1 Introduction

The aim of this practical is to give you some experience handling ChIP-Seq data. We will be working with data relating to two transcription factors: Estrogen Receptor (ER) and the forkhead protein FoxA1. These are nuclear receptors that are believed to interact in breast cancer. We have data from two replicates of the ChIP experiments for each of these two transcription factors using the breast cancer cell line MCF-7. We will use peaks derived for each of the replicates from two different peak callers, and use these peaksets to construct higher-confidence consensus peaksets. Next we will compare three consensus peaksets to determine the relationship between these two transcription factors. Finally we will determine how the overlapping peaks relate to genomic structures, specifically promoter regions. The experimental data come from the Jason Carroll group at Cancer Research UK (Cambridge Research Institute), to whom we owe our gratitude.

1.1 Galaxy

The workflow is done entirely in Galaxy and should be able to be run from any compliant browser on the internet. Galaxy is a browser-based tool aimed more at biologists and other non-computational researchers who might otherwise be using, e.g. Excel. Almost everything that can be done in Galaxy can also be done (often more efficiently) on a Linux command line if the proper toolsets are installed. Besides offering a visual interface to frequently used command-line tools, Galaxy also offers job-control and workflow interfaces that enable complex sequences of commands to be linked together.

You can load Galaxy by entering the following URL:
http://main.g2.bx.psu.edu/

You can use Galaxy without registering. If you are going to use it at all regularly (and if you are analyzing genomic data you probably will), it is worth setting up a free account, which will allow you to store your data and return to your session where you left off.

Galaxy divides the screen into three panes. The main central pane is used for entering, configuring, and examining data. The left-most pane contains the commands that Galaxy recognizes, organized in categories using a simple menu system. The right pane contains a history of your session, with the results of each command numbered and shown in order so you can examine them at any time. When a command is selected from the left pane, the options are displayed in the central pane. When you hit enter to execute the command, a new job is added to your history in the right pane. When the command is completed, it will turn green. You do not have to wait for a job to complete before starting another one, so long as all the datasets you require come from completed (green) jobs.

2 Calling Peaks

The first part involves peak calling. Rather than process all the data for all eight data sets (two factors plus two inputs times two replicates), I have uploaded data for a single chromosome for a single replicate of a single factor.
2.1 Upload data

Normally, the first step in a Galaxy session is to upload data. In order to make it easier to get to the interesting bits of the practical, I have created some sessions with the data pre-loaded so we don’t have to wait to upload sequencing data.

1. To access the first datasets, enter the following URL:
   [http://main.g2.bx.psu.edu/u/rory/h/sample-alignments](http://main.g2.bx.psu.edu/u/rory/h/sample-alignments)
2. Click on “Import history” near the upper right.
3. Click on start using this history

You should see two entries, containing aligned reads for one of the experiments (second replicate of the ER ChIP, and its associated Input):

2: ER Rep2 chr 7
1: Input Rep2 chr7

The uploaded data have already been aligned. Each line represents a unique read from the sequencer, and includes the position in the genome that it aligned to, along with an alignment score. Click on the “eye” icon next to one of the datasets. This will show you the file, in “bed” format. There are six columns in this version of the format. The first three columns give the genomic address of the where each read aligned. The fourth column contains a unique “name” for the read, showing the run, flowcell, lane, tile, and X/Y coordinates of the cluster from which it was derived). The fifth column contains a score, in this case the alignment score, while the sixth column shows what DNA strand the sequence aligned to.

2.2 Data preparation

The next step is to prepare the data for peak calling.

2.2.1 Filtering

First the data should be filtered to eliminate low-confidence alignments.

1. Click on the Filter and Sort toolset in the left pane
2. Select the Filter tool.
3. In the Filter dropdown in the center pane, select dataset #1 (Input Rep2 chr7)
4. In the With following condition box, change the entry to read c5>=15. This tells Galaxy to filter the data using column 5, keeping alignments with a score of at least 15.
5. Click Execute.
6. Repeat Steps 1-5, except in Step 3, select dataset #2 (ER rep2 chr7).
7. For each new dataset, change the name by clicking on its “pencil” icon. The name should be descriptive, e.g. “ER Rep2 Filtered”

Look at how many reads are in each dataset before and after the filtering step. What percentage of reads were discarded by filtering?

2.2.2 Matching the Input reads to the ChIP reads

If you click on title of the data sets (e.g. “3: Input Rep2 Filtered”), you will notice that the Input (Control) dataset has more reads than the ER (ChIP) dataset. It is standard practice to match the number of reads in the Input to the ChIP, so we want to sample reads from the Input to match.

1. Click on the Text Manipulation toolset in the left pane
2. Click on the Select random lines tool.
3. In the Randomly select box, enter the number of reads to want to keep. This is the number of filtered ER reads.
4. In the from dropdown, select the filtered Input reads dataset (should be dataset #3).
5. Click Execute.
6. When the operation completes, click on the “pencil” icon and change the name of the resulting dataset to something useful (e.g. “Input Rep2 Filtered Sampled”)
2.3 Peak calling

We are now ready to call peaks, using the MACS peak caller that is built in to Galaxy.

1. Lower down in the left-hand pane, there is a section called NGS TOOLBOX BETA. Select the NGS: Peak Calling tools.
2. Select the MACS tool.
3. Give the peakset a meaningful name, like “ER Rep2 MACS peaks”.
4. In the ChIP-Seq Tag File dropdown, select the dataset with the filtered ER reads (i.e. dataset #4).
5. In the ChIP-Seq Control File dropdown, select the filtered sampled Input reads dataset (i.e. dataset #5).
6. Set the Effective genome size down to 75000000 as we are calling peaks on only a part of the genome.
7. Set the Tag size to 36. This is the read length.
8. Set the MFOLD high-confidence enrichment ratio to 20 so it will find enough peaks to build its model.
9. Click Execute
10. Two jobs will be added to your history. The first will contain the peaks as a bed file representing the enriched intervals. The second is a report showing the output from running the MACS process. Wait for the jobs to complete, it will take a few minutes.
11. When the jobs are complete, look at the MACS report by clicking on the “eye” icon. This contains links to some associated data. The “model.pdf” links to a Figure showing how it is calculating the fragment size by comparing the peak centres on the forward vs. the backwards strand. The “peaks.xls” link opens a spreadsheet with each peak annotated with more information, such as the highest point and FDR calculated for the peak.
12. You can see where the peaks are called in the UCSC Genome Browser by clicking on “display at UCSC main” in the information associated with the peak dataset.

3 Consensus peaksets

For the rest of the practical, we will use pre-called peaks for the whole dataset (it would take too long for you to call all the peaks yourself). For each replicate of each factor, we have called peaks using two peak callers, MACS and SWEMBL. The data are accessible at the following link: http://main.g2.bx.psu.edu/u/rory/h/sample-peaksets

Follow the directions above (Section 2.1) to make this your current history. You may have to tell it twice that you want to use the new history as this will clear out your current work.

You should see eight peaksets:
8: FoxA1 Rep2 SWEMBL
7: FoxA1 Rep2 MACS
6: FoxA1 Rep1 SWEMBL
5: FoxA1 Rep1 MACS
4: ER Rep2 SWEMBL
3: ER Rep2 MACS
2: ER Rep1 SWEMBL
1: ER Rep1 MACS

3.1 Generating a consensus peakset for each replicate

The first step is to use the two peak callers to generate a consensus peakset for each replicate. The idea is that by restricting the called peaks to those that are identified by both peak callers, we will have higher confidence that the peaks we are using represent truly enriched regions.

There is some ambiguity regarding what is meant by overlapping or consensus peaks between two peaksets. For this practical, we will be identifying regions covered by peaks from both peaksets. If peaks from each of the sets overlap by at least one base, we will identify the region covered by both of them (having a length equal to the sum of the two read sizes minus the number of overlapping bases). We accomplish this by concatenating the peaksets files and clustering the reads so only regions with at least two overlapping reads remain.
For each pair of peaksets (MACS and SWEMBL), perform the following actions:

1. Open the Operate on Genomic Intervals toolset.
2. Select the Concatenate tool.
3. In the Concatenate dropdown, select the first peakset in the pair. For example, for the first ER replicate, select the data “1: ER Rep1 MACS”.
4. In the with dropdown, select the other peakset in the pair for the replicate, e.g. “2: ER Rep1 SWEMBL”.
5. Click Execute.
6. When it completes, rename the resultant combined peakset with a useful name, like “ER Rep1 Both”.
7. Repeat these steps three more times for the remaining replicates.

You should now have four new datasets. Now repeat the following steps for each of these datasets:

8. Select the Cluster tool.
9. In the Cluster intervals of tool, select the dataset, e.g. “9: ER Rep1 Both”.
10. Keep the max distance at 1 basepair overlap, and the min number of intervals per cluster at two. Ensure the Return type is set to “Merge clusters into single intervals”. This will ensure that only overlapping regions are included in the result.
11. Click Execute.
12. When it completes, rename the dataset to something useful, e.g. “ER Rep1 Consensus”.
13. Repeat these steps three times for the other replicates.

### 3.2 Generating a consensus peakset for each factor

Now that the eight original peaksets have been reduced to four consensus peaksets, we can reduce it still further by creating a consensus peakset for each factor. This is done in the same manner as combining peaksets from different peak callers, but now we overlap the consensus peaksets for each replicate to produce a final consensus peakset for each factor (ER and FoxA1).

1. Open the Operate on Genomic Intervals toolset.
2. Select the Concatenate tool.
3. In the Concatenate dropdown, select the consensus peakset for the first replicate of the factor. For example, for the first ER replicate, select the data “13: ER Rep1 Consensus”.
4. In the with dropdown, select the other replicate consensus peakset in the pair for the factor, e.g. “14: ER Rep2 Consensus”.
5. Click Execute.
6. When it completes, rename the resultant combined peakset with a useful name, like “ER Consensus Both”.
7. Repeat these steps for the FoxA1 replicates.

You should now have two new datasets. Now repeat the following steps for each of these datasets:

8. Select the Cluster tool.
9. In the Cluster intervals of tool, select the dataset, e.g. “17: ER Consensus Both”
10. Keep the max distance at 1 basepair overlap, and the min number of intervals per cluster at two. Ensure the Return type is set to “Merge clusters into single intervals”. This will ensure that only overlapping regions are included in the result.
11. Click Execute.
12. When it completes, rename the dataset to something useful, e.g. “ER Consensus”.
13. Repeat these steps for the FoxA1 replicates.
4 Binding factor overlaps

Now that we have consensus peaksets for the two factors, we can begin to explore how their binding patterns relate. The first step is to determine which binding sites they have in common, and which are unique to each factor.

4.1 Determining shared binding sites

Finding the shared binding sites is the same as computing consensus peaksets – the common binding sites are the consensus between the two factors.

1. Open the Operate on Genomic Intervals toolset.
2. Select the Concatenate tool.
3. In the Concatenate dropdown, select the consensus peakset for the. For example, for ER, select the data “19: ER Consensus”.
4. In the with dropdown, select the consensus peakset for the FoxA1 factor, e.g. “20: FoxA1 Consensus”.
5. Click Execute.
6. When it completes, rename the resultant combined peakset with a useful name, like “Consensus Peaks”
7. Select the Cluster tool.
8. In the Cluster intervals of tool, select the dataset, e.g. “21: Consensus Peaks”
9. Click Execute
10. When it completes, rename the dataset to something useful, e.g. “Shared Sites”

4.2 Binding sites unique to each factor

We may also want to identify binding sites that are unique to the factors, that is, that are bound in the ER ChIP but are not also bound by FoxA1, and vice versa.

1. Open the Operate on Genomic Intervals toolset.
2. Select the Subtract tool.
3. In the Subtract dropdown, select the common sites peakset we just computed, e.g. “22: Shared Sites”
4. In the from dropdown, select the consensus ER peakset, e.g. “19: ER Consensus”
5. Keep the Return dropdown set to “Intervals with no overlap” and where minimal overlap is set to 1.
6. Click Execute.
7. When it completes, rename the resultant joined peakset with a useful name, like “ER Unique”.
8. Repeat these steps to create a “FoxA1 Unique” peakset.

Have a look at the shared and unique peaksets, See how many intervals are in each set, and how that relates to how many regions were in the consensus peaksets for each factor.

5 Connecting enriched regions to genomic features: Promoters

For the final part of the practical, we will associate the computed peaksets with genomic features, specifically gene promoter regions. The idea is to see of which of the binding sites occurs in promoter regions, and if any of the subsets are particularly enriched in promoters.

5.1 Load gene data

The first step is to retrieve the coordinates of the genes in the human genome.

1. Select the Get Data toolset.
2. Click on the UCSC Main table browser tool.
3. Make sure the Genome is set to the Mar. 2006/hg18 assembly of the Human genome.
4. Select the “Genes and Gene Prediction Tracks” group, and the “RefSeq Genes” track.
5. Region should be “genome”
6. Make sure the output format is “BED”; and “Send output to” has Galaxy checked.
7. Click get output.
8. On the next screen, click “send query to Galaxy”.
9. Rename the dataset to “RefSeq Genes”
5.2 Construct promoter intervals

The dataset you have downloaded contains coordinates of genes on the human genome. Promoter regions occur upstream of the genes, so now we need to create a new dataset consisting of these regions. You can use different definitions for how big the promoter regions should be. For this example, we will take regions 1,000 basepairs long immediately upstream of each gene.

1. Open the Operate on Genomic Intervals toolset.
2. Select the Get flanks tool.
3. In the Select data dropdown, select the “RefSeq Genes” dataset we just downloaded.
4. Keep Region as “Whole Feature”.
5. Keep Location of the flanking region/s as “Upstream”.
6. Keep the Offset at 0 as we want the region immediately adjacent to the start of the gene.
7. Set the Length of the flanking region(s) to 1000.
8. Click Execute.
9. Rename the resultant dataset to “Promoters”.

5.3 Compute promoter lists

Now we can see which of our peaks occur in promoter regions. We can do this for the shared sites as well as those unique to ER and FoxA1.

1. Open the Operate on Genomic Intervals toolset.
2. Select the Intersect tool.
3. Keep the Return dropdown set to “Overlapping Intervals”.
4. Set the of dropdown to the common sites, e.g. “22: Shared Sites”.
5. Set the that intersect dropdown to the set of Promoters created above, e.g. “26:Promoters”.
6. Click Execute.
7. Rename the resultant dataset to something descriptive, e.g. “Shared sites in Promoters”
8. Repeat these steps to determine the ER Unique sites overlapping with promoters and the FoxA1 sites overlapping with promoters.

Look at the results datasets, and how many regions are identified as occurring in promoters. Are any of the subsets (shared, ER unique, FoxA1 unique) more or less likely to occur in promoters than the others?

6 Other things to try

- In the original peak data, you may notice that one of the peak callers (MACS) called many fewer peaks for one replicate (Replicate 1) of one factor (FoxA1) than the other three peaksets for this factor. This is not uncommon to see, and the method of requiring every peak to be in every peakset for the consensus means that the “weakest” peakset can dominate the analysis. Re-do the analysis without this peakset. How does this change the results?
- Change the definition of Promoters to go as far as 10,000 basepairs upstream and include the first 5,000 basepairs of each gene. How does this change how many Shared sites overlap with promoters?
- Try overlapping the peaks with the RefSeq Genes instead of Promoters. How does this change the results?