_r	0	-	chr22	17469025	17469026	rs5992604	0	+
chr22	17472762	17473066	NM_001037814_cds_8_0_chr22_17472763_r	0	-	chr22	17472784	17472785	rs116325774	0	+
chr22	17586480	17586492	NM_014339_cds_9_0_chr22_17586481_f	0	+	chr22	17586490	17586491	rs41321447	0	+
chr22	17589196	17590710	NM_014339_cds_12_0_chr22_17589197_f	0	+	chr22	17589793	17589794	rs12484684	0	+
chr22	17589196	17590710	NM_014339_cds_12_0_chr22_17589197_f	0	+	chr22	17589245	17589246	rs879576	0	-
chr22	17589196	17590710	NM_014339_cds_12_0_chr22_17589197_f	0	+	chr22	17589208	17589209	rs879577	0	-
chr22	17589196	17590710	NM_014339_cds_12_0_chr22_17589197_f	0	+	chr22	17589296	17589297	rs2229151	0	+
chr22	17589196	17590710	NM_014339_cds_12_0_chr22_17589197_f	0	+	chr22	17589566	17589567	rs879575	0	-

Figure 10.5.7 Result of joining two interval datasets, highlighting a single exon that contains (overlaps with) 4 SNPs.

c. Click Execute.

The resulting history item contains the input dataset, sorted by the number of SNPs in each exon (column 1).

d. Rename the sorted dataset. Click the new dataset's pencil icon, copy the old name to the Info text box, and type `Exon SNP Counts, sorted` in the Name text box. Finish by clicking Save.

18. Select the top 100 exons from this list as shown in Figure 10.5.6D.

a. In the Tools panel click Text Manipulation and then "Select first".

b. Set:
Select first: 100
from: Exon SNP Counts, sorted

c. Click Execute.

After execution is finished your new history item will contain a list of the 100 exons with the highest SNP density.

d. Rename the sorted dataset. Click the new dataset's pencil icon, copy the old name to the Info text box, and type `Exon SNP Counts, top 100` in the Name text box. Finish by clicking Save.

The question asked by this protocol has now been answered: The last dataset lists only the exons on chromosome 22 with the most SNPs. However, we lost some information about those exons, such as coordinates and strand, in the process. The final step will link these data back into the result.

19. Retrieve the other information for the top 100 exons as shown in Figure 10.5.6E.

a. In the Tools panel, click "Join, Subtract, and Group" and then "Compare two datasets".

b. Set:
Compare: Exons hg19 chr22
using column: c4
Against: Exon SNP Counts, top 100
using column: c2
To find: Matching rows of 1st dataset

The exon name, the common value between the two datasets, is in column 4 in the exons dataset and column two in the counts dataset.

c. Click Execute.

20. Rename and format the final result dataset.

    a. Click on the new dataset's pencil icon, copy the old name to the Info text box, and type `SNP Coding Exons chr22` in the Name text box. Finish by clicking the Save button. Click on the new history item's pencil icon to name and format the BED file.

    b. Set "Score column for visualization:" to "5".

    c. Click on Save.

The resulting dataset contains 100 rows from the Exons dataset. Each row contains a full BED record. This dataset can now be used anywhere a genomic interval dataset (see Basic Protocol 4), or BED dataset can be used. It can also be visualized in genome browsers.

BASIC PROTOCOL 2

LOADING DATA AND UNDERSTANDING DATATYPES

In Galaxy, information is stored in "datasets", which are analogous to files. Datasets can be added to your history by uploading files from your computer, or extracting from external data sources integrated with Galaxy such as UCSC's ENCODE datasets (Blankenberg et al., 2007; Raney et al., 2011). Transferring external data via http/ftp, copying from shared or public Galaxy histories and libraries, and running data manipulation and analysis tools within Galaxy are explained. In addition to their data contents, each Galaxy "dataset" is associated with "metadata". Metadata is information that describes the characteristics of a dataset. These can include the assigned and given names/annotation, the associated reference genome and build, the format datatype, and, frequently, additional datatype-specific labels and definitions.

In this protocol, we demonstrate how metadata are assigned and modified for common genome analysis datasets uploaded into Galaxy using the methods listed above. We also use Galaxy to transform a dataset from a custom format into a standard BED format.

Necessary Resources

Hardware

An Internet-connected computer

Software

Internet browser that supports JavaScript (e.g., most current browsers such as Mozilla Firefox, Safari, Opera, Chrome, or Microsoft Internet Explorer) and an FTP client, such as FileZilla

Files

None

1. Return to the main Galaxy interface by going to the URL *http://usegalaxy.org*.
2. Create a new history. In the History panel, click on Options and select Create New.
3. Name the new history by clicking on the text Unnamed History and entering `Basic Protocol 2`.
4. Import two ChIP-Seq mouse ENCODE control and tag datasets from a shared data library.

    a. In the top menu click on Shared Data.

    b. Enter "mouse" in the search box, and then click on ChIP-Seq Mouse Example in the search results.

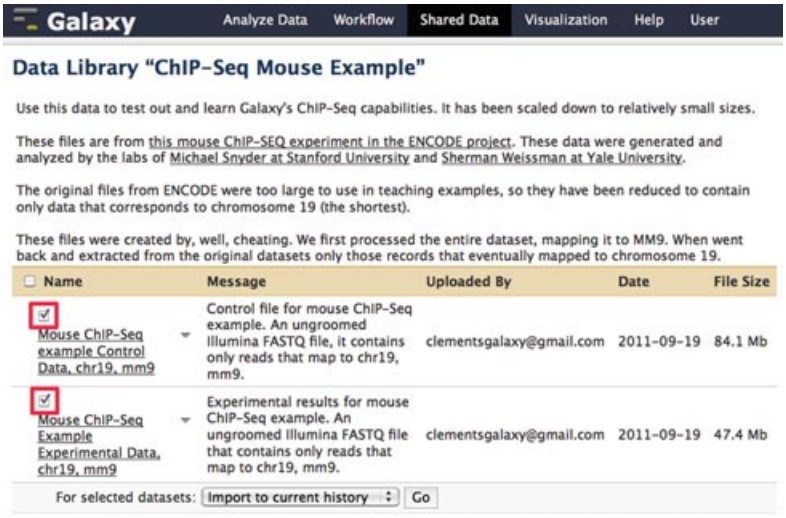


Figure 10.5.8 The data library ChIP-Seq Mouse Example is imported from a library into a history.

c. Check the "Mouse ChIP-Seq example Control Data, chr19, mm9" and "Mouse ChIP-Seq Example Experimental Data, chr19, mm9" datasets.

d. Set "For selected datasets:" to "Import to current history" and click Go as shown in Figure 10.5.8.

These datasets are raw data from an ENCODE transcription factor binding site experiment described at http://genome.ucsc.edu/cgi-bin/hgFileUi?db=mm9&g=wgEncodeSydhTfbs. The original data were generated and analyzed by the labs of Michael Snyder at Stanford University and Sherman Weissman at Yale University. An important point for this protocol is that they are all in a legacy Illumina FASTQ format and processed by Galaxy's primary tool base (as tools are backwardly compatible with older FASTQ formats). To make this protocol run significantly faster, the two datasets have been reduced to contain only data that will eventually map to chromosome 19. The original full-length files are available at http://hgdownload.cse.ucsc.edu/goldenPath/mm9/encodeDCC/wgEncodeSydhTfbs/wgEncodeSydhTfbsMelCtcfDmso20IggyaleRawDataRep1.fastq.gz. and http://hgdownload.cse.ucsc.edu/goldenPath/mm9/encodeDCC/wgEncodeSydhTfbs/wgEncodeSydhTfbsMelCtcfDmso20IggyaleRawDataRep2.fastq.gz.

e. Click "Analyze data" in the top bar to see your history.

The two imported datasets are now history items.

f. Click on the new history items' pencil icons and change their names to `Control Chr19 ungroomed` (dataset #1) and `Tags Chr19 ungroomed` (dataset #2). Finish by clicking on Save.

These datasets are in a deprecated format for short reads and will need to be "groomed" into a supported format before it is used. This grooming is done in Basic Protocol 3.

5. Upload an annotated promoter dataset via FTP.

a. Go to *http://galaxyproject.org/wiki/Datafiles/Mouse%20ChIP-Seq%20Data* in a separate Web browser window.

b. Download the file `MM9.chr19.AnnotatedPromotersWithTissueRNAP2Density.txt` to your computer.

This dataset comes from the Mammalian Promotor Database (MPromDB, http://mpromdb.wistar.upenn.edu; Gupta et al., 2011), "a curated database that strives to annotate gene promoters identified from ChIP-Seq experiment results". MPromDB is a public resource, but requires a login to download data and the data are restricted to noncommercial use.

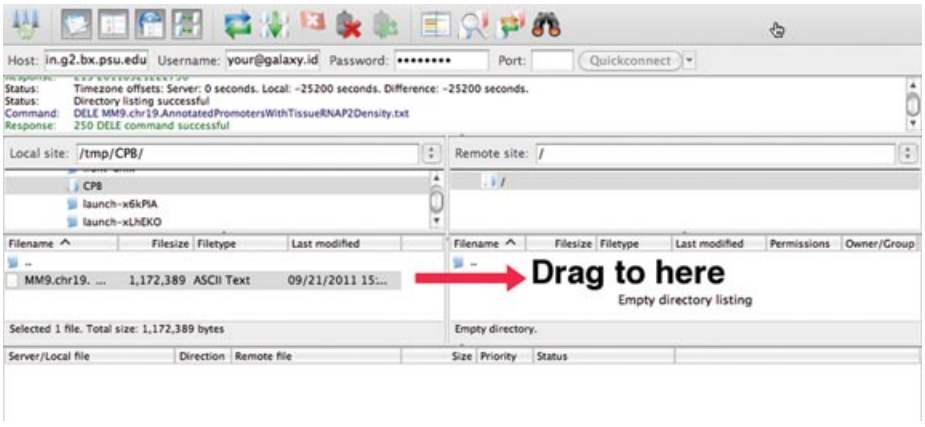


Figure 10.5.9 Filezilla (*http://filezilla-project.org)* is one example of a desktop FTP client that works well with Galaxy.

c. Launch your FTP client program. This example uses FileZilla, but any FTP client will do (Fig. 10.5.9).

d. Enter these values and click the "Quickconnect" button.
Host: `main.g2.bx.psu.edu`
Username: your username on Galaxy Main
Password: your password on Galaxy Main

e. In the "Local site:" panel in FileZilla (on the left), navigate to the directory/folder containing the downloaded file.

f. Drag the file `MM9.chr19.AnnotatedPromotersWithTissueRNAP2Density.txt` from the left panel ("Local site:") into the right panel ("Remote site:").

Depending on network speed and server load, this transfer may take several minutes.

g. Go back to the Galaxy window in your Web Browser and click Get Data in the Tools panel.

h. Click Upload File.

The `MM9.chr19.AnnotatedPromotersWithTissueRNAP2Density.txt` *file now appears in the "Files uploaded via FTP" section.*

i. Set the checkbox next to the uploaded file.

j. Set the genome text box to "mm9".

Galaxy knows about many reference genome builds, including mm9, the most recent reference for mouse. Setting this field gives Galaxy context for subsequent operations.

k. Click Execute with these parameters set as shown in Figure 10.5.10.

l. Click on the new history item's pencil icon, change the name to `MPromDB Promoters chr19`, and click "Save".

Galaxy's FTP upload interface is used with large data files to work around Web browser timeout issues when uploading files from the user's computer. Here, we first downloaded the file (from the Galaxy wiki), and then uploaded from our computer. In this case, since the file is available from a public URL, we could have just typed in the original URL in the URL/Text field and uploaded it from there (but that would not have demonstrated the FTP upload interface).

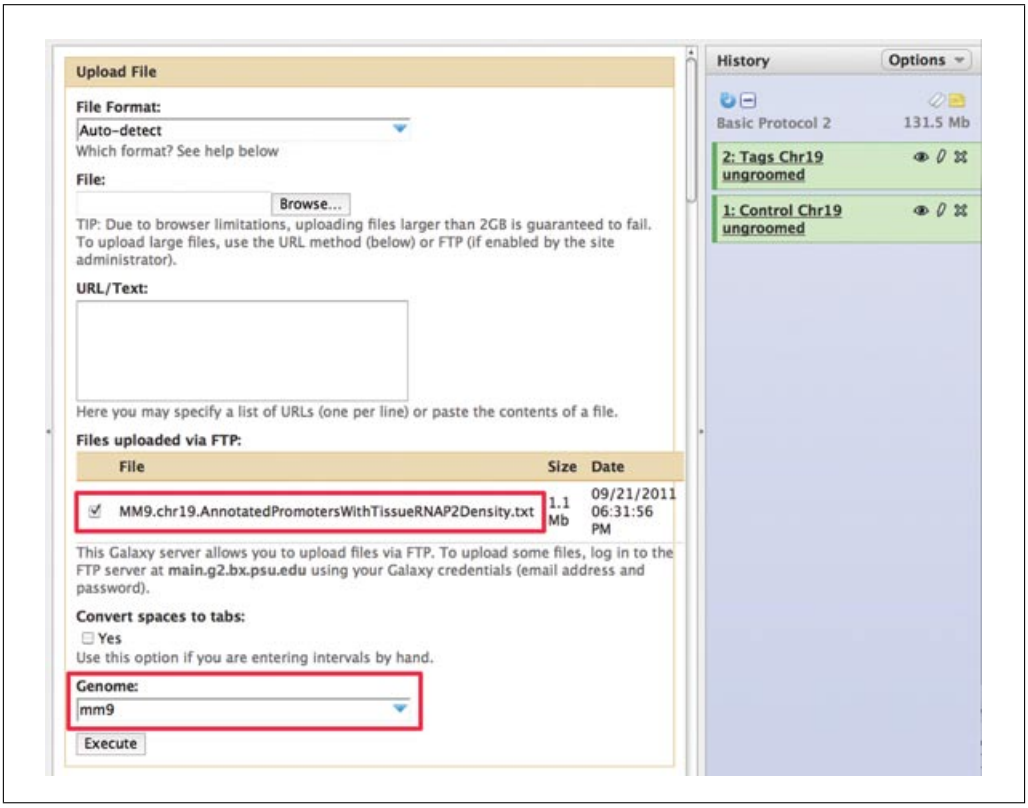


Figure 10.5.10 Get Data: Upload File tool. After a file has been uploaded using FTP, it appears in the "Files uploaded via FTP" section.

Figure 10.5.11 The Cut tool form and parameter options to select a single column (number 2, or "c2") from a tab-delimited dataset.

6. Convert the dataset to a genomic intervals format so it can be visualized and used with Galaxy's interval operations (as described in Basic Protocol 4).

   a. In the History panel, click on the "MPromDB Promoters chr19" eye icon.

      Clicking the eye icon shows a preview of the dataset in the center panel. Column 2 contains the genomic coordinates as chromosome:start..stop. To convert this file into a Galaxy genomic intervals format, this single column needs to be split into 3 columns.

   b. Locate the Cut tool in the Tools panel as shown in Fig. 10.5.11. So far, we have found tools by clicking on the tool group and then the specific tool we want. However, the Tool panel also has a search capability, which is often quicker and easier to use. To turn this on, click on Options (gear icon) in the Tools panel, and then click Show Tool Search. Type `cut` in the search box, and then click Cut under Text Manipulation.

Set:
Cut columns: c2
Delimited by: Tab
From: MPromDB Promoters chr19

Click Execute.

The resulting dataset contains only one column, the genomic coordinates, from column 2, the input dataset.

c. Split the chromosome name into its own column. Type "convert delimiters" in the Tools panel search box and click Convert under Text Manipulation. Set:
Convert all: Colons
In Query: The dataset produced by the preceding Cut operation

Click Execute.

The output dataset has two columns in it: the first containing the chromosome name, and the second the start and stop positions, separated by two periods.

d. Split the start and stop positions into separate columns. Click "Convert" under "Text Manipulation" in the Tools panel. Set:
Convert all: Dots
In Query: The dataset produced by the preceding Convert operation

Click Execute

The output dataset has three columns in it.

e. Paste the new 3 column dataset alongside the original list of promoters. Type "paste" in the Tools panel search box and select Paste under Text Manipulation. Set:
Paste: The 3 column dataset
and: MPromDB Promoters chr19
Delimit by: Tab

Click Execute.

The output dataset has 13 columns in it. The first three are the genomic coordinates, and the last 10 are from the original dataset.

f. Click the new history item's pencil icon, change its name to `MPromDB Promoters chr19 interval`, and click Save.

g. Update the new history item's datatype as well. In the center panel, under "Change data type" set "New Type:" to "interval" and click Save.

The center panel is updated and several new attributes appear, as shown in Figure 10.5.12. In this case, Galaxy correctly detects that the chromosome column is 1, and the start and end columns are 2 and 3. Galaxy did not detect the strand and name columns, but they can be easily manually assigned.

h. Tell Galaxy which columns are strand and name. In the center panel, check and set:
Strand column: 10
Name/Identifier column: 8

Click Save.

In the dataset preview in the History panel, columns 8 and 10 are now labeled Name and Strand. This dataset can now be used in any interval operation in Galaxy, including those discussed in Basic Protocol 4. This dataset can also be displayed at UCSC Main, GeneTrack, and Ensembl.

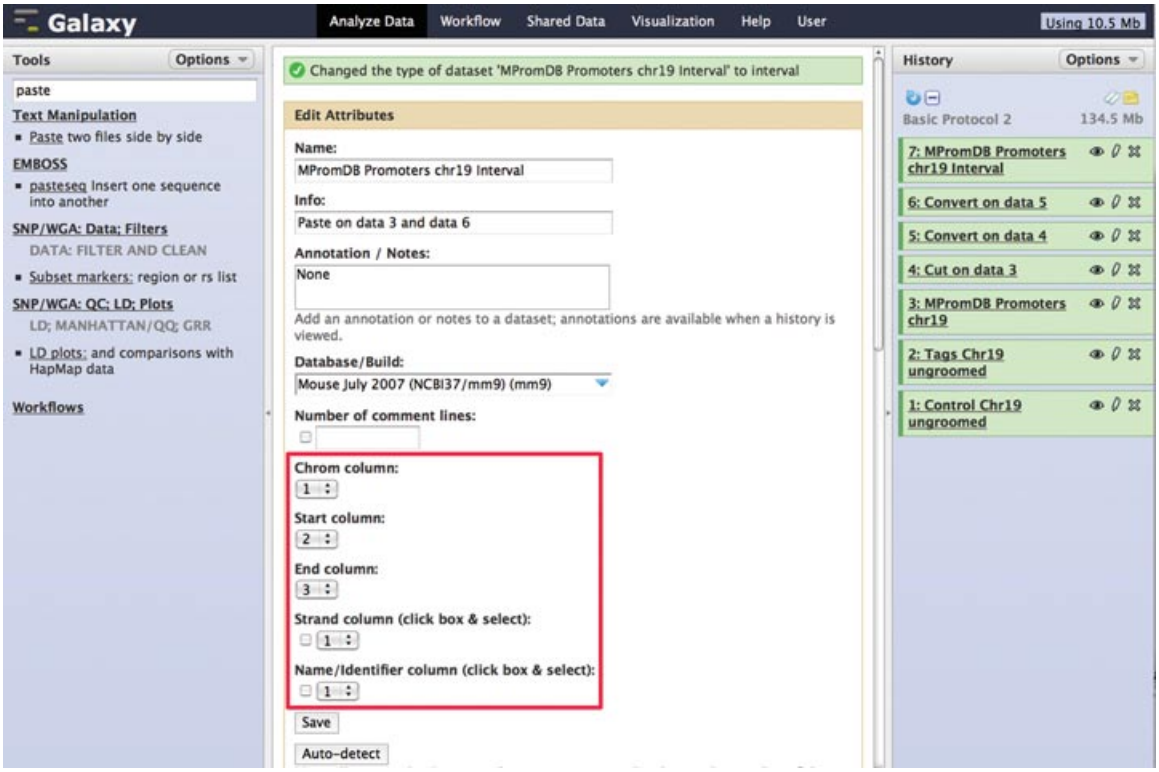


Figure 10.5.12 Edit Attributes form in center panel, showing default metadata attributes assigned for the Interval format dataset.

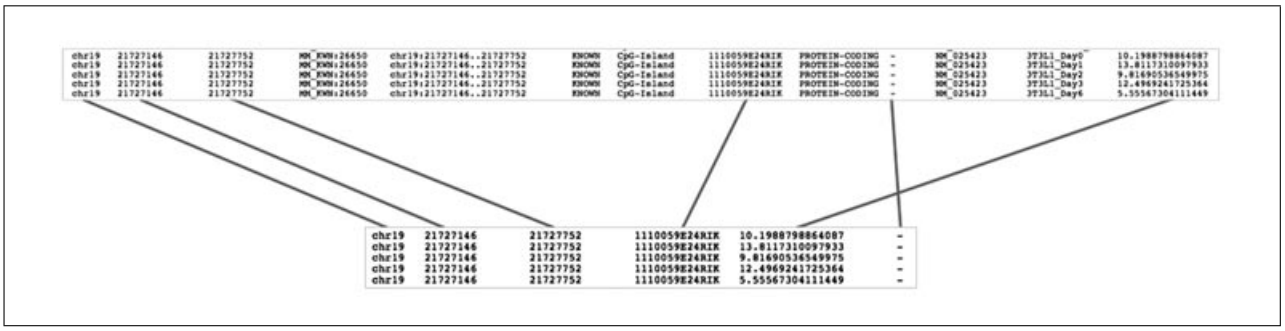


Figure 10.5.13 Diagram of the columns "Cut" from the Interval formatted dataset to create a BED formatted dataset. The result "BED6" format contains the six fields: chromosome, start (0-based), end, name, score, and strand.

7. Convert this dataset from a generic genomic interval format to BED format, which is a similar, but stricter, type of interval format. This allows the dataset to be used with tools that require BED format.

   The BED format is defined at http://genome.ucsc.edu/FAQ/FAQformat.html#format1. Not all of the data in the MPromDB file maps to columns in BED, but the data for all required and some optional BED columns are in the MPromDB dataset.

   a. Rearrange columns into BED format and drop any columns that don't exist in BED. Type `cut` in the Tools panel search box and select Cut under Text Manipulation. Set:
   Cut columns: c1,c2,c3,c8,c13,c10
   Delimited by: Tab

From: MPromDB Promoters chr19 interval

Click Execute.

The output dataset contains ~8,600 promoters (same as the input file), but contains only the 6 columns specified in the Cut tool, and those columns have been rearranged as in Figure 10.5.13. The dataset is now formatted as a BED file, but the format type has not been applied yet.

b. Click on the pencil icon for the new dataset. In the center panel, scroll past and do not use "Convert to new format". Instead, under "Change data type", enter `bed` and click Save.

This adds an additional attribute, "Score column for visualization" to the center panel. BED can include (and this dataset does) a score value in column 5. Note that if "Convert to new format" is used to transform interval to BED, the score value will be lost (and padded as "0") as it is not a defined interval format attribute.

c. Select "5" in "Score column for visualization" and click Save.

d. Rename the dataset. Set the Info text box to the default name (given by the Cut tool) and type "MPromDB Promoters Chr19 BED" in the Name text box. Finish by clicking Save.

8. Get the RefSeq gene definitions for chromosome 19.

This gene set will provide context for visualizations in subsequent protocols.

a. In the Tools panel click on "Get Data" and then "UCSC Main".

b. Make sure the following parameters are set:
clade: Mammal
genome: Mouse
assembly: July 2007 (NCBI37/mm9)
group: Genes and Gene Predictions Tracks
track: RefSeq Genes
region: position
position: chr19
output format: BED – browser extensible data
Send output to: Galaxy

c. Click the "get output" button.

This brings up the next screen of the Table Browser interface.

d. Select the Whole Gene radio button.

e. Click the "Send query to Galaxy" button.

This will create an item named "UCSC Main on Mouse: refGene (chr19:1-61342430)" in your history. This dataset contains 944 genes at the time of publication; the exact count may vary slightly as the RefSeq Genes track is updated with GenBank incremental releases (by the track source, UCSC). This has no impact on the analysis methods presented in protocols that use this dataset; however, some counts may vary slightly.

f. Click on the new history item's pencil icon and change the name to "RefSeq Genes chr19" and "Score column for visualization:" to "5" and click Save.

CALLING PEAKS FOR ChIP-seq DATA

BASIC PROTOCOL 3

The decreasing cost and increasing throughput of sequencing technologies has made chromatin immunoprecipitation followed by sequencing (ChIP-seq) an essential tool for genome-wide profiling of protein-binding, histone modification, and nucleosome positioning (Park, 2009; Pepke et al., 2009). There are numerous tools for various stages of ChIP-seq analysis, and this protocol will focus on the use of MACS (Model-based

Analysis of ChIP-Seq; Zhang et al., 2008) to perform peak calling that identifies regions of the mouse genome that are positive for zinc-finger CTCF tags versus a control. CTCF is a transcription factor that can function as either a repressor or activator. Though known to bind to several thousand different genomic locations, it has also been experimentally associated with cancer tumors including, but not limited to, testis, prostate, lung, and breast (Phillips and Corces, 2009). This protocol begins with FASTQ Tag and Control datasets that are groomed (using FASTQ Groomer, a Galaxy tool that normalizes quality scores and FASTQ formatting; Blankenberg et al., 2010) and mapped (using Bowtie, a DNA short read aligner; Langmead et al., 2009), and ends with peak calling by MACS.

Necessary Resources

Hardware

An Internet-connected computer

Software

Internet browser that supports JavaScript (e.g., most current browsers such as Mozilla Firefox, Safari, Opera, Chrome, or Microsoft Internet Explorer)

Files

Results from Basic Protocol 2, step 4 (see for sources, methods, and references).
Datasets: Main Galaxy public instance *http://usegalaxy.org*
Shared Data: Data Library: ChIP-Seq Mouse Example
1. Control file
2. Tags file
(both created or imported by user)

1. Return to the main Galaxy interface and start a new history.

   a. Go to the URL *http://usegalaxy.org*.

   b. Log into Galaxy.
      i. Hover over the top menu bar item User until the menu expands, then click on Login.
      ii. Enter Galaxy e-mail address and password.
      iii. Click on the button Login.

   c. Create a new history.
      i. Click on Options at the top of the right History panel; the submenu will expand.
      ii. Click on Create New.
      iii. Click on "Unnamed history" at the top of History panel.
      iv. Enter `Basic Protocol 3` and hit Return.

2. Load ChIP-seq input files described in Basic Protocol 2, step 4.

   a. Option A: Load from the history created by Basic Protocol 2 as shown in Figure 10.5.14.
      i. Click on Options at the top of the right History panel, the submenu will expand.
      ii. Click on Copy Datasets. A form will display in the center panel.
      iii. Select the "Basic Protocol 2" history from top left menu named "Source History:".
      iv. Click the two checkboxes for the Chip-seq datasets associated with step 4f:
      - input (control) FASTQ file named "Control Chr19 ungroomed".
      - treatment (tag) FASTQ file named "Tags Chr19 ungroomed".

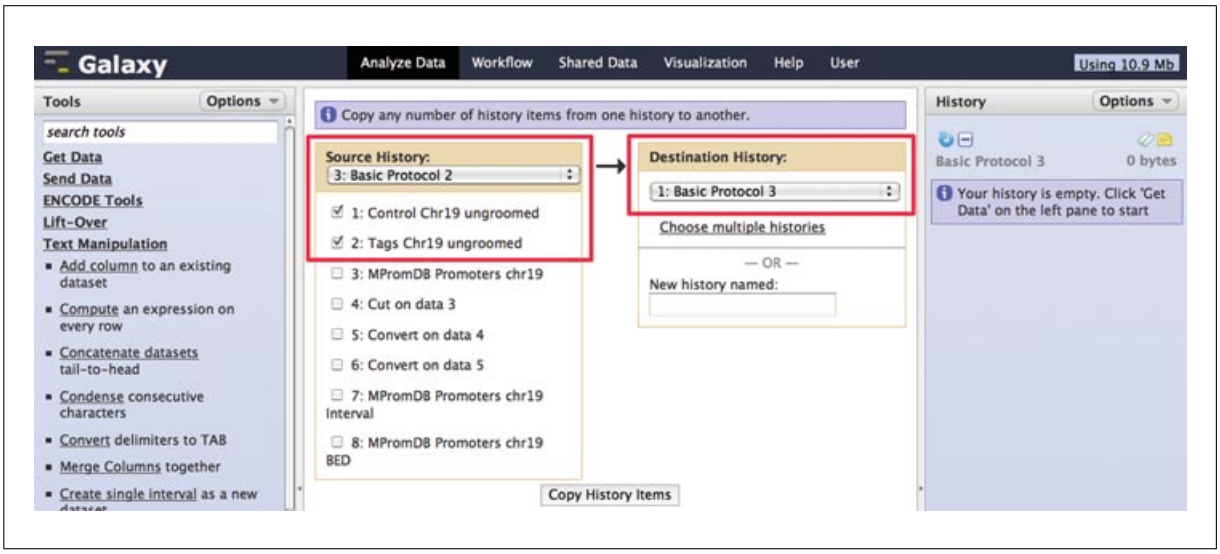


Figure 10.5.14 The Copy History form. The Source History on the left side of the center panel is the prior history from Basic Protocol 2. The Destination History on the right side of the center panel in the new history for Basic Protocol 3.

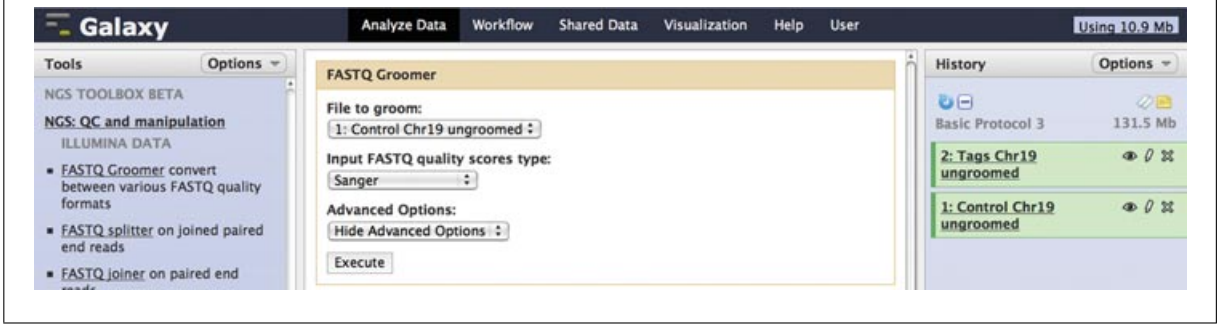


Figure 10.5.15 The FASTQ Groomer tool form in the center panel with input-data specific quality score type option selected.

v. Select the "Basic Protocol 3" history from the top right menu named "Destination History:".

vi. Click on the button "Copy History Items" at the bottom of the tool form.

After the copy completes:

- a green banner at the form top will display the following message:

"2 datasets copied to 1 history: Basic Protocol 3"

vii. Click on "Analyze Data" in the top menu bar to refresh the history panel.

The right history panel will now contain the two copied datasets.

b. Option B: Load from "Shared Data: Data Library". Follow "Basic Protocol 2, step 4".

Data from "Basic Protocol 2, step 4" are in the original, ungroomed FASTQ format from the source. These data will require grooming (format standardization) prior to mapping.

3. Groom the ChIP-seq FASTQ files as shown in Figure 10.5.15.

a. Click on "NGS: QC and manipulation" in the left Tool panel to expand the tool list.

b. Under "Illumina data:", click on "FASTQ Groomer".

c. Set “File to groom:” to “Control Chr19 ungroomed”.

d. Set “Input FASTQ quality scores type:” to “Sanger”.

e. Set “Advanced Options:” to “Hide Advanced Options”.

f. Click Execute.

g. Repeat a to f, except change c: Set “File to groom:” to “Tags Chr19 ungroomed”.

Two new history datasets will be added to the history.

h. Click on the new history items’ pencil icons and change their names to `Control Chr19 groomed` and `Tags Chr19 groomed`.

More about job status in the history panel: often the next steps in a protocol can be started before a prior job run has completed, to create a queue of related jobs that will run in sequence.

4. Map the ChIP-seq datasets to the Mouse Reference Genome using Bowtie.

a. Click on “NGS: Mapping” in the left Tool panel to expand the tool list.

b. Click on “Map with Bowtie for Illumina”.

c. Leave as default all settings except for the following, as shown in Figure 10.5.16:

d. Set “Select a reference genome:” to “Mouse (Mus musculus): mm9 Canonical Male”. Do this by typing `mm9` into the search box and selecting the genome from the match list.

“Canonical Male” indicates a reference genome that contains all of the somatic, both sex chromosomes (X and Y), and the mitochondrial genome, but none of the unmapped contigs/scaffolds.

i. Set “FASTQ file:” to “Control Chr19 groomed”.

ii. Set “Bowtie settings to use:” to “Full parameter list”.

iii. Set “Maximum permitted total of quality values at mismatched read positions (-e):” to “80”.

iv. Set “Whether or not to try as hard as possible to find valid alignments when they exist (-y):” to be “Try hard”.

v. Set “Suppress the header in the output SAM file:” by checking the box.

e. Click Execute.

This will launch the Bowtie mapping job for the input (control) dataset.

f. Repeat a to d, except change d.i., Set “FASTQ file:”, to “Tag Chr19 groomed”.

This will launch the Bowtie mapping jobs for the control and tags datasets. The result will be two new datasets added to the history.

g. Click on the new history items’ pencil icons and change their names to `Control Chr19 SAM` and `Tags Chr19 SAM`.

These SAM files represent the primary source data used to call peaks in this workflow.

SAM format (Sequence Alignment/Map) is an alignment storage file format, part of the SAM Tools utilities package (http://samtools.sourceforge.net/; Li et al., 2009).

5. Call Peaks with MACS (Model-based Analysis of ChIP-Seq).

a. Click on “NGS: Peak Calling” in the left Tool panel to expand the tool list.

b. Click on “MACS”.

c. Leave as default all settings except for the following, as shown in Figure 10.5.17:

i. Set “ChIP-Seq Tag File:” to “Tags Chr19 SAM”.

ii. Set “ChIP-Seq Control File:” to “Control Chr19 SAM”.

iii. Set “Effective genome size:” to “1.87e+9”.

Figure 10.5.16 The Bowtie tool form in the center panel with appropriate options selected. The highlighted parameters are those that are configured differently than the tool's default options.

iv. Set "Tag size:" to "36".
v. Set "Select the regions with MFOLD high-confidence enrichment ratio against background to build model:" to "32".
vi. *Optional*: Set "Parse xls files into distinct interval files:" by checking the box.

Creates optional output files in step 6, b and c.

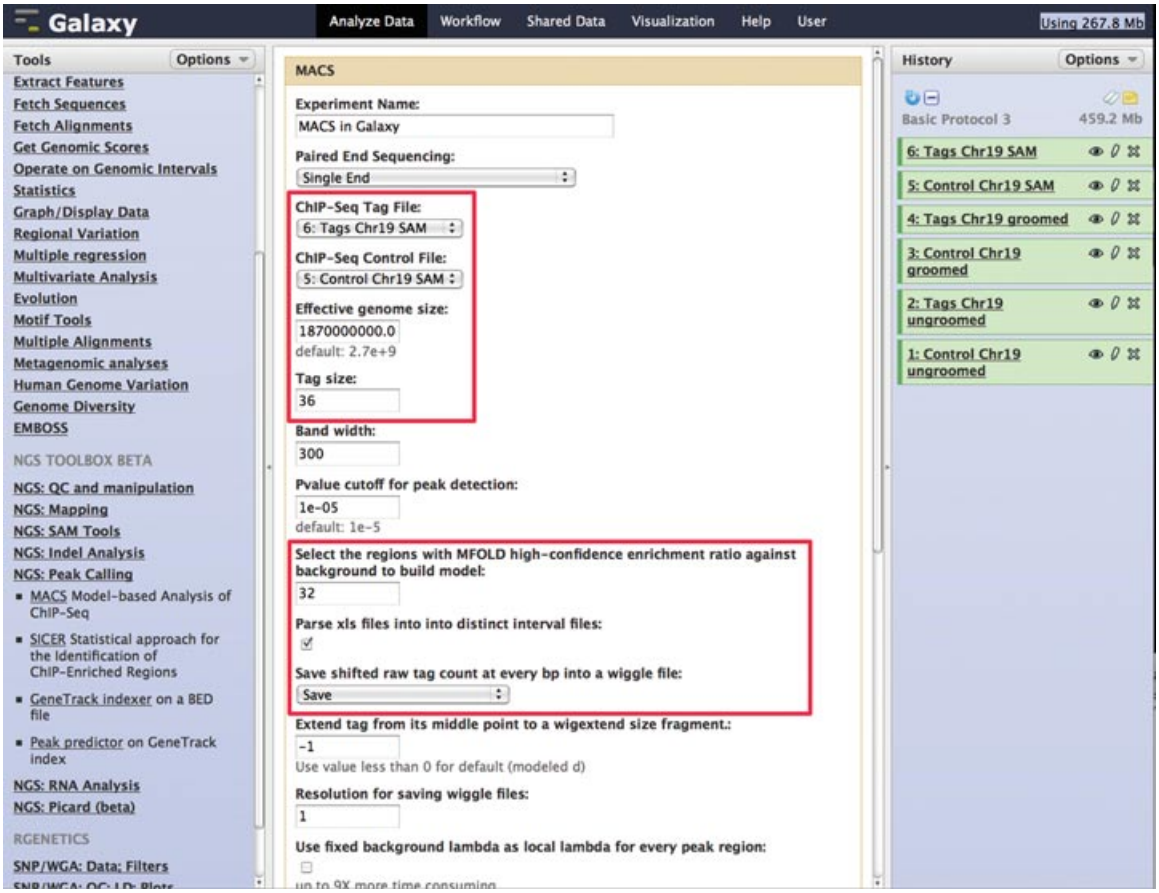


Figure 10.5.17 View of MACS tool form in the center panel with the appropriate options selected. The highlighted parameters are those that are configured differently than the tool's default options.

vii. *Optional*: Set "Save shifted raw tag count at every bp into a wiggle file:" to be "Save" and Set "Resolution for saving wiggle files:" to be "1".

Creates optional output files in step 6, d and e.

d. Click Execute.

This will launch the MACS peak calling job. The result will become 2 to 6 new datasets depending on the optional output parameters used.

6. Output datasets consist of one or more result files (a to e) and an HTML summary report (f).

Dataset results are listed in the far right history panel, and if the HTML summary report eye icon is clicked, it will display in the center panel, as shown in Figure 10.5.18:

a. standard output – peaks: `bed`
b. optional output – peaks: interval
c. optional output – negative peaks: interval
d. optional output – treatment: `wig`
e. optional output – control: `wig`
f. standard output – html report

BED and WIG are both plain-text data formats that describe discrete or continuous genome annotation features. These datatypes were developed by the UC Santa Cruz Bioinformatics Group (*http://genome.ucsc.edu*; Fujita et al., 2011).

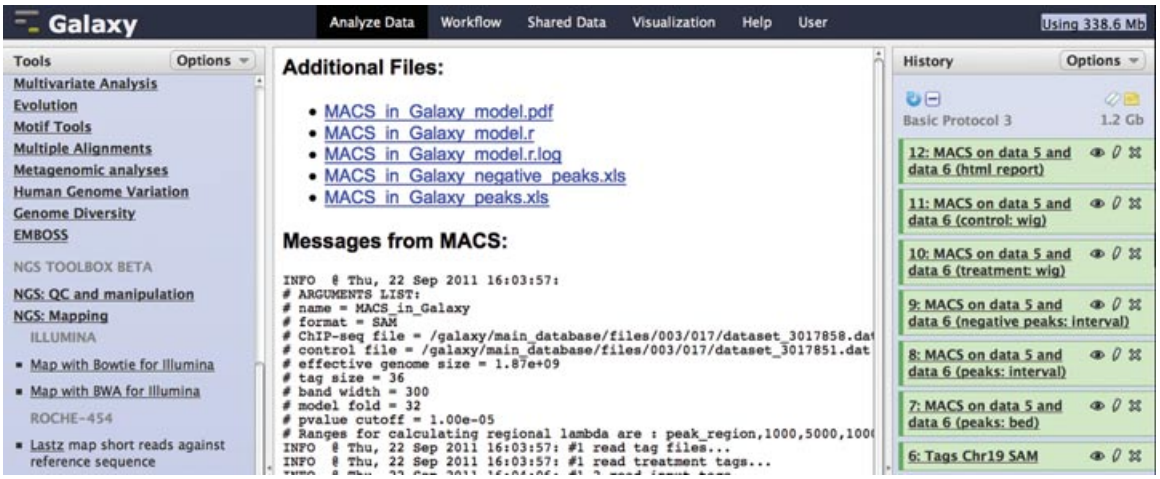


Figure 10.5.18 History result datasets and HTML report detail produced by the MACS run.

Interval format is a plain-text data format that describes discrete genome annotation features. This datatype was developed by the Galaxy Team (*http://galaxyproject.org*; Giardine et al., 2005; Blankenberg et al., 2010; Goecks et al., 2010).

7. Click on the pencil icon for dataset 6.a. to name and format the BED file.

   a. Change the name to `CTCF Peaks chr19 BED`.
   b. Set "Score column for visualization:" to "5".
   c. Click Save.

   The "CTCF Peaks chr19 BED" result file demonstrates the primary output from this ChIP-seq expression peak-calling workflow.

BASIC PROTOCOL 4

COMPARE DATASETS USING GENOMIC COORDINATES

The protocol describing finding human exons with highest SNP density (Basic Protocol 1) used the join operation to find all protein-coding exons that contain SNPs. This is just one of many interval operations offered in Galaxy, which are based on the bx-python package (*https://bitbucket.org/james_taylor/bx-python/wiki/Home*) developed at Penn State University and Emory University. These include intersect, subtract, complement, merge, concatenate, cluster, coverage, base coverage, and join. Some operations are analogous to relational database queries, such as join and coverage (*UNIT 9.2*). Other operations are analogous to set operations. Figures 10.5.19 and 10.5.20 show examples of input and output produced by individual interval operations. In the following protocol, the authors use two human chromosome 22 annotation datasets as examples. The first dataset, Exons, representing protein-coding exons, is imported from the "Basic Protocol 1" history. The second dataset "Repeats", representing interspersed repeats (also known as *transposable elements* or simply *repeats* in the text), is retrieved from the UCSC Table Browser.

Necessary Resources

Hardware

An Internet-connected computer

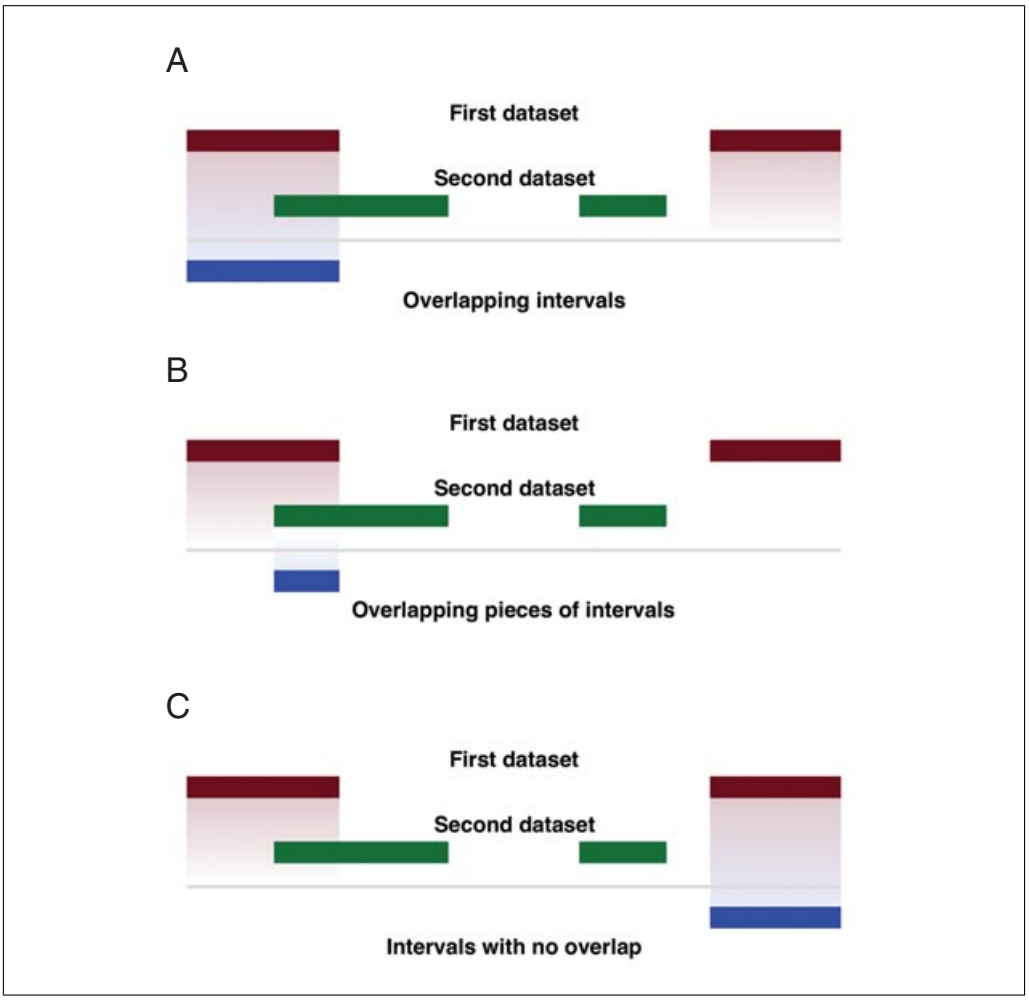


Figure 10.5.19 *(continues on following page)* Graphical explanation showing input and output datasets for several interval operations, including (**A**) Overlapping intervals, (**B**) Overlapping pieces of intervals, (**C**) Intervals with no overlap,(**D**) Non-overlapping pieces of intervals, (**E**) Concatenated intervals, (**F**) Merge.

Software

Internet browser that supports JavaScript (e.g., most current browsers such as Mozilla Firefox, Safari, Opera, Chrome, or Microsoft Internet Explorer)

Files

None

Prepare data

1. Create a new history. In the History panel click on Options and select Create New.
2. Name the new history by clicking on the text Unnamed History and entering `Basic Protocol 4`.
3. Retrieve exons for chromosome 22 dataset from the "Basic Protocol 1" history:
   a. In the History panel, click on Options and select Copy Datasets.
   b. Under the Source History pull-down menu, select Basic Protocol 1.
   c. Check the "Exons hg19 chr22" dataset.
   d. Under the Destination History pull-down menu, select Basic Protocol 4.
   e. Click Copy History Items.

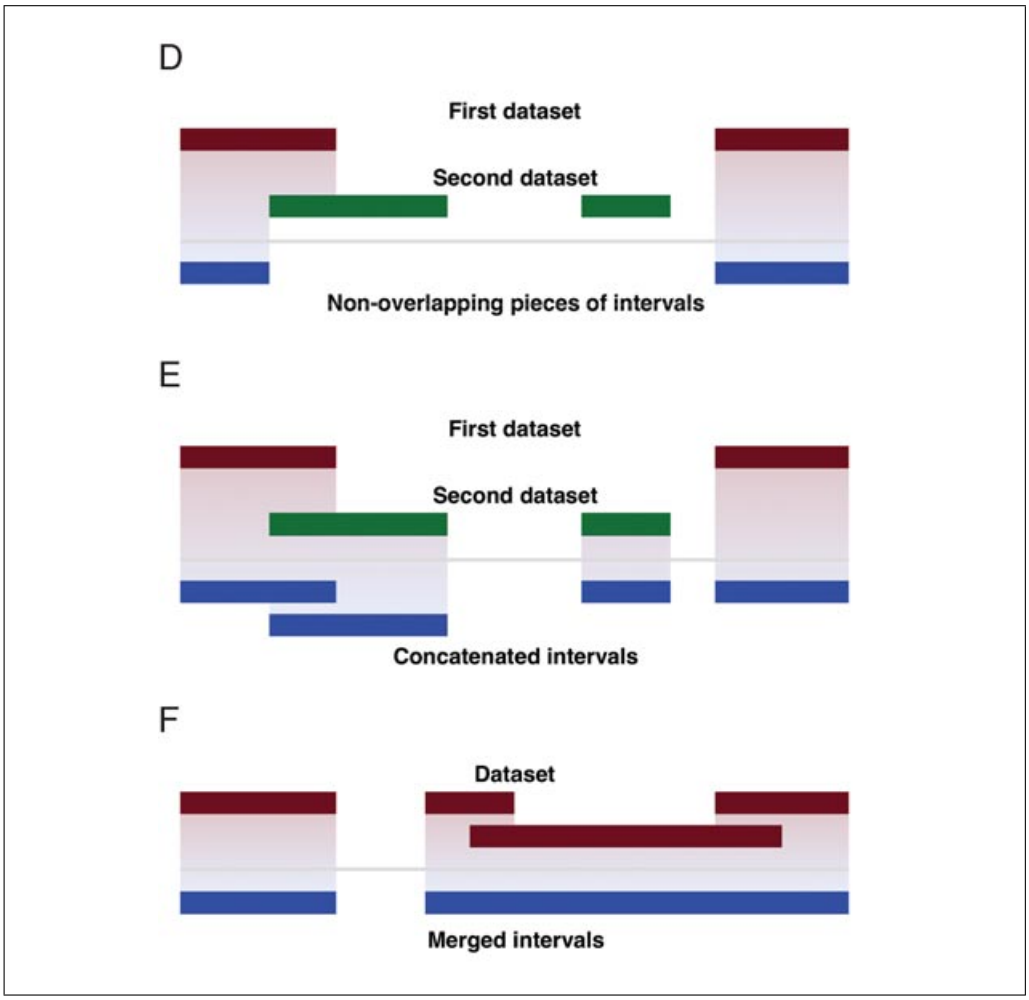


Figure 10.5.19 *(continued from previous page)*

4. Refresh the History panel and use the pencil icon (see Fig. 10.5.4) to change the name of the new dataset to Exons on the Edit Attributes form, as shown in Figure 10.5.5.

5. Retrieve repeats for chromosome 22:

   a. In the Tools panel click “Get Data” and then “UCSC Main”. Make sure the following parameters are set:
   clade: Mammal
   genome: Human
   assembly: Feb 2009 (GRCh37/hg19)
   group: Variation and Repeats
   track: RepeatMasker
   region: position
   position: chr22
   output format: BED – browser extensible data
   Send output to: Galaxy

   b. Click “get output”.

   This brings up the next screen of the Table Browser interface

   c. Click “Send query to Galaxy”.

   The history item will appear after a moment (10 to 20 sec) with the name “UCSC Main on Human: rmsk (chr22:1-51304566)”.

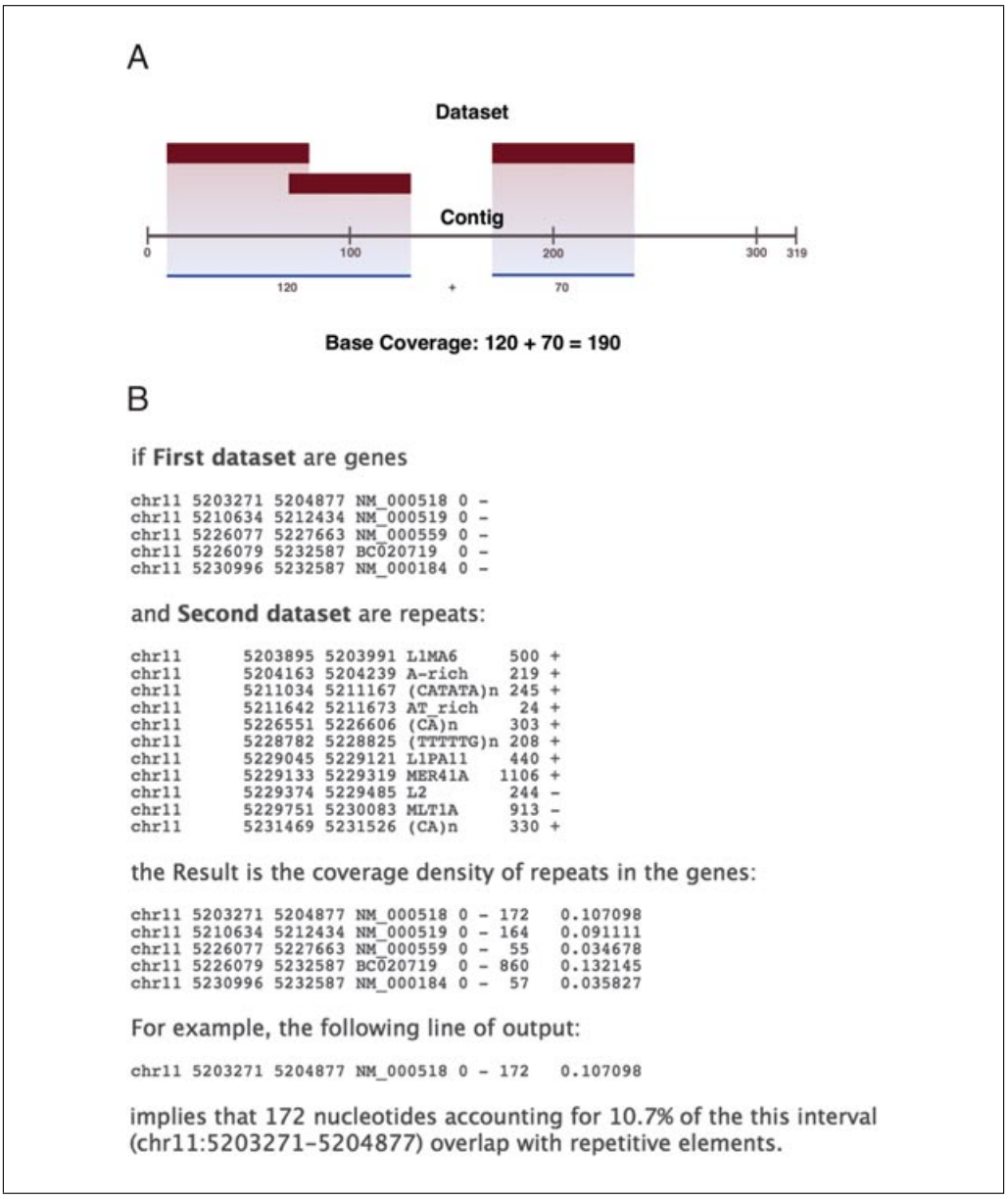


Figure 10.5.20 Examples highlighting the functionality of coverage tools.

d. Click the new dataset's pencil icon (see Figure 10.5.4) and change the name to Repeats on the Edit Attributes form, as shown in Figure 10.5.5.

You can rename the dataset before, while, or after it loads. If you rename it before or while loading, you may see warnings about missing metadata. These warnings can be ignored.

The Repeats dataset contains ~75,000 regions/rows.

We are now ready to perform interval operations on these two datasets.

6. Intersect: Find exons that overlap with one or more transposable elements, as shown in Figure 10.5.19A.

Intersect allows for the intersection of two datasets. The intersect tool can output either the entire intervals from the first dataset that overlap the second dataset (e.g., all exons containing repeats), or it can return just the intervals representing the overlap between the two datasets (e.g., only the parts of exons that are repetitive). This step demonstrates the first option.

When finding entire intervals (by setting Return to Overlapping Intervals), the order of the datasets is important. The operation will output all of the intervals in the first query that overlap any interval in the second query. It can also be thought of as a filter: intervals that do not overlap any interval in the second query will be filtered out.

a. Type `intersect` in the Tools panel search box and then click on Intersect under "Operate on Genomic Intervals". Set:
Return: Overlapping Intervals
of: Exons
that intersect: Repeats
For at least: 1

The minimum overlap of 1 requests that any overlapping regions (even if they overlap by only 1 position) will be output.

b. Click Execute.

This launches the intersect operation. A new item appears in the History panel. The resulting dataset contains ~220 regions—every coding exon that overlaps at least 1 base pair of a transposable element. The entire intervals from the coding exons dataset are output whenever there is an overlap with any transposable element interval.

7. Intersect: Find regions within exons that overlap with transposable elements, as shown in Figure 10.5.19B.

The second intersect option is to return only the pieces of intervals that overlap. When finding pieces of intervals, or the regions representing the overlap between the two datasets (by setting Return to Overlapping Pieces of Intervals), the output will be the intervals of the first dataset with the nonoverlapping subregions removed.

a. Type "intersect" in the Tools panel search box and then click on "Intersect" under "Operate on Genomic Intervals". Set:
Return: Overlapping pieces of Intervals
of: Exons
that intersect: Repeats
For at least: 1

b. Click Execute.

This launches the intersect operation. A new item appears in the History panel. The output dataset contains ~250 regions—the subregions of the exons that overlap with the intervals of the repeats. This dataset contains more regions than the previous intersect example because several exons overlap with more than one repeat.

Examine the first few rows of this dataset. The start and end columns of the new dataset are different from those in the first intersect dataset, and the exon names are repeated whenever more than one repeat intersects with that exon.

8. Subtract all: Find exons that do not contain any repeats, as shown in Figure 10.5.19C.

Subtract does the opposite of intersect. It removes the intervals or parts of intervals in the first dataset that are found in the second dataset. Like intersect, subtract can treat intervals as a whole, removing or keeping entire intervals, or it can break them apart, removing overlapping subregions. This step demonstrates the first option, returning entire intervals.

As with arithmetic subtraction, the order of the datasets is important. The second dataset is subtracted from the first dataset. The output is a variation of the first dataset and all of its columns. When subtracting whole intervals (by setting Return to Intervals with no overlap), the output will be the intervals of the first dataset that do not overlap any part of intervals of the second dataset.

a. Type `subtract` in the Tools panel search box and then click on Subtract under "Operate on Genomic Intervals". Set:
Subtract: Repeats
from: Exons
Return: Intervals with no overlap
where minimal overlap is: 1

The minimum overlap of 1 means that any overlapping regions will be removed from the output.

b. Click Execute.

This launches the subtract operation. The output dataset contains ~7000 exons that contain no transposable elements; each exon that overlaps a transposable element is removed from the output.

9. Subtract subregions: Remove subregions of exons that overlap with transposable elements, as shown in Figure 10.5.19D.

When subtracting overlapping subregions (by setting Return to "Non-overlapping pieces of intervals"), the output will be the intervals of the first dataset with the overlapping subregions removed.

a. Type "subtract" in the Tools panel search box and then click on "Subtract" under "Operate on Genomic Intervals". Set:
Subtract: Repeats
from: Exons
Return: Non-overlapping pieces of intervals
where minimal overlap is: 1

The minimum overlap of 1 means that any overlapping regions will be removed from the output.

b. Click Execute.

This launches the subtract operation. The output dataset contains ~7300 regions/rows. These are the exons minus the subregions that overlap transposable elements. This is different from the previous example: only the overlapping subregions of the exons are removed. Regions or intervals not overlapping are preserved. Thus, this dataset contains more regions than the input exon dataset: exons that overlapped with repeats have now been split into multiple regions (but still with the same exon name).

10. Concatenate and Merge: Compare coding exons and transposable elements, as shown in Figures 10.5.19E (Concatenate) and 10.5.19F (Merge).

Concatenate and Merge together are analogous to addition or union. They can be used together to combine datasets and merge (or flatten) the intervals.

Concatenate (Figure 10.5.19E) simply combines two interval datasets. The option "Both queries are exactly the same filetype" indicates that columns in both datasets are the same. If this option is unchecked, then the second dataset is adjusted to match the column assignments of the first. However, since the columns chromosome, start, end, and strand are the only columns used by the operations, all other columns will be replaced in the second dataset with a period. This option is usually left checked, as BED files are the typical interval format used within Galaxy.

Merge reads a dataset and combines all overlapping intervals into single intervals. When merging intervals, all columns besides chromosome, start, and end are lost. When two intervals are combined into one, it is ambiguous what the other columns represent or which fields should be carried over to the resulting interval. For this reason, all columns except for chromosome, start, and end are omitted from the output.

a. Enter "concatenate" in the Tools panel search box and then click on Concatenate under "Operate on Genomic Intervals". Set:
Concatenate: Exons
with: Repeats
Both datasets are the same filetype: check box

Both datasets are in BED format.

b. Click Execute.

After the operation has completed, the history item will change to a light-green color. You may click on the title of the history item to view the first few lines, or click the eye icon to view the dataset. This dataset is both datasets combined into one dataset. It contains ∼82,000 regions.

c. Type `merge` in the Tools panel search box and then click on Merge under "Operate on Genomic Intervals".

d. The previous dataset, "Concatenate on data *X* and data *Y*", should be selected in the drop-down list labeled "Merge overlapping regions of". If it is not, select the concatenated dataset.

e. Click Execute.

In this example, the two datasets are first concatenated. This outputs a BED file containing all of the intervals of both datasets. The next step, Merge (Figure 10.5.19F), merges all of the overlapping regions into single intervals. The resulting dataset has ∼59,000 rows and is a list of all of the regions on chromosome 22 that are either a coding exon, a transposable element, or both. Each region defines only the start and end position of each region. All other information is pruned from the dataset.

"Concatenate" combines datasets, and has the ability to combine interval datasets of different types.

"Merge" combines overlapping intervals into single intervals.

Together, the two operations can be used to combine intervals from different datasets into simple regions.

11. Base Coverage: Calculate the number of bases covered by all transposable elements, as shown in Figure 10.5.20A.

The Base Coverage tool (Figure 10.5.20A) calculates the number of bases covered by all of the intervals in a dataset. It does not count overlapping bases more than once; if there are two intervals referring to the same region, those bases are only counted once.

a. Type `base coverage` in the Tools Panel search box and then click on Base Coverage under Operate on Genomic Intervals.

b. Set the drop-down list labeled "Compute coverage for" to the Repeats dataset.

c. Click Execute.

Click on the title of the history item. The item will expand and display a single number in the preview area, ∼17,000,000, that is the total number of bases covered by transposable elements (about 1/3 of chromosome 22).

12. Coverage: Determine how much of each coding exon is covered by repeats, as shown in Figure 10.5.20B.

The Coverage tool (Figure 10.5.20B) is a combination of Intersect and Base Coverage. Coverage finds the number of bases each interval in the first dataset covers of the second dataset. In addition, it finds the fraction of the interval's total length that covers intervals in the second query. The resulting dataset is all of the intervals from the first input dataset, with two columns added to the end: bases covered and fraction covered. The additional two columns can be manipulated with other tools such as Filter under the Filter and Sort section of the toolbox or with Compute under the Text Manipulations section of the toolbox.

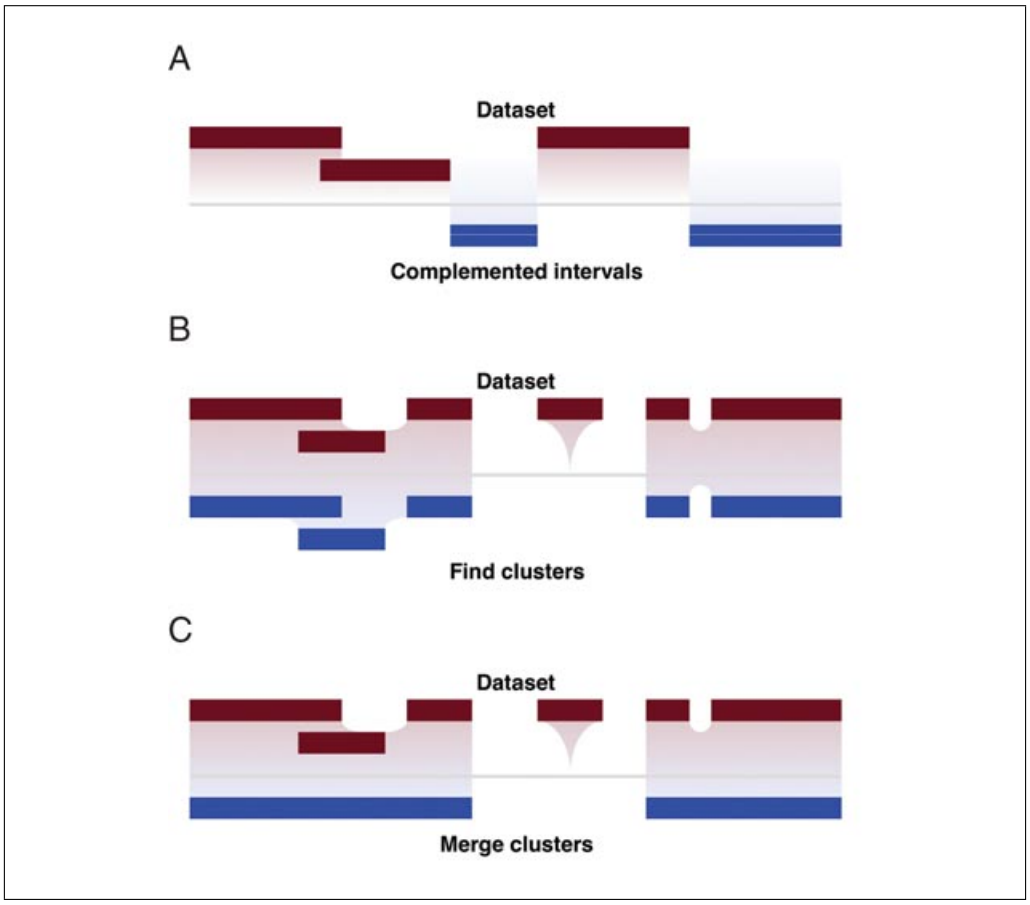


Figure 10.5.21 Graphical explanation of the (**A**) Complement, (**B**) Find clusters, and (**C**) Merge clusters interval tools.

a. Type `coverage` in the Tools panel search box and then click Coverage under "Operate on Genomic Intervals".
b. Set the drop-down list labeled "What portion of" to the Exons dataset.
c. Set the drop-down list labeled "is covered by" to the Repeats dataset.
d. Click Execute.

The resulting dataset contains all the coding exons, with two additional columns. The first additional column is the number of bases that the interval covers in the transposable elements dataset. The second additional column is the fraction of that interval that covers bases represented by the transposable elements dataset.

13. Complement: Chromosome complement of repeats on chromosome 22, as shown in Figure 10.5.21A.

The Complement tool (Figure 10.5.21A) inverts a dataset. Complement reads in all of the regions of a dataset, and outputs the regions not covered by any intervals in that dataset. The option "Genome-wide complement" allows for the entire genome to be complemented, regardless of whether a chromosome, contig, scaffold, etc., is represented in the query dataset. In a genome-wide complement of a dataset, any chromosome that does not have any intervals in the query dataset will be output in the result as the entire chromosome. In a normal complement, only the chromosomes, contigs, scaffolds, etc., that are referenced in the query dataset will be represented in the output.

a. Type `complement` in the Tools panel search box and then click Complement under "Operate on Genomic Intervals".
b. Set the drop-down list labeled "Complement regions of" to the Repeats dataset.

c. Uncheck the "Genome-wide complement" checkbox. Only chromosome 22 will be complemented.

d. Click Execute.

The resulting dataset contains ~55,000 intervals representing regions that are NOT transposable elements. Also, a normal complement is done in contrast to a genome-wide complement because the dataset was restricted to repeats from chromosome 22 (see step 5 above).

14. Cluster: Merge clusters of at least 2 transposable elements within 100 base pairs into single region elements, as shown in Figures 10.5.21B and 10.5.21C.

Cluster (Figures 10.5.21B and 10.5.21C) is one of the most versatile and powerful interval operations. Cluster finds clusters of intervals, and has a wide range of behavior depending on the options specified. The Maximum distance parameter specifies the maximum distance allowed between regions for those regions to be considered a cluster. Maximum distance can be a positive number, zero, or a negative number. When maximum distance is a positive number, regions that are at most that distance from each other are considered to be a cluster. When maximum distance is zero, Cluster considers intervals that are touching to be a cluster. This is similar to the behavior of the merge tool, but is more flexible and specific. When maximum distance is a negative number, intervals that have that amount of overlap are considered to be a cluster.

A cluster will be ignored unless it has at least as many intervals within it as specified by the parameter Minimum intervals per cluster. If this is set to 1 or lower, then all intervals, even single intervals that do not cluster with any surrounding intervals, are included in the output.

Cluster has five options for output listed in the drop-down list Return type:

***Merge clusters into single intervals** finds all of the clusters according to the criteria set by maximum distance and minimum intervals per cluster, and outputs the start and end of each cluster as an interval. The result is that clustered intervals become one large, continuous interval spanning all of the intervals within that cluster. Setting maximum distance to 0 and minimum intervals per cluster to 1 with this option produces exactly the same output as the Merge tool.*

***Find cluster intervals; preserve comments and order** finds all of the clusters according to the criteria set by maximum distance and minimum intervals per cluster, and outputs those intervals in the original order in which they were encountered in the input dataset. This option can be thought of as a filter that removes the intervals that are not found within a cluster.*

***Find cluster intervals; output grouped by clusters** finds all of the clusters according to the criteria set by maximum and minimum intervals per cluster. It is the same as the previous option, except that the intervals are grouped together in the output by cluster.*

***Find the smallest interval in each cluster** and **Find the largest interval in each cluster** first build the clusters and then return only the smallest or largest interval in each cluster.*

a. Enter `cluster` in the Tools panel search box and then click on "Cluster" under "Operate on Genomic Intervals". Set:
Cluster intervals of: Repeats
max distance between intervals: 100
min number of intervals per cluster: 2
Return type: Merge clusters into single intervals

b. Click Execute.

c. The history item changes to a light-green color when the operation completes. You may click on the title of the history item to view the first few lines, or click the eye icon to view the full ~13,500 record dataset.

The dataset returned represents clusters of transposable elements within 100 bp of each other.

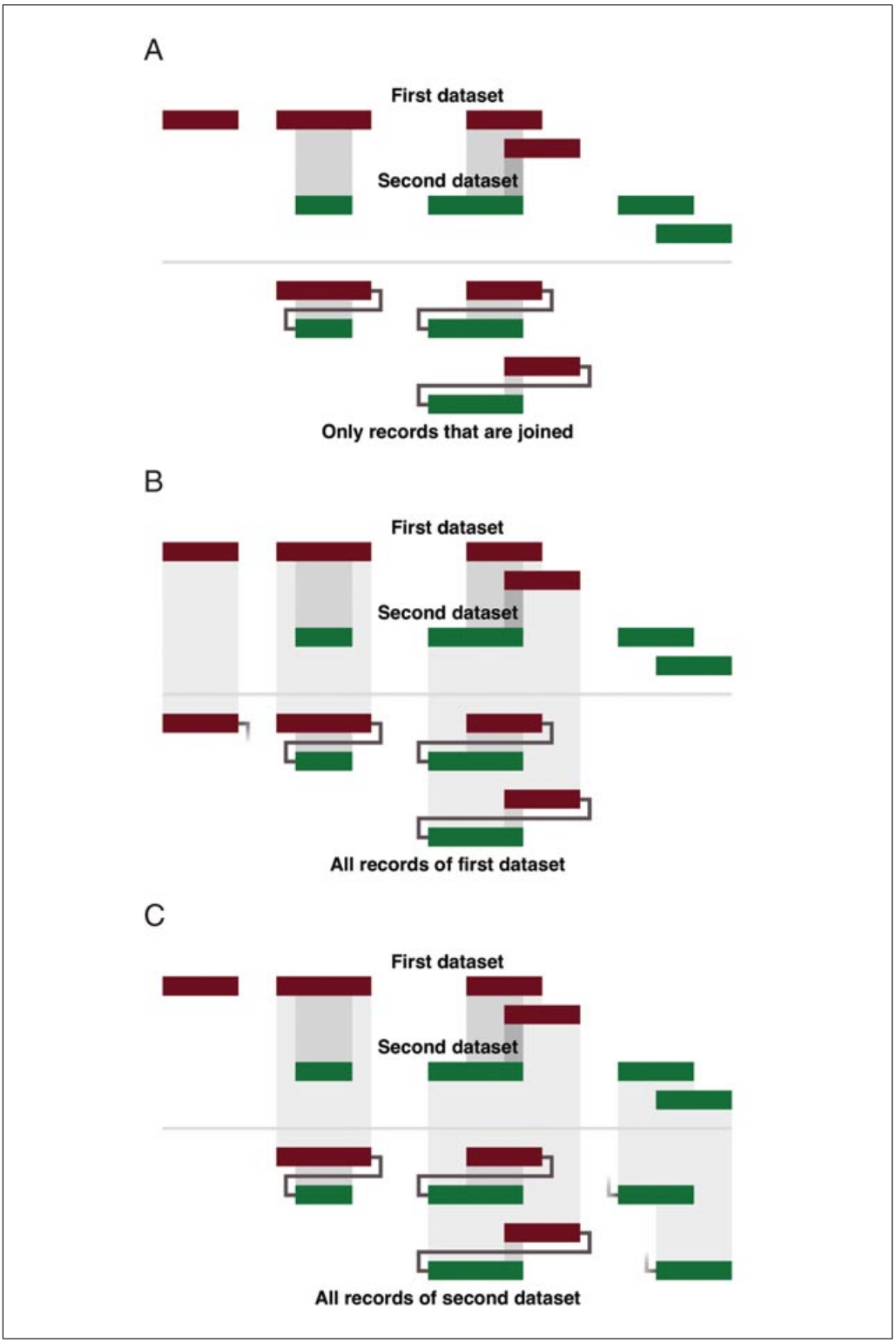
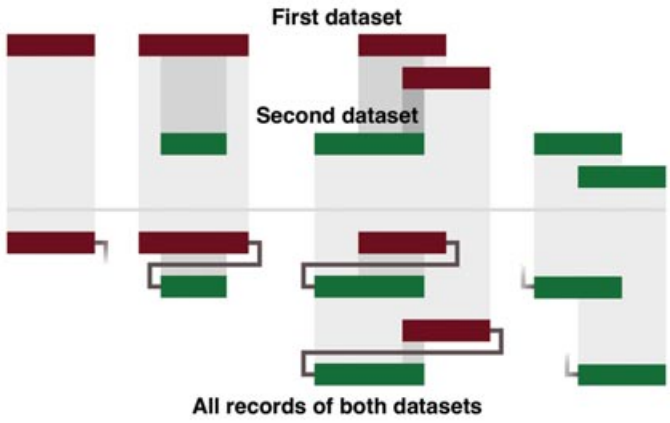


Figure 10.5.22 *(continues on following page)* Graphical explanation of genomic interval Join operations in Galaxy. (**A**) Only records that are joined, (**B**) All records of the first dataset, (**C**) Only records of second dataset, and (**D**) All records of both datasets. (**E**) Shows how all 4 variations are implemented on two small datasets.

D



E

Dataset 1

ctg15	10	49	Feature1
ctg15	70	119	Feature2
ctg15	170	209	Feature3
ctg15	180	229	Feature4

Dataset 2

ctg15	80	109	FeatureA
ctg15	150	199	FeatureB
ctg15	250	289	FeatureC
ctg15	270	309	FeatureD

Only records that are joined (INNER JOIN)

ctg15	70	119	Feature2	ctg15	80	109	FeatureA
ctg15	170	209	Feature3	ctg15	150	199	FeatureB
ctg15	180	229	Feature4	ctg15	150	199	FeatureB

All records of first dataset

ctg15	10	49	Feature1	.	.	.	.
ctg15	70	119	Feature2	ctg15	80	109	FeatureA
ctg15	170	209	Feature3	ctg15	150	199	FeatureB
ctg15	180	229	Feature4	ctg15	150	199	FeatureB

All records of second dataset

ctg15	70	119	Feature2	ctg15	80	109	FeatureA
ctg15	170	209	Feature3	ctg15	150	199	FeatureB
ctg15	180	229	Feature4	ctg15	150	199	FeatureB
.	.	.	.	ctg15	250	289	FeatureC
.	.	.	.	ctg15	270	309	FeatureD

All records of both datasets

ctg15	10	49	Feature1	.	.	.	.
ctg15	70	119	Feature2	ctg15	80	109	FeatureA
ctg15	170	209	Feature3	ctg15	150	199	FeatureB
ctg15	180	229	Feature4	ctg15	150	199	FeatureB
.	.	.	.	ctg15	250	289	FeatureC
.	.	.	.	ctg15	270	309	FeatureD

Figure 10.5.22 *(continued from previous page)*

15. Join: Compare and Join coding exons with transposable elements, as shown in Figure 10.5.22A.

    The join (Figure 10.5.22) tool's operation is similar to joins done by database management systems such as MySQL. Join looks at two datasets of intervals, and joins them based on interval overlap. Any interval in the second dataset that overlaps an interval in the first dataset will be appended to the line from the first dataset and output.

    Like intersect, join allows a minimum overlap to be specified. Intervals must meet or exceed the minimum overlap to be joined. There are several types of join that can be done, as listed in the following paragraphs. These are specified by the drop-down list labeled Return:

    ***Only records that are joined (INNER JOIN)** will only return intervals in the first query that overlap and are joined to an interval in the second query. For users of SQL databases, this is similar to an INNER JOIN (Fig. 10.5.22A).*

    ***All records of first dataset (fill null with '.')** returns all intervals from the first dataset. Any interval in the first dataset that does not join an interval in the second dataset will have the extra fields padded with a period (Fig. 10.5.22B).*

    ***All records of second dataset (fill null with '.')** returns all intervals from the second dataset. Any interval in the second dataset that is not joined to an interval in the first dataset will have fields filled in with a period (Fig. 10.5.22C).*

    ***All records of both datasets (fill nulls with a '.')** returns all of the intervals from both datasets. Intervals that do not join have fields filled in with a period (Fig. 10.5.22D). An example of output for each join option is shown in Figure 10.5.22E. Notice that in all but the first option (A), example intervals may contain invalid chromosome, start, and/or end data points (null "." values). This could result in a dataset that requires filtering to exclude "null" values before performing further operations.*

    a. Enter `join` in the Tools panel search box and then click Join under "Operate on Genomic Intervals". Set:
    Join: Exons
    with: Repeats
    with min overlap: 1
    Return: Only records that are joined (INNER JOIN)

    b. Click Execute.

    After the operation completes, the history item changes to a light-green color. You may click on the title of the history item to view the first few lines, or click the eye icon to view the dataset.

    The dataset returned contains a row for each time a coding exon overlaps a transposable element. The overlapping simple repeat is added as extra columns to the end of each line. Further analysis could use the coverage tool on this resulting dataset to calculate the amount of coverage each exon has on each repeat.

WORKING WITH MULTIPLE SEQUENCE ALIGNMENTS

BASIC PROTOCOL 5

Galaxy includes several tools to specifically work with paired and multiple sequence alignment format (MAF) datasets. The tool functions can upload, extract, and summarize the content of MAF datasets sourced from the UCSC Browser with the goal of maximizing analytical access to the underlying data. Both custom and standard MAF datasets can be uploaded and used with the majority of tools. The MAF manipulation tools used in this protocol were developed by the Galaxy team (Blankenberg et al., 2011).

Part A of this protocol will demonstrate how to extract regions from a standard Conservation MAF reference track (hg19), based on the query interval ranges from Basic Protocol 1, step 20: top 100 SNP containing human coding exons on chromosome 22.

Part B of this protocol will demonstrate how to generate coverage statistics from a standard Conservation MAF reference track (hg19), based on the query interval ranges from Basic Protocol 1, step 20: top 100 SNP containing human coding exons on chromosome 22.

Part C of this protocol will demonstrate how to extract and manipulate syntenic "transcript" FASTA sequence from a standard Conservation MAF reference track (hg19), based on the query interval ranges from a human RefSeq Genes track, as extracted in BED format from the UCSC Table Browser, limited to chromosome 22.

Necessary Resources

Hardware

An Internet-connected computer

Software

Internet browser that supports JavaScript (e.g., most current browsers such as Mozilla Firefox, Safari, Opera, Chrome, or Microsoft Internet Explorer)

Files

Results from Basic Protocol 1, Step 20 (see for sources, methods, and references):
1. SNP Coding Exons chr22
(created or imported by user)
UCSC Browser tracks for Conservation and RefSeq Genes:
2. Conservation 46-way multiZ track for hg19
(local on Main Galaxy public instance *http://usegalaxy.org*)
3. RefSeq Genes hg19 chr22
(imported by user into Galaxy history)
Workflow: Main Galaxy public instance *http://usegalaxy.org*
Shared Data: Published Workflows
4. Transform 'Stitch Gene blocks' FASTA blocks to standardized FASTA file
(imported by user)

1. Return to the main Galaxy interface and start a new history.

   a. Go to the URL *http://usegalaxy.org/*.

   b. Log into Galaxy.
      i. Hover over the top menu bar item User until the menu expands, then click on Login.
      ii. Enter Galaxy e-mail address and password.
      iii. Click on the button Login.

   c. Create a new history.
      i. Click on Options at the top of the left History pane, the submenu will expand.
      ii. Click on Create New.
      iii. Click on "Unnamed history" at the top of History pane.
      iv. Enter `Basic Protocol 5` and hit return.

Part A: Tool "Extract MAF blocks given a set of genomic intervals"

2. Copy BED file from Basic Protocol 1, step 20: "SNP Coding Exons chr22".

   a. Click on Options at the top of the right History panel; the submenu will expand

   b. Click on Copy Datasets. The form will display in the center panel.

   c. Select the "Basic Protocol 1" history from top left menu named "Source History:".

d. Click the checkbox for the file “SNP Coding Exons chr22”.

e. Select the “Basic Protocol 5” history from the top right menu named “Destination History:”.

f. Click on the button “Copy History Items” at the bottom of the tool form.

After the copy completes, a green banner at the form top will display the following message:

“1 datasets copied to 1 history: Basic Protocol 5”.

g. Click on Analyze Data in the top menu bar to refresh the history panel.

The right history panel will now contain the copied dataset “SNP Coding Exons chr22”. This data copied from “Basic Protocol 1” is a 100 line BED format file.

3. Extract conserved MAF blocks for primate species.

Primate species included in MAF Conservation 46-way multiZ (hg19)
Source: *http://hgdownload.cse.ucsc.edu/goldenPath/hg19/multiz46way/*

Human	Homo sapiens	Feb. 2009 hg19/GRCh37
Chimp	Pan troglodytes	Mar. 2006 panTro2
Gorilla	Gorilla gorilla gorilla	Oct. 2008 gorGor1
Orangutan	Pongo pygmaeus abelii	July 2007 ponAbe2
Rhesus	Macaca mulatta	Jan. 2006 rheMac2
Baboon	Papio hamadryas	Nov. 2008 papHam1
Marmoset	Callithrix jacchus	June 2007 calJac1
Tarsier	Tarsius syrichta	Aug. 2008 tarSyr1
Mouse lemur	Microcebus murinus	Jun. 2003 micMur1
Bushbaby	Otolemur garnettii	Dec. 2006 otoGar1.

a. Click Fetch Alignments in the left tool panel to expand the tool list.

b. Click Extract MAF blocks and set options as shown in Figure 10.5.23.

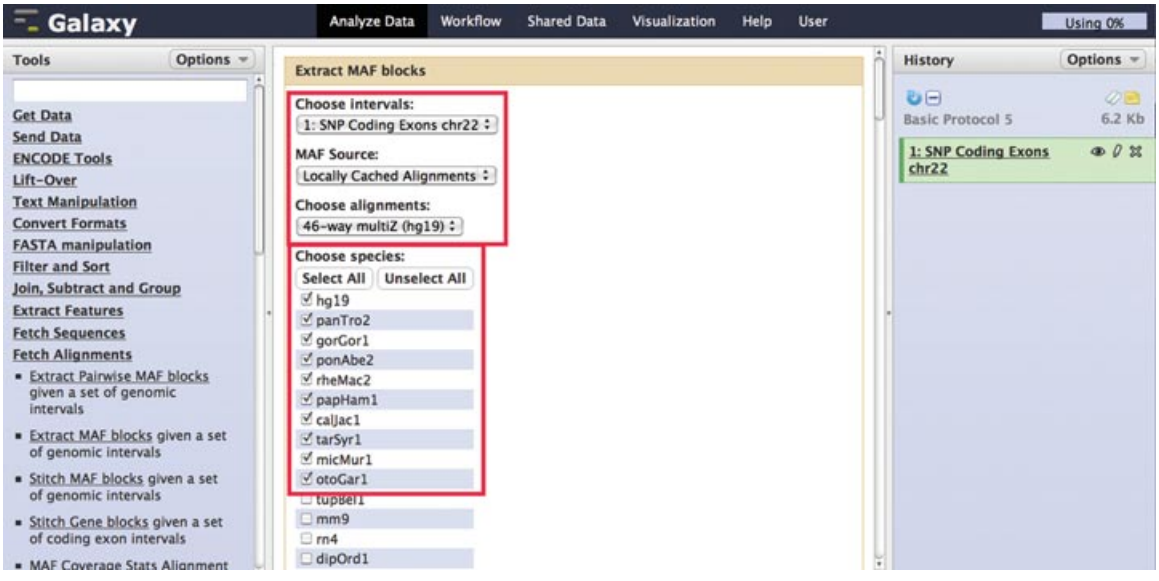


Figure 10.5.23 Extract MAF blocks tool form highlighting a subset of the tool options.

```
##maf version=1
a score=691453.0
s hg19.chr22                    8781920 71 - 51304566 CCAGCCACCATGGTGTCTTTGCTTTCCTGGTGACCCCATCCCCCTATGAGCTTTGTGCTGTGCCCCGCTAG
s otoGar1.scaffold_1065.1-12585    7572 71 +    12585 CCAGCGACCATGGTGTCTTTGCCTTCCTGGTTAGCCCTGTCCCTTACCAGCTTTGTGCTGTGCCCCGCTAG
s micMur1.scaffold_2884          118556 71 -   210213 CCAGCGACCACGGTGTCTGTCACTTCCTGGTTACCCCCTGCCCATACCAGCTTTGTGCTGTGCCCCGCTAG
s calJac1.Contig3964             199492 71 +   246953 CCAGCCCTCACGGTGTCTTTGCTTTCCTGGTGACCCCTTCCCCTTATGAGCTTTGTGCTGTGCCCCGCTAG
s papHam1.scaffold6254            96044 71 +   112551 CCAGCCACCATGGTGTCTTTGCTTTCCTGGTGACCCCATCCCCCTACGAGCTTTGTGCTGTGCCCCGCTAG
s rheMac2.chr10                 8759480 71 - 94855758 CTAGCCACCATGGTGTCTTTGCTTTCCTGGTGACCCCTTCCCCCTACGAGCTTTGTGCTGTGCCCCGCTAG
s ponAbe2.chr22                 9132863 70 - 46535552 CCAGCCACCATGGTGTCTTTGCTTTACTAAAGACCCCATCCACCTATGAGATTTA-GCTGAGACCCGCTAG
s panTro2.chr22                 8981729 71 - 50165558 CCAGCCACCATGGTGTCTTTGCTTTCCTGGTGACCCCATCCCCCTATGAGCTTTGTGCTGTGCCCCGCTAG

a score=594422.0
s hg19.chr22                    8781886 34 - 51304566 TTCAGCTTCTCGGTGCCCACTGGACAGCCCCGGC
s panTro2.chr22                 8981695 34 - 50165558 TTCAGCTTCTCGGTGCCCACCGGACAGCCCCGGC
s ponAbe2.chr22                 9132829 34 - 46535552 TTCAGATTCTCGGTGACCGCCGGACAGCACCAGC
s rheMac2.chr10                 8759446 34 - 94855758 TTCAGCTTCTCGGTGCCCGCCGGACAGCCCCGGC
s papHam1.scaffold6254            96010 34 +   112551 TTCAGCTTCTCGGTGCCCGCCGGACAGCCCCGGC
s calJac1.Contig3964             199458 34 +   246953 TTCAGCTTCTCTGTGCCCGCTGGACAGCCCCGGC
s micMur1.scaffold_2884          118522 34 -   210213 TTCAGCTTCTCGGTGCCTCCAGGACACCCCCGGC
s otoGar1.scaffold_1065.1-12585    7538 34 +    12585 TTTAGCTTCTCGGTGCCTCCCGAACACTCCCCGC

a score=1048973.0
s hg19.chr22                    8781857 29 - 51304566 TCCTCTTCTTCACCTCCCTGCTGCAGCAC
s otoGar1.scaffold_1065.1-12585    7509 29 +    12585 TCCTCTTCTTCACTTGCCTCCTGCAGCGC
s micMur1.scaffold_2884          118493 29 -   210213 TCCTCTTCTTCACCTGCCTCCTGCAGCGC
s panTro2.chr22                 8981666 29 - 50165558 TCCTCTTCTTCACCTCCCTGCTGCAGCAC
s ponAbe2.chr22                 9118221 29 - 46535552 TCCTCTTCTTCACCTGCCTGCTGCAGCGC
s rheMac2.chr10                 8759417 29 - 94855758 TCCTCTTCTTCACCTGCCTGCTGCAGCGC
s papHam1.scaffold6254            95981 29 +   112551 TCCTCTTCTTCACCTGCCTGCTGCAGCGC
s calJac1.Contig3964             199429 29 +   246953 TCCTGTTCTTCACCTGCCTGCTACAGCGC

a score=1477903.0
s hg19.chr22                    8781812 45 - 51304566 GCCGCCGTGCATGCCTCGGGGAGCCCCTGGCCCGCATGGAGCTCT
s otoGar1.scaffold_1065.1-12585    7464 45 +    12585 GCCGCCGTGCATGCCTGGGGGAGCCGCTGGCCCGCATGGAGCTCT
s micMur1.scaffold_2884          118448 45 -   210213 GCCGCCGGGCGTGCCTCGGGGAGCCCCTGGCCCGCATGGAGCTCT
s panTro2.chr22                 8981621 45 - 50165558 GCCGCCGCGCATGCCTCGGGGAGCCCCTGGCCCGCATGGAGCTCT
s ponAbe2.chr22                 9118176 45 - 46535552 GCCGCCGCGCATGCCTCGGGGAGCCCCTGGCCCGCATGGAGCTCT
s rheMac2.chr10                 8759372 45 - 94855758 GCCGCCGGGCATGCCTCGGGGAGCCCCTGGCCCGCATGGAGCTCT
s papHam1.scaffold6254            95936 45 +   112551 GCCGCCGTGCATGCCTCGGGGAGCCCCTGGCCCGCATGGAGCTCT
s calJac1.Contig3964             199384 45 +   246953 GCCGCCGCGCATGCCTCGGGGAGCCCCTGGCCCGCATGGAGCTCT
```

Figure 10.5.24 Result file produced by the Extract MAF blocks tool. Data are the MAF alignment blocks corresponding to the query interval ranges.

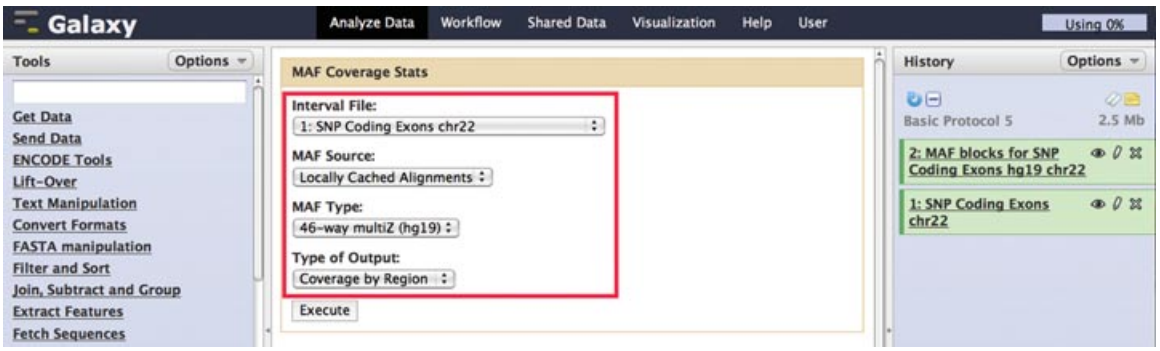


Figure 10.5.25 MAF Coverage Stats tool form highlighting the tool options.

c. Set “Choose intervals:” to “SNP Coding Exons chr22”.
d. Set “MAF Source:” to “Locally Cached Alignments”.
e. Set “Choose alignments:” to “46-way multiZ (hg19)”.
f. Set “Choose species:” by clicking the boxes for the first ten species in the list. These correspond to the primate species specified at the start of this step (5).
g. Set “Split blocks by species:” to “Do not split”.
h. Click Execute.
i. Click on the new history item’s pencil icon and change the name to `MAF blocks for SNP Coding Exons hg19 chr22`. Finish by clicking on Save.

Result file “MAF blocks for SNP Coding Exons hg19 chr22” contains the MAF alignment blocks corresponding to the 100 input hg19 exon query interval ranges. An example of this output is in Figure 10.5.24.

chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	hg19	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	bosTau4	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	calJac1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	canFam2	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	cavPor3	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	choHof1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	danRer6	173	6
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	dipOrd1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	echTel1	169	10
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	equCab2	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	felCat3	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	fr2	159	20
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	gasAcu1	173	6
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	loxAfr3	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	macEug1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	micMur1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	mm9	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	monDom5	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	ornAna1	173	6
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	oryCun2	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	oryLat2	159	20
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	otoGar1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	panTro2	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	papHam1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	petMar1	100	79
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	ponAbe2	178	1
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	proCap1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	pteVam1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	rheMac2	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	rn4	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	sorAra1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	taeGut1	179	0
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	tetNig2	163	16
chr22	42522575	42522754	NM_000106_cds_0_0_chr22_42522576_r	0	-	xenTro2	179	0
chr22	42523448	42523636	NM_000106_cds_2_0_chr22_42523449_r	0	-	hg19	188	0
chr22	42523448	42523636	NM_000106_cds_2_0_chr22_42523449_r	0	-	anoCar1	182	6
chr22	42523448	42523636	NM_000106_cds_2_0_chr22_42523449_r	0	-	bosTau4	188	0
chr22	42523448	42523636	NM_000106_cds_2_0_chr22_42523449_r	0	-	calJac1	188	0

Figure 10.5.26 Result file produced by the MAF Coverage Stats tool using the option Coverage by Region. Data are counts for covered and not covered query bases that represent predicted evidence of conservation between the two species.

Part B: Tool "MAF Coverage Stats Alignment coverage information"

4. Generate coverage statistics for "SNP Coding Exons chr22" from MAF for all species.

   a. Click Fetch Alignments in the left tool panel to expand the tool list.
   b. Click "MAF Coverage Stats" and set options as shown in Figure 10.5.25.
   c. Set "Interval File:" to "SNP Coding Exons chr22".
   d. Set "MAF Source:" to "Locally Cached Alignments".
   e. Set "MAF Type:" to "46-way multiZ (hg19)".
   f. Set "Type of Output:" to "Coverage by Region".
   g. Click Execute.
   h. Click on the new history item's pencil icon and change the name to `MAF Coverage by Region for SNP Coding Exons hg19 chr22`. Finish by clicking on Save.
   i. Repeat steps a to g, except set "Type of Output:" to Summarize Coverage.
   j. Click on the new history item's pencil icon and change the name to `MAF Summarized Coverage for SNP Coding Exons hg19 chr22`. Finish by clicking on Save.

   Result file "MAF Coverage by Region for SNP Coding Exons hg19 chr22" contains 3,440 regions, one line for each pair of query hg19 coding exon and species having an overlapping MAF alignment. Counts are for covered and not covered query hg19 exons bases that represent predicted evidence of conservation between the two species. An example of this output is in Figure 10.5.26.

   Result file "MAF Summarized Coverage for SNP Coding Exons hg19 chr22" contains 46 lines (one for each species included in the input MAF alignment data) and three columns: species, nucleotides, and coverage. "Coverage" is defined as number of nucleotides

#species	nucleotides	coverage
petMar1	34653	0.2924
mm9	101914	0.8600
gorGor1	92521	0.7808
cavPor3	106697	0.9004
eriEur1	75224	0.6348
pteVam1	97329	0.8213
panTro2	115406	0.9739
macEug1	79205	0.6684
micMur1	90866	0.7668
galGal3	67238	0.5674
proCap1	90107	0.7604
loxAfr3	99405	0.8389
echTel1	84109	0.7098
rn4	97739	0.8248
tetNig2	63319	0.5343
vicPac1	34468	0.2909
danRer6	65880	0.5560
canFam2	110150	0.9295
dipOrd1	88070	0.7432
ornAna1	73593	0.6210
sorAra1	55824	0.4711
papHam1	111249	0.9388
equCab2	107204	0.9047
bosTau4	100196	0.8455
ochPri2	85458	0.7212
myoLuc1	87795	0.7409
ponAbe2	114778	0.9686
rheMac2	107271	0.9052
oryCun2	100381	0.8471
turTru1	100676	0.8496
xenTro2	81328	0.6863
speTri1	73531	0.6205
felCat3	84483	0.7129
otoGar1	79925	0.6745
anoCar1	48072	0.4057
dasNov2	70794	0.5974
choHof1	46021	0.3884
taeGut1	67173	0.5669
oryLat2	63070	0.5322
calJac1	103396	0.8725
tarSyr1	65667	0.5542
tupBel1	77044	0.6502
fr2	63297	0.5342
gasAcu1	65029	0.5488
hg19	118499	1.0000
monDom5	84627	0.7142

Figure 10.5.27 Result file produced by the MAF Coverage Stats tool using the option Summarize Coverage. Data has three columns: species, nucleotides, and coverage, where coverage is defined as number of nucleotides divided the by the total length of the provided intervals.

divided by the total length of the provided intervals (as noted in the methods description on the MAF Coverage Stats tool form). An example of this output is in Figure 10.5.27.

Part C: Tool "Stitch Gene blocks given a set of coding exon intervals"

5. Import transcript coordinates of human RefSeq Genes from the UCSC Table Browser to Galaxy.

a. Make sure the following parameters are set:

NOTE: *Steps are identical to those in Basic Protocol 1, step 9, with the exception of step b, where Whole Gene is selected instead of Coding Exons. This similar query is shown in Figures 10.5.2A and 10.5.2B.*

clade: Mammal
genome: Human
assembly: Feb 2009 (GRCh37/hg19)
group: Genes and Gene Predictions Tracks
track: RefSeq Genes
region: position
position: chr22:1-51304566
output format: BED – browser extensible data
Send output to: Galaxy

b. Click the "get output" button.

This brings up the next screen of the Table Browser interface

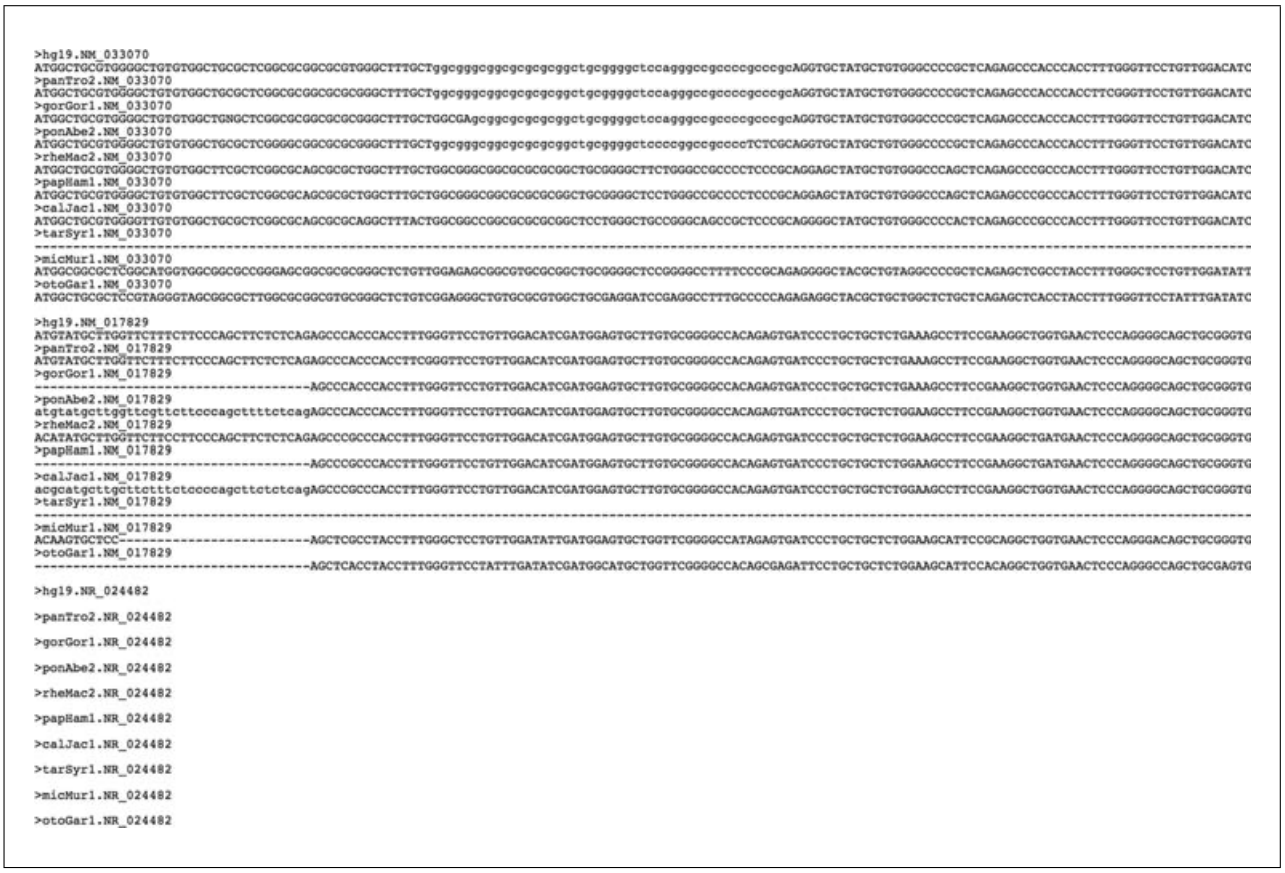


Figure 10.5.28 Result file produced by the Fetch Alignments: Stitch Gene blocks tool. Gapped bases are represented by the symbol "-". It is expected that some MAF blocks will contain results with sequence, sequence plus gaps, or gaps only. Large gaps in the query or target genome may be interpreted as a region that is not well conserved. Input type should be carefully evaluated when choosing a MAF (or any) tool. The complete absence of sequence in the input query (as in the case of a non-coding RefSeq Gene, represented in the second block of this example) produces no results (sequence or gaps) in the output. As the Stitch Gene blocks tool is specifically designed to extract and stitch coding regions from the query input BED file, this is the correct result. To perform a similar function as Stitch Gene block for non-coding genes, the tool Stitch MAF blocks would be a better choice.

c. Select the Whole Gene radio button.

d. Click the "Send query to Galaxy" button.

e. Click on the new history item's pencil icon and change the name to `RefSeq Genes hg19 chr22`. Finish by clicking on Save.

The dataset "RefSeq Genes hg19 chr22" is a 879 line, 12 column BED format file that contains complete transcription (UTR and CDS) start and stop genome coordinates.

6. Extract syntenic FASTA sequence from MAF for primate species (same 10 species as listed in Part A, Step 3). Example result data is shown in Figure 10.5.28.

   a. Click Fetch Alignments in the left tool panel to expand the tool list.

   b. Click Stitch Gene blocks.

   c. Set "Gene BED File:" to "RefSeq Genes hg19 chr22".

   d. Set "MAF Source:" to "Locally Cached Alignments".

   e. Set "MAF Type:" to "46-way multiZ (hg19)".

   f. Set "Choose species:" by clicking the boxes for the first ten species in the list. These correspond to the primate species specified at the start of Part A, step 3.

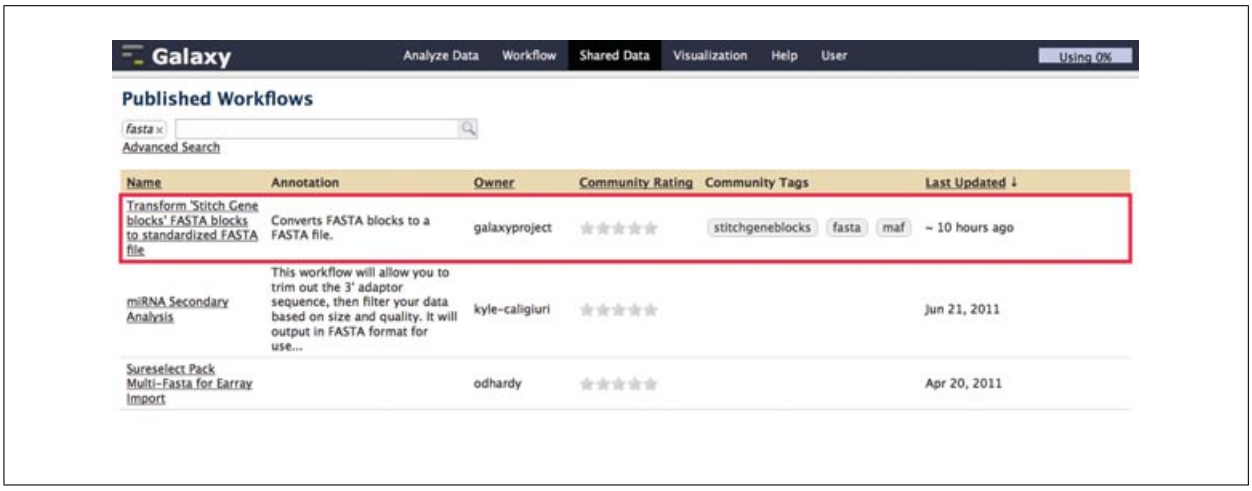


Figure 10.5.29 Shared Data: Published Workflows on the Main Galaxy instance at usegalaxy.org with the features for an individual workflow highlighted: Name (of workflow), Annotation (free text), Owner (Galaxy user name), Community Rating, Community tags (searchable keywords), Last Updated.

g. Set "Split into Gapless MAF blocks:" to "No".

h. Click Execute.

i. Click on the new history item's pencil icon and change the name to "FASTA blocks for RefSeq Genes hg19 chr22". Finish by clicking on Save.

Result file "FASTA blocks for RefSeq Genes hg19 chr22" contains predicted transcript FASTA sequence for each of the 10 species, corresponding to the input hg19 transcript query interval ranges (if conserved in the hg19 MAF data). The FASTA sequences are organized by transcript blocks and are labeled by species and the query interval's transcript name. The file will state that it contains "8,790" sequences, results of 10 species for each of the 879 input regions, but it is expected that some records will have FASTA sequence and others will not, depending on MAF content. Filtering for this content is done in steps 7 and 8. An example of the original output is in Figure 10.5.28.

7. Use a Galaxy Workflow to transform the FASTA blocks into a standardized FASTA file.

Transforming the data into a concatenated FASTA file containing only those results with sequence will make the data suitable for tools that accept nucleotide FASTA sequence as an input.

a. Hover over Shared Data in the top banner menu bar to expand the list.

b. Click on Published Workflows.

c. Enter `FASTA` into the top search box and click on the "find" icon at box's right end.

d. Click on the workflow named "Transform 'Stitch Gene blocks' FASTA blocks to standardized FASTA file", as shown in Figure 10.5.29.

e. Click on Import Workflow next to the green "plus" icon in the top right corner of the left workflow summary panel, as shown in Figure 10.5.30.

f. Click on "start using this workflow" on the confirmed import form.

g. Locate the workflow on the page "Your workflows". It will be named "imported: Transform 'Stitch Gene blocks' FASTA blocks to standardized FASTA file".

h. Click in the down arrow at the end of the workflow name to expand the list and click on Run, as shown in Figure 10.5.31.

i. Set "Step 1: Input dataset" to "FASTA blocks for RefSeq Genes hg19 chr22" in the "Running workflow:" form in the center panel, as shown in Figure 10.5.32.

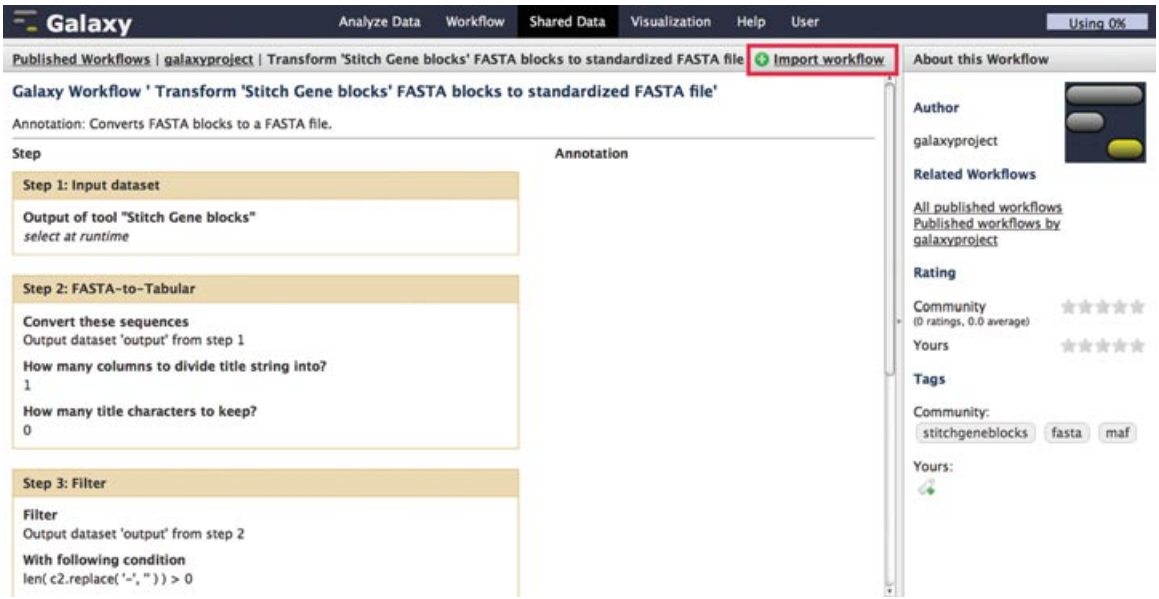


Figure 10.5.30 Detailed view of an individual workflow's steps with the "Import workflow" link highlighted.

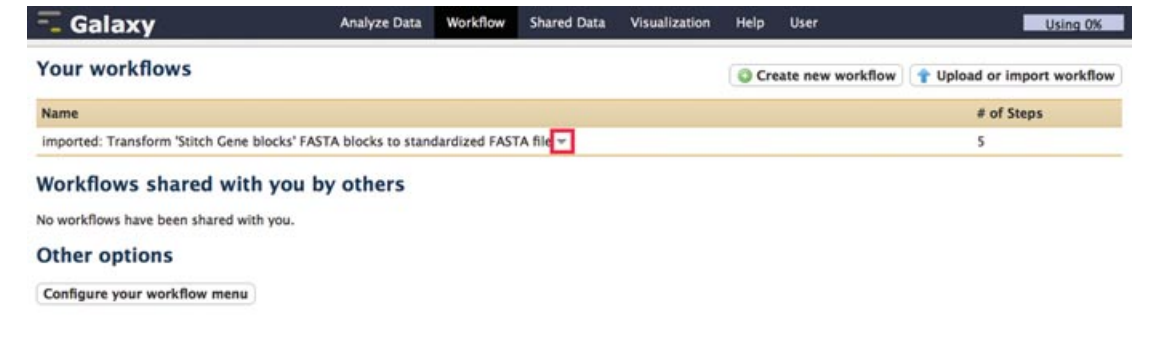


Figure 10.5.31 Your workflows page listing the newly imported workflow with the action menu highlighted. Menu selections: Edit, Run, Share or Publish, Download or Export, Clone, Rename, and Delete.

j. Click on "Run workflow".

This workflow generates 5 new datasets, some of them hidden in the history panel, as shown in Figure 10.5.33. To access these intermediate hidden datasets, click on "Options: Show Hidden Datasets" in the top right corner of the right history panel.

k. Click on the newest history item's pencil icon.
   i. Change the name to `FASTA all for RefSeq Genes hg19 chr22`.
   ii. Clear the "Database/Build:" assignment typing in `unspecified` or by clicking on the menu and selecting the top label line "—— Additional Species are Below ——", as shown in Figure 10.5.34.

   This is a dataset that contains genomic FASTA sequence from several species. To create a genomic FASTA sequence file from a single species, see the next step in this protocol (step 8).

   iii. Click on Save.

The result dataset "FASTA all for RefSeq Genes hg19 chr22" will contain 6,882 sequences and is formatted for use with tools that accept FASTA format.

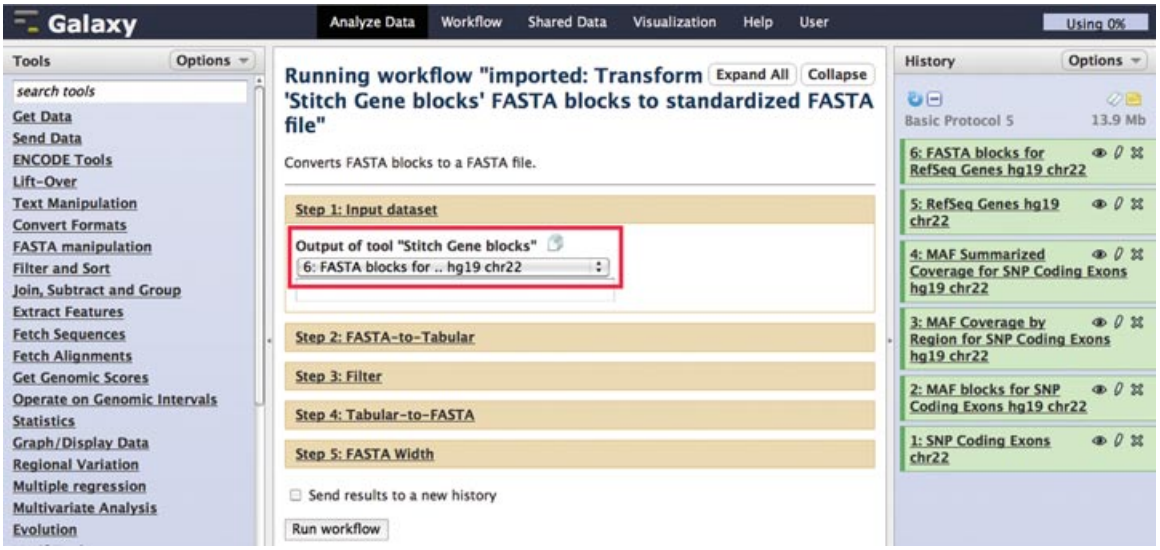


Figure 10.5.32 A workflow that is selected to Run is displayed as a form in the center panel. User-specified input selections from the current history are made by using a step's pull-down menu, as highlighted.

8. Transform the FASTA blocks into a standardized FASTA file for a single species.

   Subsetting the results by species will give the data a specific genome context and make it useable by tools that require a reference genome assignment.

   Note: Many of this protocol's operations in step 8 are the same as those bundled into the step 7 workflow. Step 8 demonstrates the individual tools in detail, showing how Galaxy's data manipulation, filtering, sorting, and format conversion tools work together in combination. Galaxy's tools most often perform a single, distinct task to maximize the ability to create customized analysis paths. Bundling multiple steps into a workflow makes customized analysis easy to apply to additional datasets and share with collaborators.

   Target species: (see step 3 for full list)

Rhesus	Macaca mulatta	Jan. 2006 rheMac2.

   "rheMac2" is the short label for the reference genome, used for the attribute "database:" and "Database/Build:" in the Galaxy user interface and file system.

   a. Click on Convert Formats in the left Tool panel to expand the list.
   b. Click on FASTA-to-Tabular and set the following options and execute.
      i. Set "Convert these sequences:" to result dataset from step 5 "FASTA blocks for RefSeq Genes hg19 chr22".
      ii. Set "How many columns to divide title string into?:" to "1".
      iii. Set "How many title characters to keep?:" to "0".
      iv. Click Execute.
   c. Click on Filter and Sort in the left Tool panel to expand the list.
      i. Click on "Filter" and set the following options and execute.
      ii. Set "Filter:" to result dataset from step b.
      iii. Set "With following condition:" to `len( c2.replace('-','')) > 0` (no double quotes), as shown in Figure 10.5.35.

      Clarification: all quotes in the string are set as single (') quotes.

      iv. Click Execute.
   d. Click on Select, set the following options, and execute.

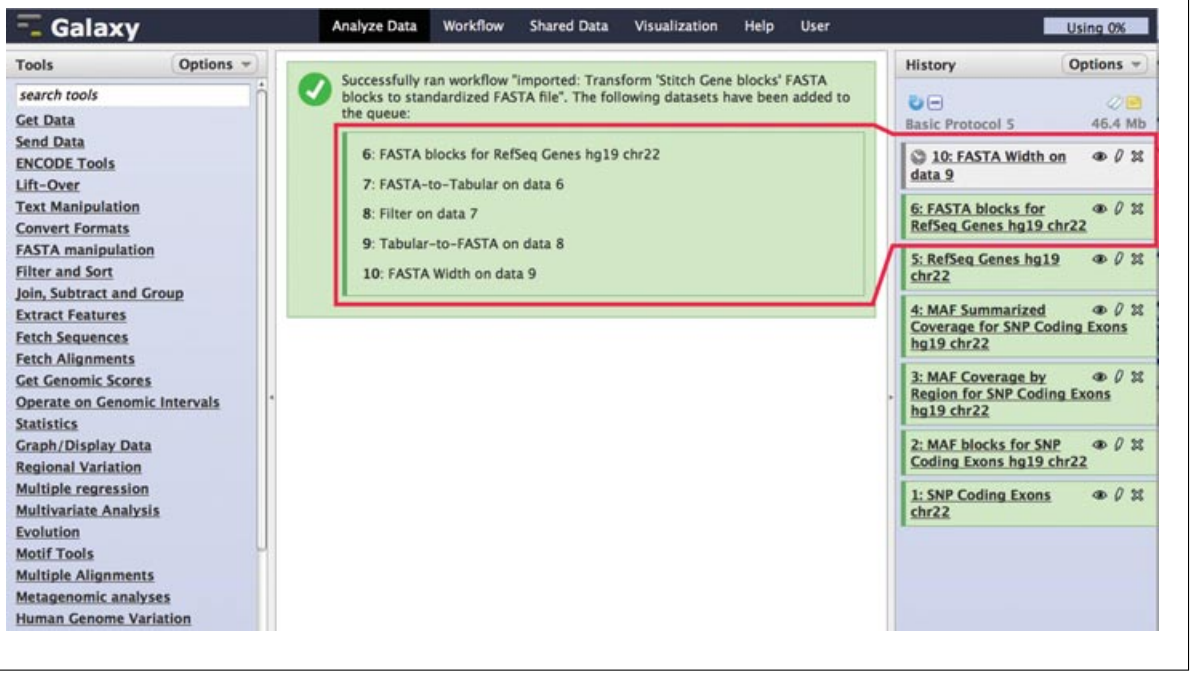


Figure 10.5.33 Confirmation display when a workflow is executed (started) successfully. As the workflow is run, individual datasets produced by the workflow steps/jobs will be independently colored as gray (waiting to run), yellow (running), green (successful), and red (error). Note that all steps in the workflow are listed, including steps that produce hidden datasets. For the color version of this figure go to *http://www.currentprotocols.com/protocol/bi1005*.

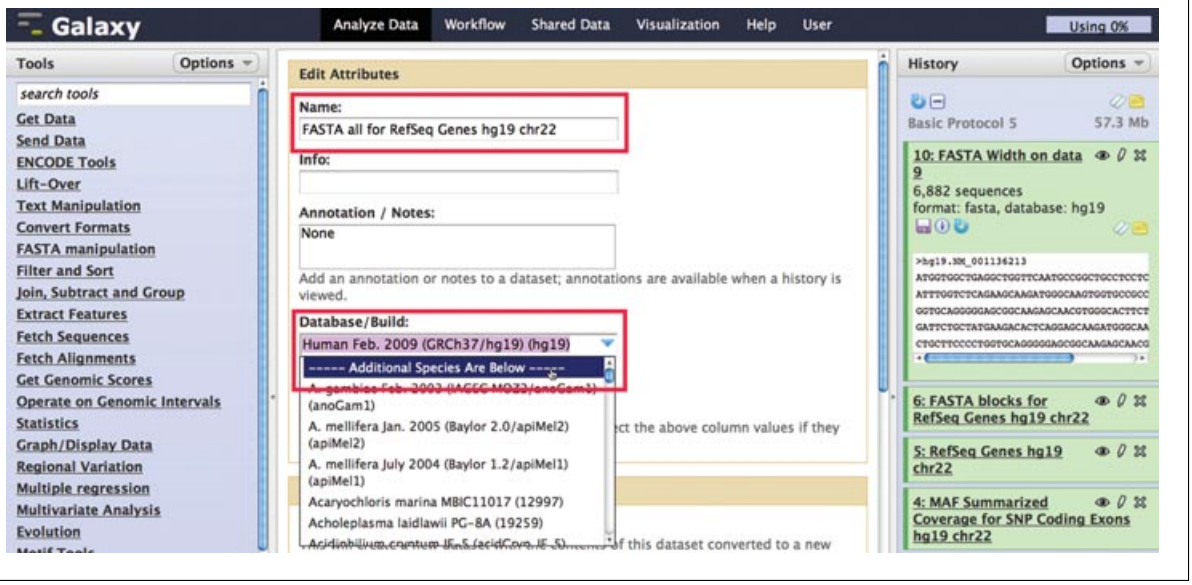


Figure 10.5.34 Tools can sometimes produce datasets that no longer should be assigned to the current (or any single) reference genome. Use the Edit Attributes form to assign/reassign a new reference genome (see Figure 10.5.37) or to unassign a reference genome (as shown) by selecting the menu title (interpreted as a “null" database) from the list.

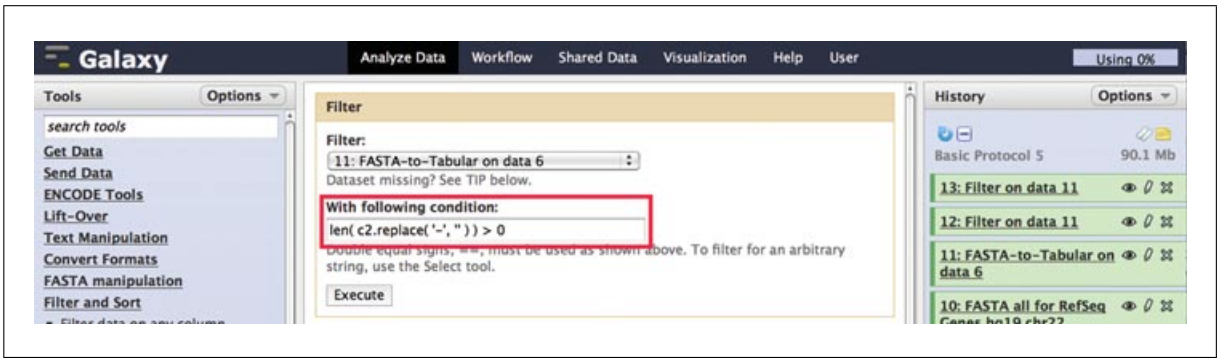


Figure 10.5.35 Filter tool form showing options, with the filter expression box highlighted containing a free text string. This specific filter string is designed to remove species rows that have no conserved genome sequence in the output of the Fetch Alignments: Stitch Gene blocks tool.

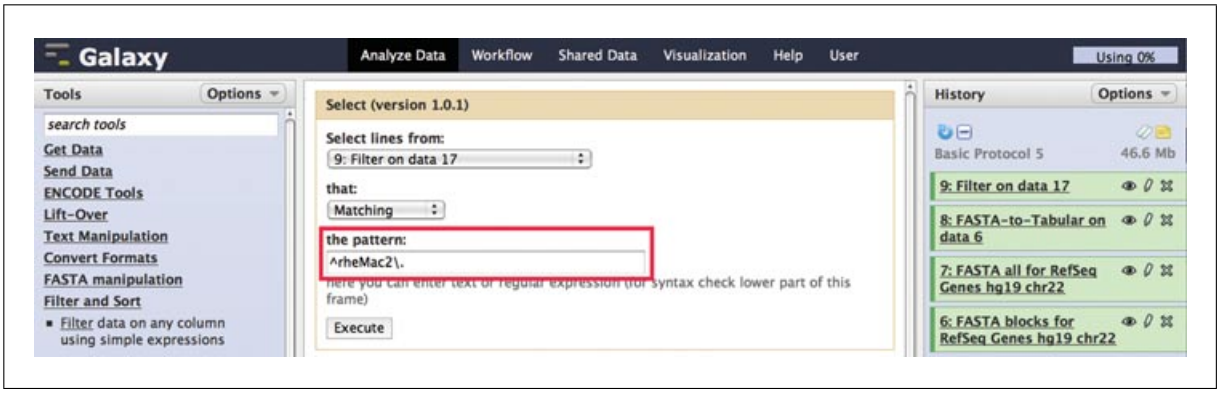


Figure 10.5.36 Select tool form showing options, with the select expression box highlighted containing a free text string. This specific select string is designed to extract lines from a file that start with "rheMac.".

i. Set "Select lines from:" to result dataset from step c.
ii. Set "that" to "Matching"
iii. Set "the pattern:" to `rheMac2/.` (no double quotes), as shown in Figure 10.5.36.

Using the reference database short label assigned in the name (FASTA sequence identifier value) to select only those sequences for this species and genome build.

iv. Click Execute.

e. Click on Convert Formats in the left Tool panel to expand the list.

f. Click on Tabular-to-FASTA and set the following options and execute.
i. Set "Tab-delimited file:" to result dataset from step e.
ii. Select "c1" in the "Title column(s):" list.
iii. Set "Sequence column:" to "c2".
iv. Click "Execute".

g. Click on "FASTA manipulation" in the left Tool panel to expand the list.

h. Click on "FASTA Width formatter", set the following options, and execute.
i. Set "Library to re-format:" to result dataset from step g.
ii. Set "New width for nucleotides strings:" to "50".
iii. Click Execute.

i. Click on the new history item's pencil icon.
i. Change the name to `FASTA rheMac2 for RefSeq Genes hg19 chr22`.

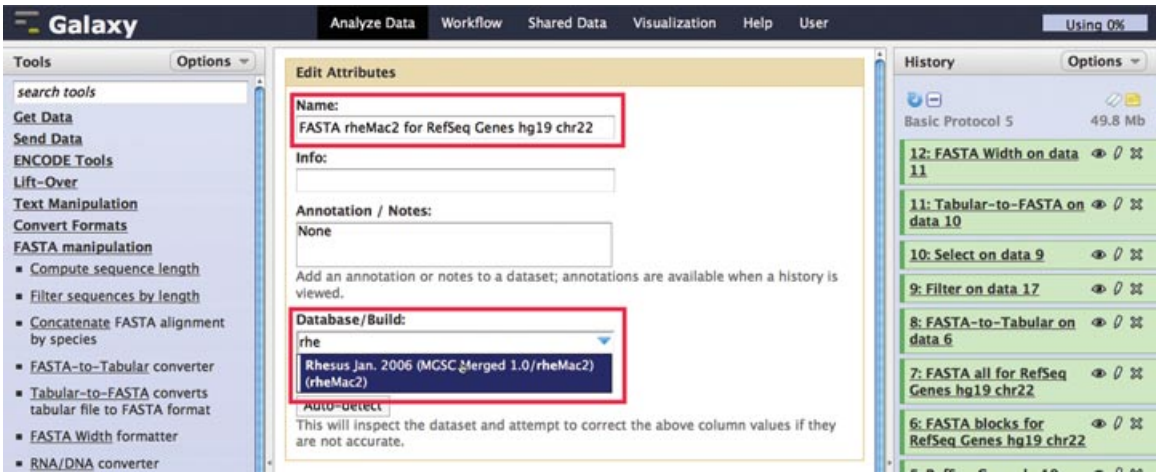


Figure 10.5.37 Tools can sometimes produce datasets that no longer should be assigned to the current (or any single) reference genome. Use the Edit Attributes form to assign/reassign a reference genome (as shown, in this case rheMac2) or to unassign a reference genome (see Figure 10.5.34).

ii. Set "Database/Build:" to "Rhesus Jan. 2006 (MGSC Merged 1.0/rheMac2) (rheMac2)". Do this by typing `rhe` into the box and selecting the full database name from the search result list, as shown in Figure 10.5.37.
iii. Click Save.

Result dataset "FASTA rheMac2 for RefSeq Genes hg19 chr22" contains predicted transcript FASTA sequence for only the rheMac2 species/build, corresponding to the input hg19 transcript query interval ranges (when conserved in the hg19 MAF data). Reassignment of the database attribute ensures that this dataset will be used correctly with downstream analysis tools.

GUIDELINES FOR UNDERSTANDING RESULTS

Galaxy was designed to be an interactive system, and in most cases results will be self-descriptive, depending on which tools were applied to the original data. As always, caution should be used when interpreting genomic data—the information produced by Galaxy is only as good as the underlying data imported.

COMMENTARY

Background Information

Modern Web-based genomic resources offer many facilities for retrieval and visualization of data. However, few of these resources offer sophisticated tools for further analysis of these data. As a result, almost every experimental biologist has to analyze data on his/her own, struggling with numerous difficulties arising from format incompatibility or incomprehensible user interfaces. Although our computational colleagues are happy to help, few are willing to devote time and resources to develop a good user interface (a significant challenge). Galaxy is a system designed to help both sides. For experimental biologists, Galaxy provides an intuitive user interface offering a direct connection to many widely used data sources and browsers, a simplified FTP data-loading procedure, and a custom genome option for most tools including the native Galaxy Track Browser (GTB, or *Trackster*). The Galaxy workspace includes a unique history system to organize, label and display data, to track datasets and analysis for sharing and/or publishing, and to extract analysis functions into workflows for re-use. For computational biologists, Galaxy provides a framework that can integrate command-line tools with almost no effort. For each tool, Galaxy generates an interface and provides all housekeeping (e.g., input and output management, job control, error catching,

and testing facilities). As this text was compiled with experimental biologists in mind, it does not contain any information on technical aspects of the Galaxy system (found at *http://galaxyproject.org*).

Critical Parameters and Troubleshooting

Galaxy allows performing an infinite number of analyses on genomic data. In designing the system, the authors tried to put as few constraints on the user as possible. In that sense Galaxy is similar to a car with a manual gearbox—it gives you more control if you know what you are doing (e.g., you do not shift from fifth to reverse). Fortunately, user feedback provides convincing evidence that a short test drive is sufficient to understand how Galaxy works. This text is equivalent to such a test drive. Below, the authors list the most common problems encountered by Galaxy users. They can be condensed into two categories: (1) data format issues and (2) genome build incompatibilities.

Data format issues

Galaxy "understands" several datatypes including genomic coordinates (e.g., BED, GFF/GTF, Wig), sequences (e.g., FASTQ, FASTA), and alignments (e.g., SAM/BAM and MAF). Most of the tools require data to be in one of these formats. For example, the genomic intervals operations described in Basic Protocol 4 can only be performed on data in *interval* format. In most cases, changing your data to interval format is as simple as correctly setting metadata, as shown in Basic Protocol 2, step 6.

Genome build incompatibilities

Galaxy supports interactive genome analyses that use a mix of different genomes within a single analysis space (History). In the authors' opinion, such "mixing" is essential for a true comparative genomics resource. The ease of mixing also means that, in some cases, users will accidentally attempt comparing data from different genomes. Thus, when using tools that operate on more than one history item (i.e., most genomic interval operations), make sure that all data come from the same genome build.

If you have questions

Galaxy has a vibrant and growing user and developer community. If you want to learn more or encounter problems, the best places to find out how to get connected are in the Galaxy Wiki (*http://galaxyproject.org*), specifically our Learning Hub (*http://galaxyproject.org/Learn*) and Support Resource (*http://galaxyproject.org/Support*) pages.

Acknowledgments

A vision for Galaxy was originally articulated by Ross Hardison, who is also the major source of support and critical feedback. The authors would like to thank Jim Kent and David Haussler for their continuing support and making UCSC Genome Browser uplink and connection possible. Istvan Albert pioneered initial aspects of Galaxy design. Efforts of the Galaxy Team (Enis Afgan, Guru Ananda, Dannon Baker, Nate Coraor, Jeremy Goecks, Greg Von Kuster, Ross Lazarus) were instrumental in making this work happen. The following individuals also contributed to the Galaxy project at different stages: Richard Burhans, Ramkrishna Chakrabarty, Laura Elnitski, Belinda Giardiane, Bob Harris, Jianbin He, Kanwei Li, Webb Miller, Cathy Riemer, Kelly Vincent, and Yi Zhang. Robert Castelo, France Denoeud, Roderic Guigo, Erika Kvikstad, Julien Lagarde, and Kateryna Makova provided critical comments during software testing. Ramana Davuluri gave permission to use the MPromDB data in these protocols. This work was funded by an NIH grant GM07226405S2 to KDM, a Beckman Foundation Young Investigator Award to AN, NSF grant DBI 0543285 and NIH grants HG004909 and HG006620 to AN and JT, NIH grants HG005133 and HG005542 to JT and AN, as well as funds from Penn State University and Penn State Institute for Cyber Science and the Huck Institutes for the Life Sciences to AN and from Emory University to JT. 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.

Literature Cited

Birney, E., Andrews, D., Bevan, P., Caccamo, M., Cameron, G., Chen, Y., Clarke, L., Coates, G., Cox, T., Cuff, J., Curwen, V., Cutts, T., Down, T., Durbin, R., Eyras, E., Fernandez-Suarez, X.M., Gane, P., Gibbins, B., Gilbert, J., Hammond, M., Hotz, H., Iyer, V., Kahari, A., Jekosch, K., Kasprzyk, A., Keefe, D., Keenan, S., Lehvaslaiho, H., McVicker, G., Melsopp, C., Meidl, P., Mongin, E., Pettett, R., Potter, S., Proctor, G., Rae, M., Searle, S., Slater, G., Smedley, D., Smith, J., Spooner, W., Stabenau, A., Stalker, J., Storey, R., Ureta-Vidal, A.,

Woodwark, C., Clamp, M., and Hubbard, T. 2004. Ensembl 2004. *Nucleic. Acids Res.* 32:D468-D470.

Blankenberg, D., Taylor, J., Schenck, I., He, J., Zhang, Y., Ghent, M., Veeraraghavan, N., Albert, I., Miller, W., Makova, K.D., Hardison, R.C., and Nekrutenko, A. 2007. A framework collaborative analysis of ENCODE data: Making large-scale analyses biologist-friendly. *Genome Res.* 17:960-964.

Blankenberg, D., Gordon, A., Von Kuster, G., Coraor, N., Taylor, J., Nekrutenko, A.; Galaxy Team. 2010. Manipulation of FASTQ data with Galaxy. *Bioinformatics* 26:1783-1785.

Blankenberg, D., Taylor, J., Nekrutenko, A.; Galaxy Team. 2011. Making whole genome multiple alignments usable for biologists. *Bioinformatics* 27:2426-2428.

Fujita, P.A., Rhead, B., Zweig, A.S., Hinrichs, A.S., Karolchik, D., Cline, M.S., Goldman, M., Barber, G.P., Clawson, H., Coelho, A., Diekhans, M., Dreszer, T.R., Giardine, B.M., Harte, R.A., Hillman-Jackson, J., Hsu, F., Kirkup, V., Kuhn, R.M., Learned, K., Li, C.H., Meyer, L.R., Pohl, A., Raney, B.J., Rosenbloom, K.R., Smith, K.E., Haussler, D., and Kent, W.J. 2011. The UCSC Genome Browser database: Update 2011. *Nucleic Acids Res.* 39:D876-D882.

Giardine, B., Riemer, C., Hardison, R.C., Burhans, R., Elnitski, L., Shah, P., Zhang, Y., Blankenberg, D., Albert, I., Taylor, J., Miller, W., Kent, W.J., and Nekrutenko, A. 2005. Galaxy: A platform for interactive large-scale genome analysis. *Genome Res*. 15:1451-1455.

Goecks, J., Nekrutenko, A., Taylor, J.; Galaxy Team. 2010. Galaxy: A comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biol*. 11:R86.

Gupta, R., Bhattacharyya, A. Agosto-Perez, F.J., Wickramasinghe, P., and Davuluri, R.V. 2011. MPromDb update 2010: An integrated resource for annotation and visualization of mammalian gene promoters and ChIP-seq experimental data. *Nucleic Acids Res.* 39:D92-D97.

Karolchik, D., Baertsch, R., Diekhans, M., Furey, T.S., Hinrichs, A., Lu, Y.T., Roskin, K.M., Schwartz, M., Sugnet, C.W., Thomas, D.J., Weber, R.J., Haussler, D., Kent, W.J.; University of California Santa Cruz. 2003. The UCSC Genome Browser Database. *Nucleic Acids Res*. 31:51-54.

Karolchik, D., Hinrichs, A.S., Furey, T.S., Roskin, K.M., Sugnet, C.W., Haussler, D., and Kent, W.J. 2004. The UCSC table browser data retrieval tool. *Nucleic Acids Res.* 32:D493-D496.

Langmead, B., Trapnell, C., Pop, M., and Salzberg, S.L. 2009. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.* 10:R25.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R., and 1000 Genome Project Data Processing Subgroup. 2009. The sequence alignment/map (SAM) format and SAMtools. *Bioinformatics* 25:2078-2079.

Maglott, D., Ostell, J., Pruitt, K.D., and Tatusova, T. 2005. Entrez gene: Gene-centered information at NCBI. *Nucleic Acids Res.* 33:D54-D58.

Park, P.J. 2009. ChIP-seq: Advantages and challenges of a maturing technology. *Nat. Rev. Genet.* 10:669-680.

Pepke, S., Wold, B., and Mortazavi, A. 2009. Computation for ChIP-seq and RNA-seq studies. *Nat. Methods* 6:S22-S32.

Phillips, J.E. and Corces, V.G. 2009. CTCF: Master weaver of the genome. *Cell* 137:1194-1211.

Pruitt, K.D., Tatusova, T., and Maglott, D.R. 2005. NCBI Reference Sequence (RefSeq): A curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.* 33:D501-D504.

Raney, B.J., Cline, M.S., Rosenbloom, K.R., Dreszer, T.R., Learned, K., Barber, G.P., Meyer, L.R., Sloan, C.A., Malladi, V.S., Roskin, K.M., Suh, B.B., Hinrichs, A.S., Clawson, H., Zweig, A.S., Kirkup, V., Fujita, P.A., Rhead, B., Smith, K.E., Pohl, A., Kuhn, R.M., Karolchik, D., Haussler, D., and Kent, W.J. 2011. ENCODE whole-genome data in the UCSC genome browser (2011 update). *Nucleic Acids Res.* 39:D871-D875.

Rosenbloom, K.R., Dreszer, T.R., Pheasant, M., Barber, G.P., Meyer, L.R., Pohl, A., Raney, B.J,. Wang, T., Hinrichs, A.S., Zweig, A.S., Fujita, P.A., Learned, K., Rhead, B., Smith, K.E., Kuhn, R.M., Karolchik, D., Haussler, D., and Kent, W.J. 2009. ENCODE whole-genome data in the UCSC Genome Browser. *Nucleic Acids Res.* 38:D620-D625.

Schneider, K.L., Pollard, K.S., Baertsch, R., Pohl, A., and Lowe, T.M. 2006. The UCSC Archaeal Genome Browser. *Nucleic Acids Res.* 34:D407-D410.

Sherry, S.T., Ward, M.H., Kholodov, M., Baker, J., Phan, L., Smigielski, E.M., and Sirotkin, K. 2001. dbSNP: The NCBI database of genetic variation. *Nucleic Acids Res.* 29:308-311.

Zhang, Y., Liu, T., Meyer, C.A., Eeckhoute, J., Johnson, D.S., Bernstein, B.E., Nusbaum, C., Myers, R.M., Brown, M., Li, W., and Liu, X.S. 2008. Model-based analysis of ChIP-Seq (MACS). *Genome Biol*. 9:R137.