MochiView

User Manual (v1.45)

Table of Contents

TIP: Click an item in the Table of Contents to jump to that page.

Getting Started

The information in this section is highly redundant with (and less complete than) the information offered on the MochiView website. It is recommended that you follow the instructions posted on the website (this is also where you will find the files):

http://johnsonlab.ucsf.edu/

Just click the 'MochiView' link at the top of the page and follow the instructions.

Requirements

Java

MochiView requires Java v1.6 (AKA Java *v6.0* or Java *SE 6*). Please see the website (http://johnsonlab.ucsf.edu/sj/mochiview-sysreq/) for detailed instructions on how to verify your Java version and upgrade if necessary.

Operating System

MochiView is written purely in Java, and can therefore be used with any operating system that supports the required version of Java. Unfortunately, this excludes older Macs. As of this writing, the only Macs with Java v1.6 support are 64-bit Intel-based Macs running OS X 10.5 or higher. OS X 10.5 Macs do not have v1.6 installed by default (OS X 10.6 does).

Memory

MochiView requests up to 1GB of memory from your system when running (mainly to allow caching of data for smoother performance), but typically reserves no more than 256MB. If your computer only has 1GB of memory, please consult the memory management instructions (http://johnsonlab.ucsf.edu/sj/mochiview-memory-management/) on the website for details on running the program with lower memory demand.

Hard Drive Space

MochiView stores data in a local database, and sometimes requires additional space for temporary files. The amount of required space depends on the amount of data imported, but as a guideline it is recommended that you have at least 5GB of hard drive space available.

MochiView Versions

For ease of use, MochiView is bundled differently for Macs than other platforms. Download the version appropriate to your operating system, and follow the instructions below to launch the program.

Launching the Program

PC Users

Extract the contents of the '`zip`' file anywhere on your computer. Everything required to run MochiView resides within the extracted folder.

Launch the program by clicking the file called '`MochiView.exe`'.

Mac Users

For Mac users, MochiView is bundled into an application, and can be run (after extracting the zip file) with a simple double-click. If it does not launch, there are two likely problems:

1. Your system does not have the most recent version of Java installed and/or set as the default (see above)
2. MochiView does not have adequate permissions to operate. Giving read/write/execute privileges to the contents of the application bundle is a possible workaround.

Linux Users

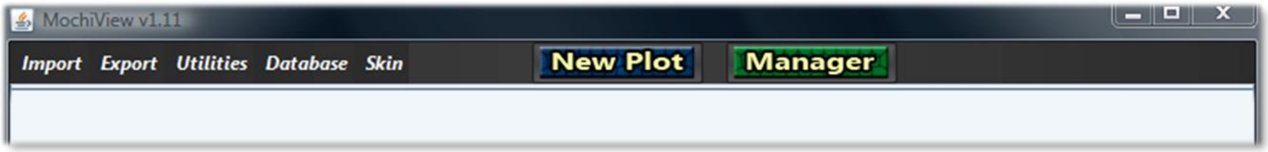
Run MochiView from the command line by navigating to the '`INTERNAL_USE`' folder and typing the following:

```
java -jar -Xms512m -Xmx1024m ChipView.jar
```

Tutorial

It is ***strongly recommended*** that first-time users get acquainted with MochiView by downloading and completing the tutorial. The tutorial is available on the website, and is written with the assumption that you have not read the manual beyond this point. Also, please note that if you are planning on displaying RNA-Seq or CGH data, you may find the brief guides in *Appendix A* and *Appendix B* of this manual useful.

Usage Overview



The top menu ribbon of the software provides access to three modes of use

There are three primary usage modes in the MochiView software:

1. Data Import/Export [Menu Bar]

The drop-down menus in the menu bar provide access to various data import/export utilities. These utilities are described in more detail later in the manual.

2. Data Update/Remove/Overview

The **MANAGER** view displays the contents of the database via a series of tables. A few basic tips:

- Tables can be sorted by `L-click`ing on the header (hold down '`Ctrl`' to set up a multi-column sort).
- A more detailed (and editable) description of any entry can be accessed by `L-click`ing the 📄 icon after selecting a row (or by `double-click`ing on the row).
- Provided that no plot tabs are currently open, you can delete database entries by selecting a row and then `L-click`ing the 🗑 icon at the top of the table. Multiple rows can be selected by holding down '`Ctrl`'.

3. Create Plot

The **NEW PLOT** menu allows the creation of a new data plot (multiple data plots can be open at any given time, each in their own tab). There are two basic types of data displayed in a plot: ***Primary Axis Elements*** and ***Track Axis Elements***. While both types of element make use of the X-axis (sequence coordinates), they differ in their use of the Y-axis:

PRIMARY AXIS ELEMENT (Data Sets, Tiled Sets)

These elements utilize the primary Y-axis of the plot to display data as either a line or bar graph. The term "primary y-axis" refers to the vertical axis that spans the entire height of the plot (and is used to distinguish this axis from the "track axis" described below).

TRACK AXIS ELEMENT (Data Sets, Location Sets, Tiled Sets, Motifs, Alignments, etc.)

These elements appear in a narrow *Track* drawn on the plot, and can either:

- Display data as a line or bar graph using a condensed y-axis (scaled to match the height of the *Track*). The numbers corresponding to the "track axis" are displayed on the right edge of the track.
- Consist purely of location information (i.e. lacking data to plot on a y-axis)
- Display data as text on the location marker

Data Categories

A basic understanding of the types of data stored in the MochiView database is essential for successful navigation of the MochiView software. An overview of the most important categories of data is provided below, followed by more detailed descriptions. (Throughout this manual, data categories are emphasized using bold text.)

CATEGORY OVERVIEW

CATEGORY	*DESCRIPTION*
Sequence Set	A collection (typically a genome) of **Sequence**s (typically chromosomes).
Location Set	A collection of **Location**s (e.g. array probes, genes, binding regions…) represented by coordinates on a **Sequence**. Each **Location Set** also has a **Location Type**, some of which have special properties when plotted (e.g. *Gene*, *Alignment Block*).
Data Set	A collection of values (e.g. $\log_2$ ratios, p-values, etc...) associated with one or more **Location**s in a **Location Set**.
Tiled Set	Resembles a **Data Set**, but is designed for large amounts of data associated with non-overlapping **Location**s of a uniform length.
Data Type	A description of the type of data associated with a **Data Set** or **Tiled Set**, accompanied by guidelines for plotting.
Motif	A DNA motif, typically represented by a frequency matrix.
Project	An optional means for subcategorizing **Location Set**s, **Data Set**s, and **Motif**s for organizational purposes.

SEQUENCE SET (and SEQUENCE)

Examples

Chromosome sequences, contig sequences, etc.

Description

A **Sequence Set** is a general annotation (*name*, *description*, ***Species***) applied to a group of **Sequence**s. Each **Sequence** is assigned a unique *name* and *nucleotide sequence*. Note that while there is no hard limit to the number of **Sequence**s allowed in a **Sequence Set**, the software is designed such that **Sequence**s are selected by pull-down menus, and thus having more than ~100 is not recommended.

LOCATION SET (and LOCATION)

Examples

Microarray probes, gene locations, binding sites, promoter regions, alignments, etc.

Description

A **Location Set** is the general annotation (*name*, *description*, **Location Type**, **Sequence Set**, and **Project**) applied to a group of **Location**s. Each **Location** consists of two inclusive coordinates (*start* and *end*) and a **Sequence** *name*. If the *start coordinate* is greater than the *end coordinate*, the **Location** lies on the complementary strand (note that this means that **Location**s of size 1bp are always considered to be on the plus strand). The first coordinate on a **Sequence** is '1'. Thus, the first five bases of the plus strand of a **Sequence** would be represented by a *start coordinate* '1' and an *end coordinate* of '5'.

Anything that can be plotted on a **Sequence** can be imported as a **Location Set**. Each **Location** in a **Location Set** can contain *Location Annotations* (a small 15 character *Annotation Tag* and a larger *Annotation Description*), which can be provided either during initial import or afterwards by using the '*Import/Update Location Annotations*' utility or by double-clicking individual **Location**s displayed in a *Track*.

LOCATION TYPE

Fixed Nomenclature

'Array Probe', 'Binding Region', 'Promoter Region', etc.

Description

Each **Location Set** is assigned a **Location Type** from a list of possible options. This designation serves primarily to aid the user in viewing/sorting data, but in the case of the **Location Type**s of '**Gene**' and '**Alignment**' this designation bestows additional properties. **Location Set**s assigned these types are imported through specialized import utilities and have additional annotation that other **Location Set**s lack (see import guide for details).

DATA SET

Examples

- Microarray expression ratios transformed to $\log_2$ and assigned to a '*Gene*' **Location Set**
- ChIP-Chip p-values ($-\log_{10}$ transformed) assigned to an '*Array Probe*' **Location Set**
- **Motif** scores assigned to a '*Binding Motif*' **Location Set**

Description

A **Data Set** is assigned a *name, description*, **Data Type**, **Project**, and **Location Set**, and contains a map of [**Location** → *Numerical Value*] for one or more **Location**s in a **Location Set**.

TILED SET

Examples

- Tiled read counts from a ChipSeq experiment
- Per-base values associated with a sliding window scoring a genomic feature (e.g. protein hydrophobicity within genes)

Description

A **Tiled Set** is assigned a *name, description*, **Data Type** and **Project**, and in many respects resembles a **Data Set**. A **Tiled Set** differs from a **Data Set** in the following ways:

- The underlying **Location**s must all have the same length and cannot overlap (i.e. they are tiles of a specific size)
- The underlying **Location**s have no strand assignment
- The **Tiled Set** is for plotting only, and is not available in MochiView's utilities or *Data Browser*
- The **Tiled Set** can display very large amounts of data without slowing plot performance
- The **Tiled Set** cannot be annotated or used in *Edit Mode* (described later)

Display limitations

Please be aware that plots of **Tiled Set**s at anything but close zoom will not be 100% accurate, because shortcuts are taken to pre-calculate the data at different zoom-levels (using either averages or maximum deviation from zero, as described for the import utility).

DATA TYPE

Examples

Log_2-transformed ratios, LOD scores, p-values, etc.

Description

Each **Data Set** and **Tiled Set** is assigned a **Data Type**, which provides information about the preferred Y-axis range for plotting of the **Data/Tiled Set** and annotation (*name*, *description*) of the type of data (e.g. -$\log_{10}$ P-values, $\log_2$ ratios, and LOD scores).

VALUE TYPE

Fixed Nomenclature

'*Integer*', '*Decimal*', or '*Boolean*'

Description

Each **Data Set** is assigned a **Value Type**. This designation is primarily for internal use (it influences error-checking during data import and database storage), but note that:

1. '*Decimal*' **Value Type**s support values from -99,999 to 99,999 and are precise out to three decimal places. Attempts to import values that do not fall within this range will result in an error (try log-transforming such data).
2. '*Integer*' **Value Type**s support values from -2,147,483,648 to 2,147,483,647.
3. Only '*Boolean*' **Data Set**s can be used in *Edit Mode* (described later). These values are imported as either `1` (i.e. "true") or `0` (i.e. "false").

PROJECT

Description

Each **Location Set**, **Data Set**, **Tiled Set**, and **Motif** can be assigned to a specific **Project**, or assigned to no **Project** at all (called *'Global'*). **Project**s provide additional annotation to assist in the sorting/viewing of data. In addition, the **Project** influences the text of the plot legend: when all plotted **Data/Tiled Set**s belong to the same **Project**, the **Project** name is omitted in the legend. Otherwise, the legend displays the **Project** name followed by the **Data/Tiled Set** name.

SPECIES

Fixed Nomenclature

Many Species names as well as a catch-all of '*Other*'. (Let me know if you'd like to see a species name added!)

Description

The **Species** designation is assigned to both **Sequence Set**s and **Motif**s, and serves primarily to assist in sorting data in the **NEW PLOT** and **MANAGER** menus. However, **Species** is also taken into account when determining the appropriate codon table for a plot. (Specifically, MochiView correctly represents the atypical usage of '`CUG`' as a serine codon in several *Candida* species. Let me know if you'd like any others species-specific changes added.)

MOTIF

Examples

- Motifs from existing libraries (see the MochiView website)
- Frequency matrices from motif-identification software such as MEME[1] or BioProspector[2]
- Binding affinity matrices from MatrixREDUCE[3,4]

Description

A **Motif** is stored in the database as either a Position-Specific Frequency Matrix (PSFM) or a Position-Specific Affinity Matrix (PSAM). When plotted, a sliding window scores the matrix against the sequence. Several utilities are available via the menu bar for further analysis/manipulation of imported **Motif**s. For more information on **Motif**s, consult the section titled *Scoring and Identifying Motifs*.

MANAGER Menu (exploring the active database)

Click the Manager button ('Alt+M' hotkey) at the top of the screen to open this menu. The **MANAGER** contains tabs corresponding to categories of data in the database (reviewed above). Each tab contains a table in which the rows correspond to items in the database.

Exploring the contents of the database

Navigating the tables

Please consult the section titled *Common Menu Elements* for information on how to sort tables and select/deselect items in MochiView tables (both use common keyboard/mouse conventions).

The tables in the **MANAGER** (and in most other menus as well) can be sorted by `L-click`ing on the header. Click multiple times to cycle through *ascending*, *descending*, and *disabled* sorting (a triangle icon in the header will indicate the current sort status). You can hold down 'Ctrl' and `L-click` multiple headers in turn to set up a multi-column sort.

Searching the tables

If your table contains a large number of rows, you may want to search for rows matching the name of an item of interest (e.g. searching for **Motif**s with 'TUP' in their name). This can be accomplished by opening the search menu using the 'Ctrl+F' hotkey. You will see a search menu initially populated with every name in the currently viewed table. As you type in your search term the list will be culled to only show matching terms. `L-click` on a matching term to close the search menu and scroll (if needed) to the row containing the item. The text of the selected item will be displayed in bold.

Editing items in the database

Selecting a row and `L-click`ing the 📝 icon (or just `double-L-click`ing the row) in the **MANAGER** opens up a new window that often contains additional annotation information (sometimes in the form of tooltips). This window also offers the ability to edit various aspects of the item's annotation (e.g. name, description, assigned **Project**). For certain tabs (**Location Set**, **Data Set**, **Tiled Set**, and **Motif**), you can select multiple rows (hold down 'Ctrl' and `L-click` multiple items or hold down 'Shift' and `L-click` to select an interval) and then `L-click`ing the 📝 icon to set the **Project** of all selected items. (With the exception of the **Motif** tab, all selected items must belong to the same **Sequence Set**).

Deleting items in the database

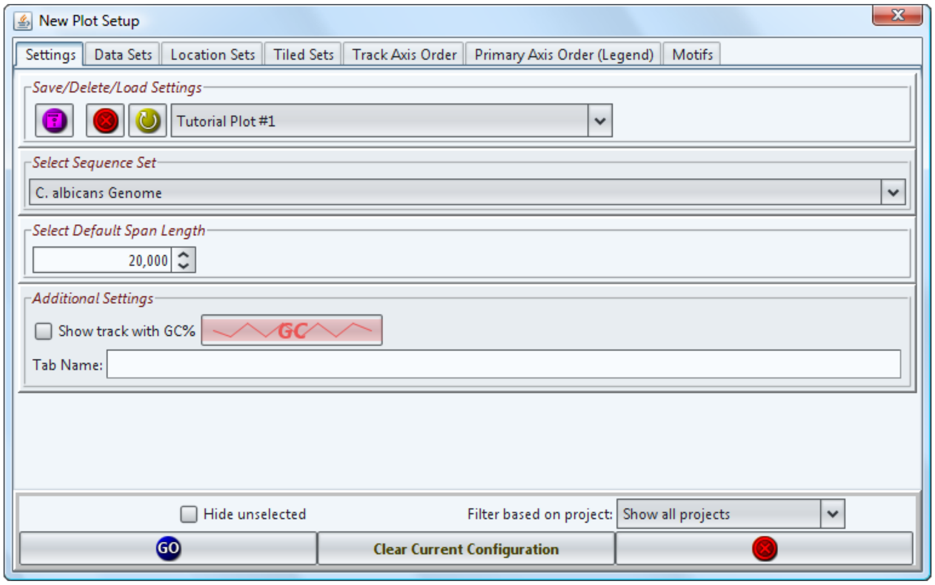
When one or more rows in a table are selected (select multiple rows using the standard 'Ctrl' and 'Shift' conventions) the 🗑 button becomes highlighted. You can click this button to delete the items from the database. Importantly, all other items in the database that are associated with the deleted items will also be deleted! For example, if you delete a **Project**, all items assigned to that **Project** will be deleted as well (you will be warned with a confirmation box notifying you that additional items will be deleted).

NEW PLOT Menu (configuring a chart)

The **NEW PLOT** menu facilitates the configuration and display of plots. Multiple plots can be open in MochiView at any given time – each is displayed in a separate tab. The menu is opened by clicking the **New Plot** button at the top of the screen ('`Alt+P`' hotkey). If you already have plot tabs open, the menu will be configured to display the configuration of the currently selected plot tab (you can clear this configuration using the button at the bottom of the menu, as described below).

If you would like to adjust the settings of a currently opened plot tab, you can use the '`Alt+Q`' hotkey (or the equivalent item in the plot's `R-click` menu) to simultaneously close the tab and open the **NEW PLOT** menu with the plot's configuration in place. You can then make adjustments and click the ● button to re-open the tab.

The **NEW PLOT** menu contains a lower panel of filter options and settings in addition to multiple tabs containing configuration options. These are described in order below.



Save/Delete/Load Settings: Tutorial Plot #1

Select Sequence Set: C. albicans Genome

Select Default Span Length: 20,000

Additional Settings

□ Show track with GC%

Tab Name:

□ Hide unselected

Filter based on project: Show all projects

The NEW PLOT menu ('Settings' tab is showing)

Lower Panel (menu filters and buttons)

'Hide Unselected' checkbox

When checked, the configuration tables in the *Data Sets*, *Location Sets*, and *Motifs* tabs will only display the rows in their respective tables that contain items selected for display in the plot. This box is automatically checked when you load saved settings (see '*Settings*' tab below).

TIP: *If you find yourself wondering what happened to the contents of your database, check whether you have filtering in place.*

Filter Based on Project

This pull-down menu can be used to filter the tables in the *Data Sets*, *Location Sets*, and *Motifs* tabs so that only those rows containing the selected **Project** (or a *Global* **Project**) are displayed.

Clear Current Configuration

Clicking this button will deselect all items in your current plot configuration (since most plots will contain a gene track, there is an option in *Utilities →Preferences* to add a gene track upon clearing).

Submit and Cancel buttons

The ✓ button launches a plot containing using current configuration. The ⊗ button closes the menu without opening a plot.

'Settings' Tab

Save/Delete/Load Settings

The current settings in the **NEW PLOT** menu can be saved 💾, deleted ⊗, or loaded ↻ using this panel. Every time you launch the program, the default mode is to load the last saved settings (and hide unselected data).

Select Sequence Set

Only those **Data Set**s, **Tiled Set**s, and **Location Set**s belonging to the currently selected **Sequence Set** are displayed in the corresponding tabs.

Select Default Span Length

This setting determines the default sequence length to be displayed on the X-axis. Depending on the setting chosen in *Utilities→Preferences*, this default can be re-enforced every time a new region is selected (i.e. using the *Gene*, *Data*, *Sequence*, or *Location* browsers).

Additional Settings

Show *Track* with GC%

When selected, the plot will include a *Track* displaying the '`GC`' content as a percentage (see the section of the manual on '*Track Styles*' for details). The color and height of the *Track* can be adjusted by clicking the button next to the checkbox.

Tab Name

The entered value will be displayed in the plot tab (useful if multiple plots are opened at once).

'Data Sets' Tab

(This tab is searchable using the search menu (`Ctrl+F`) *described in the section on the* **MANAGER)**

This table displays all **Data Set**s for the currently selected **Sequence Set** (chosen in the '*Settings*' tab). The '*Hide/Show Unselected*' button below the table can be used to toggle the visibility of **Data Set**s that currently have no assigned *Primary* or *Track Axis Elements*. Each row contains annotation associated with the **Data Set** plus three adjustable columns:

Primary Axis Element (*Line* and *Bar Styles*)

Check the box to display the **Data Set** using either ***Line Style*** or ***Bar Style*** with y-values mapped to the primary y-axis of the plot. To customize the settings click on the representation of the ***Line/Bar*** next to the checkbox. This brings up a menu that allows customization of line color and style.

Track Axis Element (*Shape*, *Line*, and *Bar Styles*)

Check the box to display the **Data Set** in a *Track*. To customize the *Track* settings click on the representation of the ***Track*** next to the checkbox. The *Track* customization menu provides several additional options:

Track Height

This number represents the % of available vertical plot space assigned to the *Track*. (If the plot contains elements mapped to the *primary axis*, some space is reserved for display of these features). For a detailed explanation of track heights please consult the section titled *Using the Track Height/ Title Manager.* Note that you can also adjust the height of all tracks while viewing a plot using the `Ctrl+H` hotkey.

Select Track Style

A pull-down menu allows selection of several different plot styles (see the section of the manual titled '*Track Styles*' for details).

***Gradient Mode* (Shape Style only)**

By default, all **Location**s displayed in a *Track* are represented by the chosen *Track* color. However, it is sometimes desirable to color-code individual **Location**s by an associated value. Any **Data Set** associated with the same **Location Set** as the base **Data Set** (the one in the table row currently being customized) can be selected to override the default color using a color scale ranging through user-defined minimum, middle, and maximum values and colors. Note that the text representation of the value will still reflect that of the base **Data Set**. This feature is particularly useful for associating a color scale with gene expression data tied to a gene **Location Set.**

***Edit Mode* (Shape Style only)**

This feature (described in the tutorial) allows the dynamic adjustment of a **Data Set** of **Data Type** '*Boolean*' by clicking on the *Track* **Location**s in the plot window. Clicking cycles through '`true`' (YELLOW),'`false`' (BLUE), and '`unselected`' (DARK GREY). The shape's `right-click` menu also offers the option to select from these choices. This feature is particularly useful for manual curation of predicted binding sites. The resulting **Data Set** can then be applied as a *data filter* (described in the section titled *Common Menu Elements*). If your plot contains a single *Edit Mode Track* and you have the detailed data browser open to display the underlying **Location Set**, each row in the data browser table will be color-coded to reflect the current *Edit Mode* state of the **Location** represented by the row.

Y-Axis Coordinate Location (Shape Style only)

When selected, the *Track* is bound to the primary axis at a y-value of zero. The most common use of this feature is to bind genes to the `y=0` axis so they are more easily viewed in conjunction with *primary axis* data.

Data Filter

The final column indicates whether any *data filter*s are currently attached to the **Data Set**. Clicking on cells in this column brings up a filter configuration menu. Filters are described in detail later in the manual in the section titled *Common Menu Elements*. The buttons in the filter column indicate the current filter state:

⊕: Currently no filters… click to add a filter
✓: One or more filters are currently in place… click to modify/view the current configuration
⊗: No filter permissible (e.g. a **Location Set** with no associated **Data Set**s)

Using *data filter*s incurs a speed/memory cost as you navigate your plot, so their use is not recommended for large **Location Set**s (> ~250,000 **Location**s).

'Tiled Sets' Tab

(This tab is searchable using the search menu (`Ctrl+F`*) described in the section on the* **MANAGER)**

Each row in the table in this tab corresponds to a **Tiled Set** of the currently selected **Sequence Set**. The description of the **Data Set** tab generally applies to this tab as well, with two exceptions:

1. Filters are not available for **Tiled Set**s
2. Some track configuration options (e.g. *Edit Mode*) are not available for **Tiled Set**s

'Location Sets' Tab

(This tab is searchable using the search menu (`Ctrl+F`) described in the section on the **MANAGER)**

Each row in the table in this tab corresponds to a **Location Set** of the currently selected **Sequence Set**. The description of the **Data Set** tab generally applies to this tab as well, with three exceptions:

1. **Location Set**s can only be displayed as ***Track***s.
2. Only **Location Set *Track***s display *Location Annotations* (**Data Set *Track***s display data values instead)
3. Filters applied to **Location Sets** can be declared as '*Global*' (see section below).

'Track Axis Order' Tab

This tab allows the user to select the order (from top to bottom) in which *Track*s are plotted. The table displays all *Track*s that are currently enabled in the *'**Data Sets**' tab* (blue text), *'**Tiled Sets**' tab* (purple text), *'**Location Sets**' tab* (red text), and ***'Motifs'*** tab (green text) with the exception of those *Track*s bound to the '`y=0`' axis. To change the *Track* order, select a *Track* (`L-click` and hold) and drag it to the desired position. Most *Track*s of similar type (e.g. line, bar, motif, shape) can be joined (select multiple *Track*s by holding down '`Shift`' and `L-click`ing them and then click the join button). Locations in joined *Track*s are plotted on a single *Track*. This arrangement is particularly useful for *Motif Tracks* (and is the default for these tracks) as well as for *Tiled Set Tracks* of strand-specific data (e.g. ChIP-Seq or RNA-Seq) in which one track uses positive values and represents the plus strand and the other uses negative values and represents the minus strand.

'Primary Axis Order (Legend)' Tab

This tab allows the user to select the order (from top to bottom) in which *Primary Axis Elements* are displayed in the legend text of the plot. Display order is adjusted as described for the '*Track Axis Order*' tab.

'Motifs' Tab

(This tab is searchable using the search menu (`Ctrl+F`) described in the section on the **MANAGER)**

This tab allows the selection of **Motif**s that will be scanned against the **Sequence** region displayed in the chart. Any region that passes the user-selected cutoff score (click on the cells in the '*Score Cutoff*' column to adjust) will be represented on a *Track* in the chart with a text representation of the score. By default, all selected **Motif**s are placed on a shared *Track* (they can be split in the '*Track Axis Order*' tab). A few additional comments:

- PSFM **Motif**s are scored against a background model specified in the '*Utilities→ Preferences*' menu.
- The graphical representation of the PSFM **Motif**s is adapted from the concept of Sequence Logos and the WebLogo[5] implementation (http://weblogo.berkeley.edu/). The PSAM **Motif** logos are based on the approach described by Foat *et al.*[4].
- Clicking on the **Motif** logo will display the reverse complement logo.
- Selecting a large number of **Motif**s for scanning (>~20) is not advised, as the program will slow significantly.

For more information on **Motif**s, consult the section titled "*Scoring and Identifying Motifs*".

Plot Window Overview

Once you have configured your plot settings in the **NEW PLOT** menu and clicked the '*Create Plot in New Tab*' button, a new tab will be created and your plot will begin to load. By default, the first region displayed is the beginning of the first **Sequence** (sorted alphabetically by name). The elements of the plot window are summarized below:

Legend

The legend is displayed in the top-left corner of the plot. If both *Primary* and *Track Axis Elements* are displayed, these entries are separated by a small divider. Moving the mouse over the legend text will display a Tooltip with further information, and will also highlight the corresponding *Element***.** Clicking on the legend text for a *Primary Axis Element* toggles display of that *Element* on/off (as represented by an '`X`'). The legend itself can be hidden using either the `R-click` menu or the '`Alt+N`' hotkey. By default, the menu resizes to try to avoid overlapping tracks. You can override this and keep the legend full size using either the `R-click` menu or the '`Alt+H`' hotkey

No legend entry is made for the gene ***Track*** (if bound to the primary y-axis).

Primary Axis Elements

(Described in the 'Usage Overview' section of this document.)

Tooltips can be accessed by moving the mouse over individual data points. *Primary Axis Elements* are displayed in either *Bar* or *Line Style*. These styles are similar to the *Bar* and *Line Style*s described in the '***Track Styles***' section of the document, with the exception that they utilize the primary y-axis and that individual data-points give tooltips (provided you are not zoomed out too far).

Track Axis Elements

(Described in the 'Overview' and 'Track Styles' sections of this document.)

A semi-transparent background is displayed for each *Track*, and can be used to access a tooltip description of its contents. Each *Track Axis Element* has a descriptive title that can be hidden using either the `R-click` menu or the '`Alt+T`' hotkey.

Information Ribbon

The bottom-left corner of the software window is reserved for displaying text feedback. Types of feedback provided include:

- **Location** coordinates when moving the mouse over a **Location**
- Display of width and **Location** coordinates when moving or resizing the display area
- Notification that dynamic **Motif** scanning is complete for the displayed area
- Notification that sequence has been copied to the clipboard (see below)

The bottom-right corner of the software window indicates whether the plot is still loading data.

Interacting With the Plot

Browsing the Sequence Set

There are numerous options available for browsing the **Sequence Set**. These are described in the section titled *Plot Navigation*.

Tooltips

Most items in the plot provide tooltips with additional information when the mouse if hovered over them.

Sequence → Clipboard

The sequence of any **Location** displayed in the plot window can be copied to the clipboard by `L-click`ing on the **Location**. Holding down '`Ctrl`' while clicking copies the complementary sequence, and holding down '`Alt`' while clicking prevents the inclusion of a FASTA-style header row that contains the sequence coordinates. Holding down '`Shift`' while clicking copies a fragment centered on the midpoint of the location (the fragment size is specified in *Utilities →Preferences*).

Summary of Mouse Actions in the Plot Window

Details for mouse-interactivity are provided throughout the manual, but are summarized here for convenience:

Legend

- `L-click` on the text for a *Primary Axis Element* legend item to toggle it visible/hidden in the plot.
- Hover the mouse over any item in the legend to highlight it in the main plot

Navigation

- `L-click` and drag to pan along the x-axis of the plot
- `L-click` and/or drag in *Location Mini-Browser* to move the plot window along the current **Sequence**
- `L-click` on the **Sequence** length display in the *Location Mini-Browser* to open a **Sequence** selection menu
- Use the mouse wheel to zoom in/out on the plot x-axis
- '`Shift`' + `R-click` on a region of the plot to zoom in to a sequence-level view centered at the x-axis value at the point clicked.
- '`Alt`' + `R-click` and drag across a region of the plot to zoom in to the region. (*Mac users*: hold both '`Option`' and '`Command`' and `L-click`.)

Popup Menus

- `R-click` on the plot (outside of any *Tracks*) to access the main plot menu
- `R-click` on a gene to access the option to copy protein sequence to the clipboard (when available) or to launch the web browser gene entry (when a URL has been provided in *Utilities →Preferences*)
- `R-click` on a *Track* containing a line or bar plot to see *Track*-specific options (e.g. rescaling y-axis or changing line format)
- `R-click` on a *Track* containing an alignment to toggle between showing all bases and only showing those that diverge from the reference genome.

Interactive Tracks (all of these options are also available through R-click menus)

- *Edit Mode Track* **Location**s can be toggled between on/off/undecided by `L-click`ing
- **Location**s in *Track*s that are displayed in *Shape Style* can be `double-L-clicked` to edit the *Location Annotation* (with the exception of **Location Set**s of the **Location Type** '*Alignment Block*').
- **Motif** *Track* hits can be `double-L-clicked` to open a visual comparison between the **Motif** and the matched sequence.
- `L-click`ing on a **Location** of a *Track Axis Element* (or *Primary Axis Element*) will copy the sequence of the **Location** to the clipboard.
- '`Alt`' + `R-click`ing on the gene name text of a **Location** of **Location Type** '*Gene*' will launch your browser with the *feature name* of the gene appended to the URL prefix provided for the current **Sequence Set** in *Utilities →Preferences*. (*Mac users*: hold both '`Option`' and '`Command`' and `L-click`.)

Using the Track Height/ Title Manager

Usage

You can open the '*Track Height/Title Manager*' in any plot tab using either the `R-click` menu or the '`Ctrl+H`' hotkey. This manager allows you to fine-tune the heights of the items in your plot as well as rename the track titles. Note that you can also specify track heights as you configure your plot in the **NEW PLOT** menu, but this menu allows you to make global adjustments and quickly observe how they change the appearance of the plot.

How does it work (AKA wow, this is confusing)?

The apportionment of the vertical space of the plot is quite complicated because it is split between multiple different types of plot features, many of which are only present in a subset of plots. (MochiView thinks of the available vertical space as a percentage rather than a specific number of pixels, thus things rescale as you resize the window.) The rules for apportioning space are enforced in the order presented below (i.e. each item gets dibs on the remaining available space):

[1] Space for displaying sequence upon zoom

A minimum of 15% of the space at the top of the plot is reserved for displaying DNA sequence when the plot is sufficiently zoomed. If the primary axis is in use, primary axis elements can also use this space (the sequence is then juxtaposed on the y=0 intersection of the primary y-axis).

[2] Space for track titles and the gap between tracks

A small amount of space is reserved for each the space between each track. If you are displaying track titles (toggle with the '`Alt+T`' hotkey) and the titles are being displayed above the tracks (adjustable in the '*Preferences*' menu), then additional space is reserved for these titles. This space is *NOT* considered part of the track heights.

[3] Space for displaying primary axis data and *Shape Style* tracks bound to the primary axis

Typically primary y-axis data will consist of some line/bar graphs and possibly a gene set pinned to the y=0 axis. The plot reserves sufficient space for the y-axis to fulfill the following requirements:

1. The y=0 axis is above all tracks
2. The primary y-axis observes the preferred minimum and maximum values associated with the **Data Type**(s). This includes the 'Keep **Data Type** preferred y-min above all tracks' option if it is selected in the '*Preferences*' menu.
3. There is room to display any *Shape Style* tracks bound to y=0 of the primary axis according to their track heights

[4] Space for tracks

Once all of the above conditions are made, the remaining space is made available to the tracks based on their individual heights. Recall that if multiple items are bound to the same track, the track height is calculated as the largest of the heights of the individual items.

OK, how does this relate to what I see in the Track Height/Title Manager?

The manager can be divided into three sections:

[1] TOP SECTION: Axis-bound track height (only visible if such tracks exist)

If you have *Shape Style* tracks bound to the y=0 axis (typically a gene track) you can set the height of these items using this value. ***The value can be considered the percentage of the plot vertical space that should be reserved for these items*** (provided that this much is available once all higher-priority space requirements are met).

[2] MIDDLE SECTION: Track heights (and editable track titles)

These values are ***the percentage of the plot vertical space that should be reserved for these items.*** If the sum total of these items exceeds or equals the available space (you can find out the available space by looking at the bottom item) then the tracks fill all available space and the heights are simply apportioned relative to each other. If the sum total is less than the available space, the remaining space is apportioned to primary axis data (or is blank space if no primary axis data is being displayed).

[2] BOTTOM SECTION: Track height rescale and current track height total

As noted above, if you want your tracks to occupy less than the available space (in order to give more of the plot space to your primary axis data), the sum of their heights must be less than the available space. This bottom section tells you the current sum of track heights, the available space, and provides a box where you can recalibrate the track heights to a new sum (maintaining their values relative to each other, with the caveat that no track height can go lower than `1`).

Snapshot Utility

This utility can be launched from any plot from the *R-click* menu (or by typing '`Ctrl+S`').

Usage

This utility allows the user to select a **Location Set** and export plot images centered on the **Locations** within the set. The basic configuration of the plot is preserved for each image (*i.e. Primary* and *Track Axis Element*s, visibility of legend/gridlines/*Track* titles, etc.). The menu provides options to configure several additional aspects of the exported image(s):

Optional Data Set

The user can choose a **Data Set** associated with the chosen **Location Set**. Doing so has three effects:

1. Images are only created for **Location**s in the **Location Set** that have a value associated with them in the **Data Set**
2. If '`pdf`' output is chosen, the associated value is displayed in the header above the plot image
3. The order in which the images are output can be sorted based on value

Location Constraint

This submenu (see tooltips) provides several options for determining the sequence width displayed in each plot image.

Sort Method

This submenu (see tooltips) provides several options for determining the order in which the **Location**s from the selected **Location Set** are output.

Page Layout (tab #2)

This submenu (see tooltips) allows configuration of the width/height (in pixels) of the images produced. The default allows the inclusion of 5 plots per page in the '`pdf`' output.

Print Type: vector or raster (tab #2)

Vector format saves each plot in a format that can be opened and manipulated in Adobe Illustrator (or similar programs), and also provides infinite scalability. Raster format saves plots as '`png`' files, which are not scalable. Vector is recommended… the only advantage to the raster format is that certain plot elements that use color gradients will not look as nice (the gradient is not supported and is removed). A quick tip, if you want to edit the '`pdf`' files in Adobe Illustrator, you have to remove a lot of clipping masks. Do this quickly in Illustrator by selecting everything ('`Ctrl+A`') and then pressing (`'Alt+Ctrl+7`') numerous times. This will strip away the clipping masks.

Print Type: plot-per-file or all-in-one-file (tab #2)

Images can either be saved as individual files (you will be prompted for the directory, and the files will be labeled '`SNAPSHOT_00001`', '`SNAPSHOT_00002`', etc.), or as a single '`pdf`' file (you will be prompted for the name). The all-in-one-file format adds a header that includes the following information:

- The value associated with the **Location** (if an optional **Data Set** is selected)
- The **Sequence** name and coordinates for the **Location**
- Names of the genes in the visible region (up to 5 max)

Filters (optional; tab #3)

Additional filters can be applied to the selected **Location Set** to reduce the number of plots created (filtering is described in the section titled *Common Menu Elements*).

Gene names (optional; tab #4)

If you selected a **Location Set** of **Location Type** *Gene* in tab#1 you have the option of restricting the **Location Set** to specific genes. Just paste the names in this tab. Note that in this case filters (tab#3) will be ignored.

Tips for Plot Presentation

When making large plots (in terms of dimensions), the saved images will look almost exactly like they do on your screen. However, smaller plot sizes can potentially run into issues where displayed text overlaps and/or shrinks and becomes unreadable. Here are a few recommendations for making nice looking plots:

- Try playing around with the height of your *Track*s (`Ctrl+H`).
- Try turning off the legend, grid lines, and *Track* titles in your plot.
- Experiment with configurations that do not utilize the primary axis element.

Plot Navigation

There are five modes of plot navigation in MochiView.

1. Pan/Resize

Access

Mouse, keyboard, or the `R-click` menu.

Usage

The mouse can be used to grab-and-pan the plot along the x-axis and the mouse-wheel can be used to zoom in/out on the plot x-axis. In addition, there are numerous hotkeys for zooming, scrolling, and paging along the x-axis (see section titled *Keyboard Commands*). For example, arrow keys can be used to scroll the x-axis along the **Sequence** (←/→) or expand (↑) or contract (↓) the display width.

2. Data Browser

Access

The *Data Browser* exists in three modes: *hidden*, *ribbon* (at top of plot), and *detailed* (right side of plot). You can toggle between the three modes with the '`Alt+D`' hotkey, and you can configure which mode is shown by default in *Utilities→Preferences*.

Usage

This is the most powerful navigational tool in MochiView, allowing the user to jump to **Location**s in any **Data Set** (sorted by value) or **Location Set** (sorted by **Sequence** position) of the currently selected **Sequence Set**. The only exception is that **Data Set**s of **Value Type** *Boolean* are not available in the *Data Browser*.

Ribbon view

This version of the *Data Browser* lacks many of the features of the *detailed* view (most notably the abilities to browse **Location Set**s), but takes up less screen real estate. If you are typically navigating small **Data Set**s this browser may suit your needs.

The **Data Set**s (chosen in the 2nd pull-down menu) are sub-categorized by **Project** (chosen in the 1st pull-down menu). The sort order of the **Location** values associated with the **Data Set** is chosen using the 3rd pull-down menu. The final pull-down menu displays the sorted values/**Location**s associated with currently selected **Data Set**. Each entry in this final menu displays:

- The associated value
- The closest gene - more specifically, the gene with the closest start coordinate to the midpoint of the **Location**. (The genes are obtained from the default **Location Set** of **Location Type** *'Gene'* that was selected in *Utilities →Preferences* at the time the plot tab was opened).
- The **Location** coordinates

These entries can be navigated using the pull-down menu and GO button or by using the '`Alt+←`' and '`Alt+→`' keyboard commands.

Detailed view

This version of the *Data Browser* offers several additional features that are unavailable in the *ribbon* view:

1. Browse **Location**s in a **Location Set**

2. Jump to a specific **Data Set** value or list position
3. Quickly page (in increments of 100 items) through the set items

The *detailed* browser is heavily supported by tooltips… mouse-over the various section headers for information. The current set items are displayed in the table in the lower section of the browser, and can be navigated using the '`Alt+←`' and '`Alt+→`' keyboard commands (as with the *ribbon* view).

Keep in mind…

There are two important things to keep in mind when using the *Data Browser*:

1. Each plot tab has a separate *Data Browser* that reflects the contents of the database at the time the tab was opened. Thus, imported data are not represented in currently open plot tabs.
2. **Data Set**s of **Value Type** *Boolean* cannot be browsed.

3. Gene Browser

Access

This browser is accessible either by clicking the Gene button in the upper-right portion of the plot window or by using the '`Alt+G`' keyboard command.

Usage

This option centers the current plot on a gene, using the primary **Location Set** of **Location Type** '*Gene*' that was assigned in *Utilities → Preferences* at the time the plot tab was opened. In the top-right corner of the plot window is a text field for entering gene names. When searching for genes from the database, the search queries *Feature Names* and *Gene Names*. If no matches are found, *Aliases* are searched. If a unique match is found, the gene browser closes and the plot immediately centers on the gene. Otherwise, a table containing all matches is provided and the user may select the desired gene. Note that the search is case-insensitive.

Wild-cards in the form of asterisks (`*`) are supported. For example, a search for `*c*5` would match `cdc5`, `ecm25`, and `sec5`.

4. Sequence Browser

Access

This browser is accessible either by clicking the Seq. button in the upper-right portion of the plot window or by using the '`Alt+S`' keyboard command.

Usage Overview

This browser identifies all matches to a sequence and displays them in a table at the side of the plot window. The plot can then be re-centered around the matches by clicking on the table rows (and zoomed to 100bp by `double-L-clicking`). Using the options in the menu, the search can either be:

1. Constrained to the currently viewed region
2. Constrained to a specific **Sequence**
3. Cover an entire **Sequence Set**

Your search can incorporate any of the *IUPAC* base symbols:

A	adenine	**M**	A,C	**K**	G,T
C	cytosine	**R**	A,G	**V**	A,C,G

G	guanine	W	A,T	H	A,C,T
T	thymine	S	C,G	D	A,G,T
N	anything	Y	C,T	B	C,G,T

Note that a search term with the IUPAC symbol 'V' will match 'A', 'C' and 'G', as well as 'M', 'R' and 'S' (because the latter three symbols contain a subset of the bases allowed by 'V').

Usage Mode#1: STANDARD (IUPAC and optional variable length tags)

This is the default usage mode, allowing a search for *IUPAC* symbols and, optionally, the inclusion of tags indicating acceptance of a variable number of a base. The tags must follow an *IUPAC* symbol, and take the following form:

```
<base>{<min>,<max>}
```

Example

Consider the following search term: `ACM{3,6}TT`. This term will match any sequence beginning with 'AC', followed by between 3 and 6 'A's or 'C's, and ending in 'TT'. Note that in cases such as this one where a more complex *IUPAC* symbol is used, the variable length region can be all 'A's, all 'C's, or a combination of the two letters.

Variable length searches and overlapping matches

NOTE: This is a level of detail that is unnecessary unless you need to understand the precise implementation.

MochiView slides a window from left-to-right along the sequence and searches for a single match in each window. It is possible that a variable-length search term will match at multiple lengths (consider `T{1,5}` in a long string of 'T's). In such cases, MochiView will only retain the greedy (in regular expression terms) match that begins from the left side of the window and will then shift the window and search again. If '*allow overlap*' is selected, the window shifts by one base, otherwise the window shifts beyond the matched sequence.

Usage Mode#2: GAPPED REPEATS (direct and inverted repeats with optional gap)

This mode takes the entered search term and searches for either a direct or inverted repeat of the term separated by a gap of user-defined minimum and maximum length. The search term may include any *IUPAC* symbol, but may not include variable length tags.

Example

Consider the search term `ACMTT` with a minimum gap length of '0' and a maximum gap length of '3'. If the user searches for a direct repeat, this term will match sequences such as 'ACATTACATT' (no gap) or 'ACCTTGAACCTT' (gap of 2bp). If the user searches for an inverted repeat, this term will match sequences such as 'ACATTAATGT' (no gap) or 'ACCTTGAAAGGT' (gap of 2bp). Note that matches to non-specific symbols must be the same for both repeats (e.g. if 'M' equals 'A' for one half of the repeat, it must also equal 'A' for the other half).

Variable length gaps and overlapping matches

NOTE: This is a level of detail that is unnecessary unless you need to understand the precise implementation.

Unlike the variable length tags from '*Standard Mode*', a search will match all qualifying gap sizes if the user chooses to '*allow overlap*'. If this option is not selected, MochiView moves the search window (sized to the maximum possible match size) base-by-base from left to right, keeps the largest match that begins from the left side for any match, and then advances the window beyond the matched hit.

Usage Mode#3: DETAILED REPEATS (complex partial direct and inverted repeats)

This mode is rather complex, but is one of the most powerful, allowing the user to require only a subset of bases in a search term to be associated with a repeat. The search term may include any *IUPAC* symbol, but may not include variable length tags. In addition, the search term can utilize the character 'P' to specify bases in the search term that must be part of a repeat. The search term is treated as a symmetrical reflection, such that bases are paired as follows:

DIRECT REPEAT:	'1234512345' (even)	'12345x12345' (odd)
INVERTED REPEAT:	'1234554321' (even)	'12345x54321' (odd)

Note that in the case of an odd number of bases the middle character is not grouped (if it is assigned a 'P', it is simply treated as an 'N'). Each group in the search term is considered as follows:

1. If neither base in the group has a 'P', the base is treated normally.
2. If both bases contain a 'P', then both bases must either be the same (direct repeat) or complementary (inverted repeat).
3. If one base contains a 'P' and the other contains another *IUPAC* letter, the same rules as described in '2.' apply, except the allowed range of bases is constrained by the letter.

Example

Consider the following search term: PCMTTPGPP.

If the user searches for a direct repeat, the middle 'T' is treated normally, and the remaining base pairings are as follows: 'P→P', 'C→G', 'M→P', and 'T→P'. Groups #2 and #4 are unambiguous, since group#2 is treated normally and group#4 requires that the 'P' be a 'T'. Group#1 requires that the two 'P's be the same, and group#3 requires that the 'P' be either an 'A' or a 'C', depending on what the 'M' matches. Examples of valid matches include 'ACCTTAGCT' or 'GCATTGGAT '.

If the user searches for an inverted repeat, the middle 'T' is treated normally, and the remaining base pairings are as follows: 'T→P', 'M→G', 'C→P', and 'P→P'. Groups #1, #2, and #3 are unambiguous, since group#1 requires that the 'P' be an 'A', group#2 is treated normally, and group#3 requires that the 'P' be a 'G'. Group#4 requires that the two 'P's are complementary. Examples of valid matches include 'CCATTAGGG' or 'TCCTTAGGA '.

5. Location Browser

Access

This browser is accessible either by clicking the Loc. button in the upper-right portion of the plot window or by using the 'Alt+L' keyboard command.

Usage

This browser allows the user to jump to specific coordinates on a chosen **Sequence**, provided the selected span is between 20 and 10,000,000 bases in length. Upon opening the menu, the currently selected region is graphically represented in green at the top of the menu. There are three ways in which the menu can be used to change the currently selected region:

[1] Enter Location as simple text entry

The browser provides a text field in which you can type or paste a **Location** entry containing the **Sequence** name, first coordinate, and second coordinate (it doesn't matter which is the larger coordinate, and the numbers can contain commas). The name and first coordinate can be separated by either a colon or whitespace, and the two coordinates can be separated by either a hyphen or whitespace. For example:

```
sequencename:1000-10000
sequencename 1000 10000
```

Optionally, L-click the ↻ button to update the green region highlight (and make sure your entry is valid). When ready to jump to the new region in your plot, L-click the GO button.

[2] Specific Coordinates

Select a **Sequence** from the pull-down menu, type in specific *Start* and *End* coordinates. Optionally, `L-click` the ↻ button to update the green region highlight (and make sure your entry is valid). When satisfied with the new region, `L-click` the GO button to close the menu and re-center your plot at the new region.

[3] Move the Region by Mouse

You can also `L-click` and drag the mouse along the graphical representation of the **Sequence** to move the selected region (while preserving the current plot width). This functionality is also available through the mini-browser in the bottom-right corner of each plot.

6. Location Mini-Browser

Access

The mini-browser is located in the bottom-right corner of each plot.

Usage

This browser is similar to the *Location Browser* in that you can drag the current viewing window along the **Sequence** length. The length of the current **Sequence** is displayed on the right-hand side of the mini-browser. Clicking the length display opens up a menu that allows you to switch between different **Sequence**s (provided there are less than 50 **Sequence**s in your **Sequence Set**).

Track Styles

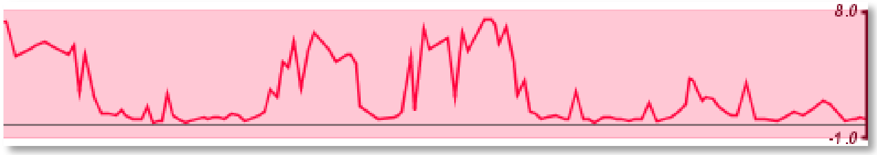
When displaying *Track Axis Elements* in MochiView, there are multiple *Styles* to choose from. Note that the x-axis on all of these *Track*s corresponds to the coordinates of a **Sequence**.

Interpreting Y-axis Information for Track Axis Elements

Some *Track*s (e.g. '*Shape style*') contain no information on the vertical axis. However, the remaining *Tracks* have an internal y-axis that does not correspond to the primary y-axis of the plot. The bounds of the *Track* y-axis are dictated by the preferred y_{max} and y_{min} associated with the **Data Type** of the plotted **Data Set**. Specifically, the top of the *Track* represents the preferred maximum y-value and the bottom of the *Track* represents the preferred minimum y-value. If the y-value of zero falls between these two values, a horizontal black line is drawn to represent the `y=0` axis. (The y_{max} and y_{min} values are displayed at the right-hand side of the *Track*.)

1. STYLE: Line

Sample Line Track



Constraints

Line Style can only be used with **Data Set**s and **Tiled Set**s.

Description

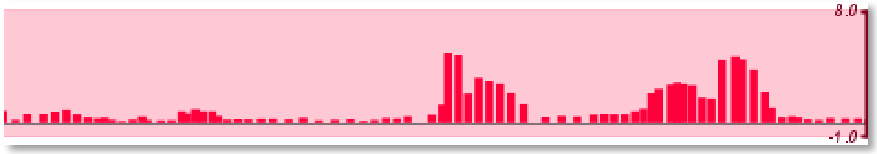
For each **Location** in the chosen **Data/Tiled Set** that has an associated value, an '*x,y*' coordinate is determined using the midpoint of the **Location** and the associated value. A point is plotted on the *Track* at this coordinate and connected to the nearest points on the x-axis. You can `R-click` on the track to toggle between displaying the line, the line with circles drawn on the individual data points, or just the circles.

Examples of Usage

Line Style is best used when **Location**s are close together and the **Location**s are of similar size (so the midpoint of the **Location** is adequate as x-axis coordinate). The primary usage of this style is to display data associated with probes on a tiling array or ChipSeq reads.

2. STYLE: Bar

Sample Bar Track



Constraints

Bar Style can only be used with **Data Set**s and **Tiled Set**s.

Description

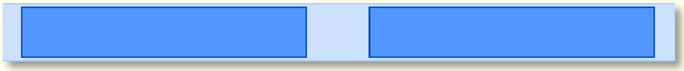
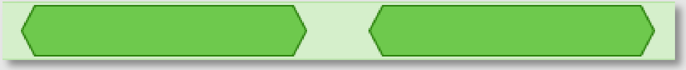
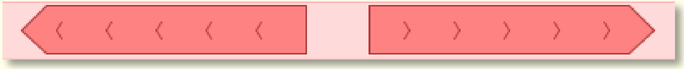
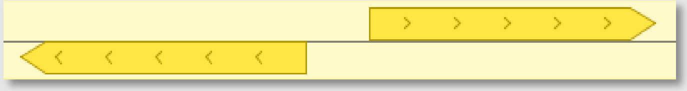
For each **Location** in the chosen **Data/Tiled Set** that has an associated value, a bar is drawn from the `y=0` axis of the *Track* to the coordinate equal to the value. The width of the bar corresponds to the width of the **Location**.

Examples of Usage

Bar Style is best used with **Data/Tiled Set**s that have a meaningful relationship with the zero-axis, such as:

- P-values ($\log_{10}$ transformed)
- Expression data ($\log_{2}$ transformed)
- Tiled ChipSeq read counts

3. STYLE: Shape

SHAPE	SAMPLE *(colors optional)*	COMMENTS
Rectangle		Basic shape (no directionality)
Hexagon		Aesthetic variant of *Rectangle*
Directional		Conveys **Location** directionality (right arrow = plus strand)
Directional: Axis		Conveys directionality with both arrow and shape placement above/below a drawn axis

Description

This style displays each **Location** in the selected **Location/Data/Tiled Set** as a shape. If a **Location Set** was selected, any available *Annotation Tag* is written on the shape. If a **Data Set** or **Tiled Set** was selected, the value associated with the **Location** is written on the shape. The four available shape options are presented in the table above. The primary difference between the *Shape Style*s is that only two, *Directional* and *Directional: Axis*, indicate **Location** directionality (i.e. plus or minus strand). The latter bisects the *Track* with a horizontal axis, and places plus-strand **Locations** above the axis and minus-strand **Location**s below.

All *Shape Style Track*s for **Data/Tiled Set**s default to using *Gradient Mode* (see the section titled '*SHAPE STYLE MODE: Gradient*' below for details). This default can be toggled on/off in the preferences menu.

Examples of Usage

*Shape Style*s can be used to display:

- ChIP-chip binding regions (optionally with p-values)
- Protein domains
- Promoters
- Motif locations (created with the '*Motif→Data Set*' utility)

Location Set Inner Text: Location Annotations

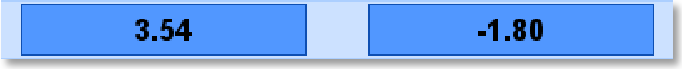
*Sample **Location Set** annotation text (any shape style can be used)*

I am annotated!
So am I!

If a **Location Set** displayed using *Shape Style* has any *Location Annotations*, the *Annotation Tag*s are displayed on the annotated **Location**s. *Location Annotations* can be modified or added by double-clicking on a **Location**. This applies to **Location**s of both **Data Set** and **Location Set** *Track*s, but only **Location Set** *Track*s display the *Annotation Tag*. When you modify a *Location Annotations* via a **Data Set** you are modifying the **Location** associated with the underlying **Location Set** (i.e. *Location Annotations* are only associated with **Location Set**s). Also, note that adding *Location Annotations* to very large **Location Set**s has the potential to slow display of the plot.

Data/Tiled Set Inner Text: Data Value

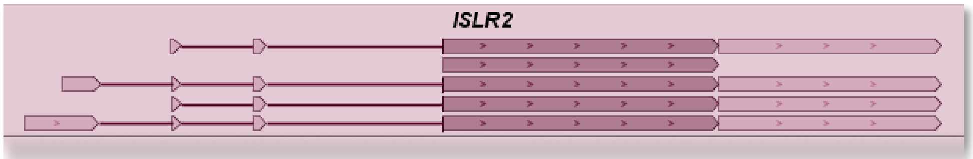
Sample Data Set value text (any shape style can be used)



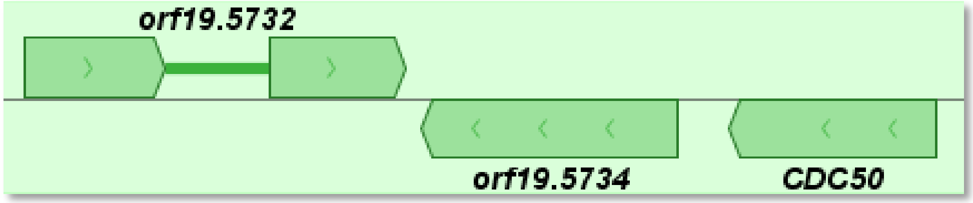
If a **Data Set** or **Tiled Set** is displayed using *Shape Style* the value associated with the **Location** is displayed on the shape. (In the case of **Data Set**s, the *Location Annotation* of the underlying **Location** can still be viewed and modified as described above.)

Enhanced Display: Gene Location Sets

Sample display of a complex gene (multiple isoforms including introns, coding and non-coding exons)



Sample display of simple genes (single isoforms and only one gene with an intron)



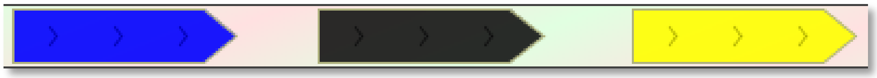
(NOTE: These gene track screenshots are a bit out of date. The most notable change is that only the last exon has the directional rectangle shape… the others are just regular rectangles)

Location Sets of **Location Type** '*Gene*' are given special formatting when displayed in *Shape Style* using the '*Directional: Axis*' shape (as pictured above). Specifically, the following special characteristics are added to the *Track*:

1. The gene name for each **Location** is displayed (or the feature name if no gene name is available)
2. Each isoform is displayed separately as a series of exons connected by a line. Non-coding exons have higher transparency than coding exons.
3. Each isoform (and each exon) has a customized tooltip.

SHAPE STYLE MODE: Edit

Sample Edit Mode display (TRUE, UNDECIDED, OFF)



Constraints

Gradient Mode and *Edit Mode* are mutually exclusive.

Description

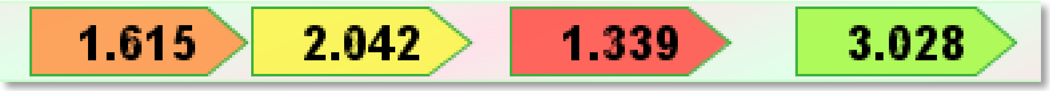
A *Shape Style Track* can be placed in *Edit Mode* by pairing it with a boolean **Data Set** that shares the same **Location Set** as the *Track Axis Element*. *Edit Mode* is described in the section covering the **NEW PLOT** menu. Briefly, the shapes can be toggled between three states by clicking with the mouse, and this state is saved in the associated boolean **Data Set**. Blue corresponds to the '`true`' state, yellow corresponds to the '`false`' state, and dark grey corresponds to '`undecided`'. (A reminder of the color code is given in the *Track* tooltip.)

Examples of Usage

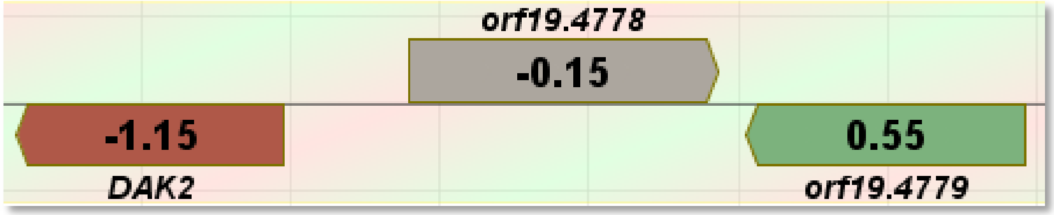
Edit Mode is useful for manual curation of a **Data Set**. For example, a set of putative binding regions identified by ChIP-chip could be further refined with manual curation using a plot that also displays the tiling array data.

SHAPE STYLE MODE: Gradient

Sample Gradient Mode Track (on a Directional Shape)



Sample Gradient Mode Track (on a Directional: Axis Shape gene track with log2-transformed expression data)



Constraints

Gradient Mode and *Edit Mode* are mutually exclusive.

Description

A *Shape Style Track* can be placed in *Gradient Mode* by pairing it with any **Data Set** that shares the same **Location Set** (in the case of **Tiled Set**s only a self-gradient is available). *Gradient Mode* is described in the section covering the **NEW PLOT** menu. Briefly, the shapes are colored according to the values in an associated **Data/Tiled Set**.

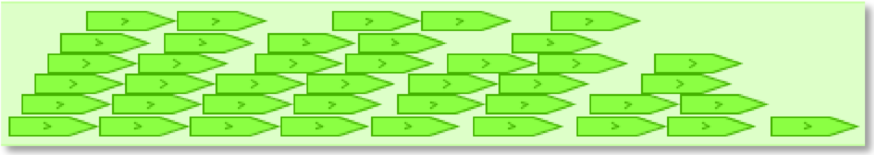
It is important to note that when a **Data Set** *Track* is placed in *Gradient Mode* (as opposed to a **Location Set** *Track*) the numerical values written on the shapes belong to the primary **Data Set**, not the **Data Set** dictating the shape color.

Examples of Usage

Gradient Mode is perhaps most useful for associated expression array data with a **Location Set** of **Location Type** '*Gene*' (as displayed in the sample above). Another usage is to turn a dense **Tiled Set** into a heat-map.

SHAPE STYLE MODE: Stack

Sample Stack Track



Constraints

Works for any *Shape Style Track* with the exception of *Track*s created from **Tiled Set**s or from a **Location Set** of **Location Type** '*Gene*' in the *Directional: Axis style*.

Description

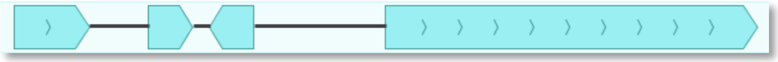
Stack Mode facilitates the display of overlapping **Location**s, by stacking them above each other on the *Track*. The user defines the parameter '*number of shapes to stack*', which dictates the maximum height of any individual **Location** shape in the *Track* (track height divided by '*number of shapes to stack*'). Thus, if the value is set at '1', all **Location**s will be as tall as possible. In the sample above, the value is set at '6', so the **Location** at the far right is the same height as the others (if the setting were '2', it would be half the size of the *Track*).

This is all rather convoluted, but the choice of parameter can be summarized as follows:

1. If you are simply enabling *Stack Mode* to prevent the overlapping of **Location**s when necessary, set the parameter at a value of '1'.
2. If you are using *Stack Mode* to visually emphasize **Sequence** regions with many overlapping **Location**s, set the parameter at a higher value (roughly at the maximum overlap level you would expect).

SHAPE STYLE MODE: Grouped

Sample Location Group



Constraints

Does not work in *Stack Mode*.

Description

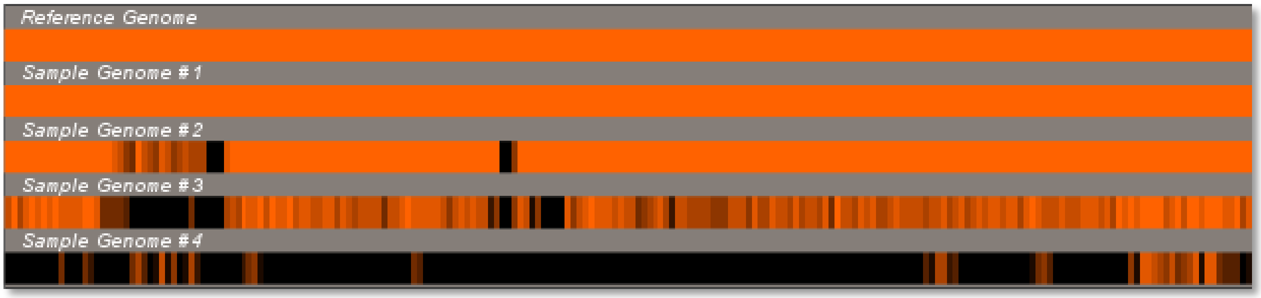
If a **Location Set** is uploaded with group assignments (i.e. the file has a 'GROUP' column), displaying **Location Set**s (or **Data Set**s associated with the **Location Set**) in *Shape Style* will result in all grouped **Location**s being connected by lines, as pictured above.

SPECIAL STYLE: Alignment Track

When a **Location Set** of **Location Type** '*Alignment*' is displayed in a plot, special formatting is applied to the *Track*. Each aligned genome is displayed as a "sub-*Track*" (the reference genome is always the top "sub-*Track*"), and the information content displayed varies with zoom level.

Zoomed Out: Conservation Display

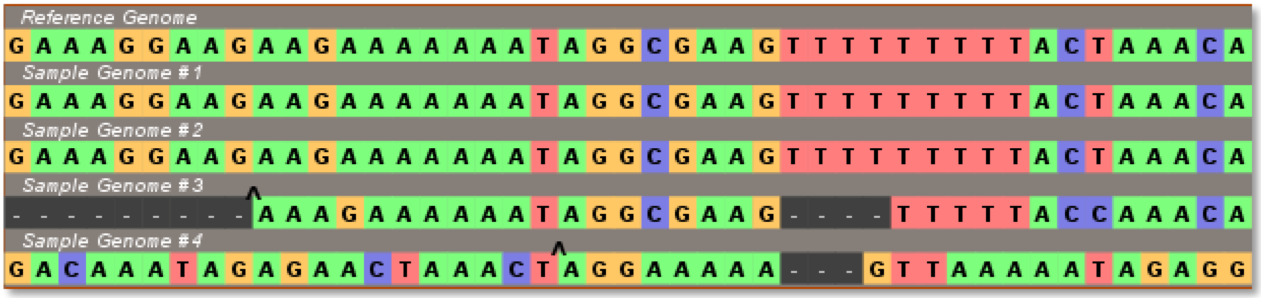
Sample 5-genome Alignment Track (ZOOMED OUT)



When the plot is zoomed out beyond the distance where individual nucleotides can be visualized, the plot displays a color-coded chart that indicates the degree of conservation between in each genome and the reference genome. The calculation of conservation is performed in 100bp blocks, and the block color ranges from black (< 10% conservation) to the selected Track color (orange in the case of the example above; > 90% conservation). Note that this calculation does not include inserts in non-reference genomes (i.e. if the 100bp reference genome sequence is perfectly matched by an aligned genome, yet the aligned genome contains a 500bp insert within this sequence, the conservation is still considered to be 100%).

Zoomed In: Nucleotide Display

Sample 5-genome Alignment Track (ZOOMED IN)

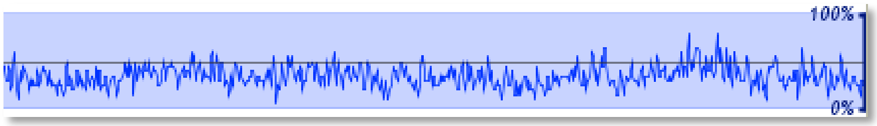


Reference Genome
G A A A G G A A G A A G A A A A A A A T A G G C G A A G T T T T T T T T T A C T A A A C A
Sample Genome #1
G A A A G G A A G A A G A A A A A A A T A G G C G A A G T T T T T T T T T A C T A A A C A
Sample Genome #2
G A A A G G A A G A A G A A A A A A A T A G G C G A A G T T T T T T T T T A C T A A A C A
Sample Genome #3
- - - - - - - - - A A A G A A A A A A T A G G C G A A G - - - - T T T T T A C C A A A C A
Sample Genome #4
G A C A A A T A G A G A A C T A A A C T A G G A A A A A - - - G T T A A A A A T A G A G G

When the plot is zoomed in to the level of nucleotide resolution, the *Alignment Track* displays the individual aligned nucleotides. In addition, carets are displayed to indicate the location of any sequence inserts in the aligned genomes (the size and sequence of the insert can be determined from the associated tooltip). By default, conserved bases are not drawn, but this option can be toggled by `R-click`ing on the *Alignment Track* (as in the sample above).

SPECIAL STYLE: GC% Track

Sample GC% Track



Constraints

This *Track* can only be selected on the first tab of the **NEW PLOT** menu.

Description

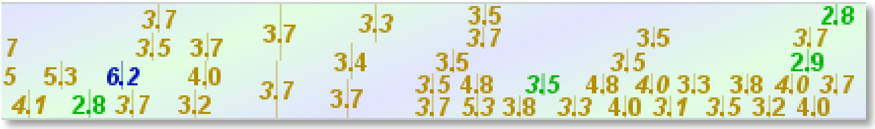
The *GC% Track* is a specialized version of the *Line Style* that dynamically calculates the '`GC`' content of the currently viewed **Sequence**. The '*GC%*' is calculated in a 20bp window at 10bp intervals along the **Sequence**. If you zoom out, the window/interval sizes increase (>250,000bp = 200bp/100bp; >1,000,000bp = 1000bp/500bp). The top of the *Track* corresponds to 100% 'GC' content (i.e. no '`A`' or '`T`' nucleotides), and the bottom corresponds to 0% '`GC`' content. Unlike typical *Line Style Tracks*, the horizontal black line represents 50% instead of zero.

Examples of Usage

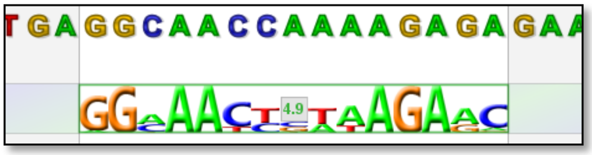
Include this *Track* in your plot if you'd like to monitor whether the other data you are plotting is influenced by local '`GC`' content.

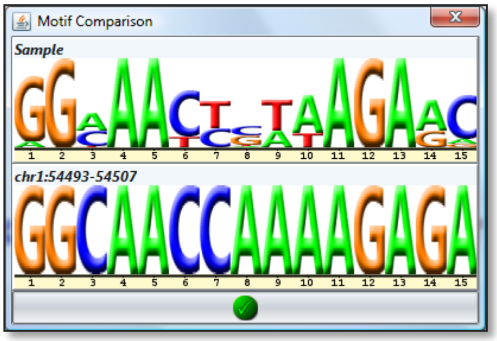
SPECIAL STYLE: Motif Track

Motif Track (ZOOMED OUT, with scores for three different Motifs plotted)



Motif Track (ZOOMED IN, single motif with comparison pop-up)





Description

The *Motif Track* displays the scores of **Motif** hits that exceed a user-defined cutoff. (See the section titled "*Scoring and Identifying Motifs*" for details on scoring.) This style of *Track* is configured using the '*Motifs*' tab of the **NEW PLOT** menu. In this menu you can check/uncheck the **Motif**s that you would like to include and adjust the score cutoff by clicking in the '*Score Cutoff*' column of the table. You can further customize the color of the displayed score by clicking on the sample **Motif** *Track* that appears when you enable the checkbox for a row in the table. By default all selected **Motif**s will be displayed on a single *Track*. You can split the **Motif**s between multiple *Track*s using the '*Track Order*' tab.

When viewing the *Motif Track* in a plot, the *Track* tooltip will review the colors assigned to the **Motif**s, and the tooltip for a **Motif** hit will provide information about the matching sequence. For a visual comparison of the **Motif** logo and the sequence that passed the cutoff, `double-L-click` on the hit location.

Common Menu Elements

Many features of the MochiView menus are reused in several different places. Rather than describe them repeatedly for the different menus, they are discussed below.

Pulldown Menus

These menus are found throughout MochiView, typically allowing you to choose between members of a data category.

Menus are sometimes chained together

In some cases your selection in one pulldown menu will influence the contents of a pulldown menu below. The most common example is that a pulldown menu that allows **Location Set** selection if often populated with **Location Set**s that belong to a **Sequence Set** that was chosen in a pulldown menu above it. A similar relationship often exists between **Location Set** and **Data Set** pulldown menus.

Virtually all pulldown menus have tooltips

Hover the mouse over any item in a pulldown menu and you will see a tooltip providing further information.

Tables

Most of the tables in MochiView can be sorted by `L-click`ing on the header. Click multiple times to cycle through *ascending*, *descending*, and *disabled* sorting (a triangle icon in the header will indicate the current sort status). You can hold down `'Ctrl'` and `L-click` multiple headers in turn to set up a multi-column sort. In tables where multiple rows can be selected (e.g. **MANAGER** tables), select a row by `L-click`ing and then either select additional rows by hold down `'Ctrl'` and `L-click`ing or select an interval (i.e. a second row and all rows in between) by holding down `'Shift'` and `L-click`ing.

Data Filters

Several menus in MochiView provide the option to implement *data filters* (e.g. the **NEW PLOT** menu, *'Utilities →Location/Data/Tiled Set →Refine/Location Data Set'*, and *'Export → Location Set →Format: FASTA'*)

Data filters can be used to limit the **Location**s in a **Location Set** (or **Data Set**) that are included in the operation relevant to the menu (e.g. which **Location**s are displayed in a new plot or which **Location**s are exported to a text file). Any **Data Set** that shares the same **Location Set** as the item being filtered (which will be either a **Data Set** or a **Location Set**) can be used as a *data filter*.

To apply a data filter:

- Choose a **Project** from the 1st pull-down menu
- Choose a **Data Set** from the 2nd pull-down menu
- Choose an *operator* from the 3rd pull-down menu (e.g. >, <, =)
- Choose a *cutoff value* from the 4th pull-down menu
- Click the ⊕ button

The *data filter* will then be displayed as a row in a table inside the filter menu, and a background calculation of the number of **Location**s passing the filter will commence. (While calculating, the number is displayed as `'???'`). Only those **Location**s that meet the criteria designated by the operator and cutoff are considered to pass. Note that any **Location** that lacks a value in a **Data Set** used for filtering is always considered as not passing. Individual *data filter*s can be removed using the ⊗ button next to the entry.

Overlap Filters

With the exception of the **NEW PLOT** menu, the filter configuration menu consists of an upper panel for configuring *data filters* and a lower panel for configuring *overlap filters* (plots cannot use overlap filters because they are too resource-intensive). *Overlap filter*s act downstream of the *Data filters* and are applied based on overlap between the exported **Location** and a chosen **Location Set**. Unlike the *data filters*, which either include or exclude a **Location**, *overlap filter*s can also truncate **Location**s (see below). When used in conjunction with the '*Restrain to a maximum length*?' panel (see below), these truncations are applied first.

Overlap Filter Criteria

Keep if any overlap

If any **Location** in the filter **Location Set** overlaps with the export **Location**, the **Location** is included in the *FASTA* file.

Exclude if any overlap

If any **Location** in the filter **Location Set** overlaps with the export **Location**, the **Location** is excluded from the *FASTA* file.

Keep non-fragmented overlapping regions

Only the portion of the export **Location** that overlaps with **Location**(s) in the filter **Location Set** is included in the *FASTA* file. However, if this criterion results in fragmentation of the export **Location**, the export **Location** is excluded. This option is not offered in some filter menus.

Exclude non-fragmented overlapping regions

Only the portion of the export **Location** that does not overlap with **Location**(s) in the filtered **Location Set** is included in the *FASTA* file. However, if this criterion results in fragmentation of the export **Location**, the export **Location** is excluded! This option is not offered in some filter menus.

'Choose Background Frequency Calculation Method' panel

This panel appears in all menus that involve the creation of new **Location Set**s, and provides options for determining how the base background frequencies (used for PSFM **Motif** scoring) of the **Location**s in the **Location Set** should be calculated. The default option (calculate directly from the **Location**s) is almost always the correct one, unless you are working with a very large genome (e.g. human genome). In this case, the calculation can take quite some time, and it may be preferable to choose the option to either use the background frequencies from the underlying **Sequence Set** or enter them manually. If you are not using a very large genome, you can disable the display of this panel in *Utilities→Preferences*.

'Restrain to a Maximum Length?' panel

Panels with this header appear in several MochiView menus. When the checkbox in the panel is checked, the export **Location** is truncated to a '*maximum length*' supplied by the user. The '*maximum length*' is applied by extending the **Location** from the chosen centering point (see below) to either side by half of the '*maximum length*' (or up to the **Location** boundary). Note that in cases where the '*maximum length*' exceeds the **Location** size or the centering point is not at the **Location** midpoint, it is possible that the selected sub-region of the **Location** will be less than the '*maximum length*'.

Centering Options

Peak Centering

The user selects a **Data Set**, and the following approach is taken to determine the centering coordinate for each export **Location**:

1. All **Location**s of the chosen **Data Set** that have a midpoint that falls within the export **Location** and have an associated value are considered. (If no **Location**s meet these criteria, '***Midpoint Centering***' is used).
2. Of these **Location**s, the midpoint of the **Location** with the highest associated value is chosen as the centering point for the export **Location**.

Midpoint Centering

The midpoint of the export **Location** (rounded down) is used for centering.

'Gene Proximity Assignment' panels

It is often helpful to associate each **Location** in a **Location Set** with one or more genes (e.g. assign ChIP-Chip binding regions to genes). MochiView provides a utility with numerous options for configuring such assignments. These options introduce considerable complexity, and if you'd rather just forge ahead, the defaults are reasonable for most applications (except one might want to increase the default maximum search distance if using a genome with large intergenic/promoter regions).

Proximity Calculation

The panels titled '*Select gene proximity criterion*' and '*Select location proximity criterion*' contain pull-down menus that determine the approach used to calculate the proximity of a gene to a **Location**. This proximity calculation is used to determine which gene(s) are closest to the **Location**. The different criteria are described in the tooltips. The default selections of '*Gene start*' and '*Full Location*' indicate that the proximity is calculated as the distance from the start of the gene **Location** to the closest end of the query **Location**. If the start of the gene **Location** falls within the query **Location**, the distance is 0bp (thus, a very large query **Location** may have many genes assigned with proximity of 0bp).

Additional Settings

Maximum number of genes

This setting provides the upper limit for the number of genes associated given with a given **Location**. However, this limit will be exceeded in cases where multiple genes have the same proximity!

Maximum search distance (bp)

This setting indicates the maximum allowed proximity value for the query **Location** and the closest gene. If no gene is within this maximum the query **Location** is not assigned to any genes.

Maximum search distance beyond closest gene (bp)

If the current configuration allows more than one gene to be associated with a query **Location** (see *Maximum number of genes*), this setting dictates the distance beyond the closest match that will be searched for additional genes. This is helpful in preventing the inclusion of genes that are at the far end of the maximum search range provided that the closest gene is at the low end of the maximum search range. In other words, consider the default:

- *Maximum number of genes*: 3
- *Maximum search distance*: 10,000bp
- *Maximum search distance beyond closest gene*: 1,000bp

If the three closest genes to the query **Location** have proximities of 100bp, 9,500bp, and 10,200bp, only the closest gene is assigned (because the 2nd and 3rd closest genes have proximities of more than 1,100bp). However, if the closest gene were 9,300bp away, all three genes would be assigned (because the 2nd and 3rd closest genes have proximities less than 10,300bp. In this latter case the assignment is more ambiguous, so it makes sense that all three genes should be included.

Maximum one gene per strand

This checkbox restricts the number of assigned genes to include at most one gene from the plus strand and one gene from the minus strand, unless multiple genes have identical proximities (in which case all equidistant genes are included).

Part of the location must be upstream of the start of the gene

As stated in the title, this checkbox imposes the requirement that some part of the query **Location** must lie upstream of the start of the gene **Location**. This is helpful when one wants to enforce the assumption that the query **Location** (e.g. ChIP-Chip binding region) should lie upstream of the gene.

Allow at most one gene one each side

(When this box is checked, certain other options are locked in a specific state.) When enabled, the search is confined to the closest gene upstream and downstream of the **Location** (multiple genes might be included if they are equidistant). Genes with a *proximity distance* of zero are also included, and considered as overlapping. If using this feature in conjunction with *'Export→Location Set→Gene Proximity Assignments'*, the output format will be a custom version of the "Locations in Rows" format in which the genes are subdivided into the groups LEFT/CENTER/RIGHT, to indicate their spatial relationship to the **Location**.

'Tiled Set storage settings' panel

This panel (or portions thereof) can be found in the following utilities (list may not be complete):

- *'Import→Tiled Set→Format: WIG'*
- *'Import→Tiled Set→Format: Eland extended'*
- *'Utilities→Location/Data/Tiled Set→Combine Tiled Sets'*
- *'Utilities→Location/Data/Tiled Set→Combine Tiled Set groups (addition/subtraction)'*
- *'Utilities→Location/Data/Tiled Set→Combine smoothed Tiled Set from Tiled Set(s)'*
- *'Utilities→Location/Data/Tiled Set→Combine smoothed Tiled Set from Data Set(s)'*

This panel provides several settings that control the final storage/data format of the newly created **Tiled Set**.

Select data compression type

MochiView's **Tiled Set**s store data in a compressed format modeled after that described for Wiggle tracks (http://genomewiki.ucsc.edu/index.php/Wiggle). Depending on the range of values present in the data, some precision may be lost during compression. Three different compression levels are offered:

Compression Level	Bytes / Data Point	Data Resolution
Low	3 bytes	16,711,679
Moderate	2 bytes	65,279
High	1 byte	254

The precision is calculated on a per-**Sequence** basis using the range of values associated with that **Sequence** (i.e. if the lowest value is -`500` and the highest is `2500`, the range is `3000`). A detailed explanation is provided in the link above, but the upshot is that if your range is 254 and your data are integers, the '*High*' compression level does not compress your data at all. Similarly, if your range is 254 and your data contains fractional units (e.g. `2.54`), use the '*Moderate*' compression level to preserve the fractional information. If in doubt, just use the default '*Low*' compression level.

Adjust signs of values in file?

This setting provides the option of storing imported values as either positive or negative. This is useful when creating two **Track Set**s that describe tiling data for the plus- and minus-strands and then displaying them on a single *Track* using a bar graph. (In this example, the plus-strand data would be stored as positive values and the minus-strand data would be stored as negative values.) Needless to say, do not use this option if the values are not all either positive or negative.

Apply $\log_2$ transformation?

When displaying data containing values that range from zero (or higher) and upwards, it may be easier to visualize the full range of data by transforming the data. As a bit of a fudge to make small values visible and avoid negative numbers post-transformation, any value greater than `0` and less than `2` is stored as `0.5`, and any value below `0` is stored as `0`. This option is typically only useful for the *Import* utilities. While you can pre-transform data prior to import, be aware that MochiView displays tiled pre-computer averages of your data when you zoom out the plot (see below), and the averages of log-transformed data would not be accurate. When the '*Apply* $\log_2$ *transformation?*' option is used instead, the log-transformation can be applied after averaging your data. (This is also a problem when using various utilities that manipulate log-transformed data in the database to create a new **Tiled Set**. In many of these cases an option is provided to un-transform the data first.)

When zoomed out display largest deviation from zero

MochiView pre-calculates values for larger display spans so that **Tiled Set** data can be displayed smoothly when zoomed out in a plot. If this option is chosen, the value deviating the most from zero is used instead. Otherwise, the value is the average of the values within each span. Please note that in this latter case missing values are not considered, so if the imported file, for example, represents sequence read counts and omits positions that have no reads, these positions are not factored into the average. In other words, a large span with a single entry will have an average equal to that entry's values. In contrast, if all other positions had entries of zero, the average would include those as well and be much smaller.

Apply multiplier? (import only)

If you plan on viewing multiple **Tiled Set**s on a single track (or the primary axis), it may be necessary to apply a multiplier to the values of some tracks to normalize them to a similar value (e.g. comparing ChIP-Seq experiments with different read counts). If enabled, this option applies the entered multiplier to each value. (If you have also elected to apply a $\log_2$ transformation the multiplier is applied before transformation.)

Omit zero-value entries from final Tiled Set? (non-import only)

Many of the utilities that create a new **Tiled Set** from existing data can end up with tiles with a value of zero. If you are typically displaying **Tiled Set**s as bar graphs, keeping track of all the zeros is takes up additional hard-drive space and may (slightly) slow rendering of plots. This is especially the case for very large genomes. This setting provides the option of omitting these zeros from the **Tiled Set** (they become "missing data" instead). There are, however, two downsides to this approach:

1. If you display the data as a line graph the lines will not connect through the zeros.
2. When zoomed out, if you did not choose the "*When zoomed out display largest deviation from zero*" option, the zeros will not be factored into the zoomed out median-value display (see description of this option for details).

'Import' Menu

All database entries are created by importing the contents of appropriately formatted text files. With the exception of the formats used to import **Sequence Set**s and alignments, all files are assumed to be tab-delimited. The first row is always assumed to be the header row, and no two columns can have the same header. If the required file headers are not found, the import will not proceed. Note that additional columns should (in theory) never cause problems with import (they are either ignored, or in the case of certain types of **Data Set** import you are given the option of including them).

Import→Sequence Set→Format: FASTA

File Format

The utility accepts either a FASTA file with an entry for each **Sequence** or a GFF file (version 3) that contains FASTA-formatted sequences at its end. (Note that the file selection menu allows you to select multiple files or an entire directory.) Files can be `gzip`ped (with names ending in "`.gz`"), but may not be `tar` archives. All IUPAC symbols are allowed (`A`,`C`,`G`,`T`,`N`,`M`,`R`,`W`,`S`,`Y`,`K`,`V`,`H`,`D`,`B`). However, please note that when scanning sequence for **Motif**s any window containing a non-`ACGT` base is ignored.

The name of the **Sequence** (maximum of 50 characters) is extracted from the FASTA headers. By default, the text following the "`>`" symbol up to the first whitespace is used as the name (there is a checkbox in the import menu that allows you disable this setting and include all text beyond the ">"). Each **Sequence** name must be unique for a given **Sequence Set**. It is strongly recommended that the names be kept very short (*e.g.* `Chr1`) to prevent display cropping issues.

All lines preceding the first line starting with ">" are ignored, as are all empty lines and those that begin with "#".

Comments

Importing a very large genome (e.g. human genome) will take roughly 20-30 minutes; during this time no other action can be taken. Start the import, make sure the validation step is successful, and then go grab some coffee. Smaller genomes (e.g. *S. cerevisiae*) will take only a few seconds to import.

Once imported, **Sequence**s in a **Sequence Set** are immutable. In other words, you must import all **Sequence**s at one time, and the nucleotide sequences and **Sequence** names can never be changed.

Stored in Database

The database stores the provided **Sequence Set** annotation (*name*, *description*, *species*), plus the names and nucleotide sequences of the individual **Sequence**s.

Import→Location Set→Format: MochiView

Required Headers

SEQ_NAME, START, END

Optional Headers

STRAND, ANNO_TAG, ANNO_DESC, GROUP

File Format

Guidelines for individual columns (max # refers to the maximum length of the entry):

SEQ_NAME (required; max 50)

The entry must correspond to the name of a **Sequence** in the database.

START (required) and END (required)

These columns must contain integer values, neither of which exceeds the length of the indicated **Sequence**. The 'START' and 'END' coordinates are inclusive, and the 'START' coordinate should be less than or equal to the 'END' coordinate.

STRAND (optional)

If this column is omitted, all **Location**s are assumed to be on the plus strand. If included, the value should be '+' (no quotes) for plus strand **Location**s and '−' for minus strand **Location**s (in practice, MochiView assigns anything without a minus to the plus strand).

ANNO_TAG (optional; max 15) and ANNO_DESC (optional; max 500)

These columns supply additional *Location Annotation*s for the imported **Location**s. The value supplied for 'ANNO_TAG' is displayed as an *Annotation Tag* on **Location**s when the **Location Set** is displayed on a *Track***.** The value supplied for 'ANNO_DESC' is an *Annotation Description* that is visible when a **Location** in a plot is double-clicked. If only an 'ANNO_DESC' column is supplied (or the 'ANNO_TAG' column has a blank entry), the 'ANNO_TAG' is created from the first 15 characters of the 'ANNO_DESC' entry. Note that *Location Annotations* do not have to be supplied for all rows.

GROUP (optional)

If included, this column assigns **Location**s to groups. Currently the only effect of group information in MochiView is that *Shape Style* plots (described later) will connect the **Location**s in a group with a thick black line. This column must contain a numerical integer entry (above zero) for each row. Any **Location**s sharing the same integer are considered to be in the same group.

Comments

Importing 250,000 **Location**s takes roughly one minute.

Stored in Database

The database stores the **Location Set** annotation (*name, species*, **Sequence Set**) plus all of the associated **Location**s.

Import→Location Set→Format: BED

File Format

This import utility utilizes the '*BED*' file format (http://genome.ucsc.edu/FAQ/FAQformat#format1) to create a **Location Set**. The **Location** is extracted from the following columns:

- *Column#1 (chrom):* Equivalent to the standard MochiView 'SEQ_NAME' column.
- *Column#4 (chromStart) and Column#5 (chromEnd):* Similar to the standard MochiView 'START'/'END' columns, except that the entry in column#4 begins with a '0' coordinate and the entry in column#5 is exclusive. In other words, a **Location** represented as 1-100 in MochiView is entered as 0-100 in this format.
- *Column#6 (strand):* Equivalent to the standard MochiView 'STRAND' column.

The file is assumed to have a one line header of some sort (**the first line is ignored!**). Import of *Location Annotations* is not supported when using this file format.

Import→Location Set→Format: GFFv3

File Format

This import utility utilizes the '*GFF*' (version 3) file format (with some restrictions… see below) to create a **Location Set**. The **Location** is extracted from the following columns:

- *Column#1 (seqid):* Equivalent to the standard MochiView 'SEQ_NAME' column.
- *Column#4 (start) and Column#5 (end):* Similar to the standard MochiView 'START'/'END' columns.
- *Column#7 (strand):* Equivalent to the standard MochiView 'STRAND' column, but note that all non '–' entries are considered plus strand (MochiView does not support un-stranded **Locations**)

Import of *Location Annotations* is not supported when using this file format. Although GFFv3 allows FASTA sequence to be appended to the end of a file, this utility does not support this and you will get an error message. Also, please note that any **Location**s that share the exact same coordinates and strand will be considered duplicates and will only be entered once.

Import→Location Set→Format: GFFv3 (by type)

This utility is similar to the *'Import→Location Set→Format: GFFv3'* utility described above, except that:

1. The file is pre-scanned and you are then given the choice of which '*feature type*' categories (column#3 in the GFF format) to include in the **Location Set**.
2. Annotation is added to the **Location**s using items in the '*attributes*' column (column#9 in GFF). You choose which '*attribute*' to apply to the *Annotation Tag* and which to apply to the *Annotation Description* (you can also choose to make the '*feature type*' the *Annotation Tag*.
3. This importer is restricted to terms without a '*Parent=*' entry in the '*attributes*' column (column#9 in GFF).
4. If the GFF file contains FASTA sequences (which is allowed in GFFv3 format), you are given the option to import the sequences as a **Sequence Set**.

See also the more detailed description for the *'Import→Location/Data Set→Format: GFFv3 (by type)'* utility, which is just like this utility except it also imports a **Data Set** using values found in the '*score*' column (#6).

Import→Location Set (genes)→Format: MochiView

Required Headers

SEQ_NAME, START, END, STRAND, FEATURE_NAME, TXN_START, TXN_END, EXON_COUNT, EXON_STARTS, EXON_ENDS

Optional Headers

GENE_NAME*, ALIASES, DESCRIPTION, CDS_START, CDS_END, ISOFORM_NAME, IS_PRIMARY

*Was required in versions prior to v1.26

File Format: Overview

The entry in the 'SEQ_NAME', 'START', 'END', and 'STRAND' columns are interpreted as described for *'Import→ Location Set→Format: MochiView'*. The remaining format is a hybrid of the formats used by the *Saccharomyces Genome Database* (SGD) and the *UCSC Genome Browser*. MochiView can display multiple isoforms of a single gene. A separate line should be made for each isoform. The entries under the blue headers are specific to each isoform. The entries under the red headers must be the same for each isoform.

File Format: GENE FIELDS

FEATURE_NAME (required; max 50)

This column must contain a unique descriptor of the gene.

GENE_NAME (optional v1.26+; max 50)

The gene name should be unique if not left blank.

ALIASES (optional; max 500)

Aliases are provided as a pipe delimited text string of alternative names (*e.g.* HEX7|Contig12.5231|YUP7). Aliases do not have to be unique (but the program currently makes no use of non-unique aliases).

DESCRIPTION (optional; max 500)

This column can be left blank, or can contain a description of the gene that will be displayed in tooltips.

File Format: ISOFORM FIELDS

TXN_START and **TXN_END (required)**

These mandatory columns delineate the (inclusive) boundary coordinates of the isoform transcript. It does not matter which is the lower of the two values (MochiView infers the direction from the gene 'STRAND' entry). The coordinates for 'TXN_START' and 'TXN_END' must fall within the boundary delineated by the gene 'START' and 'END' coordinates.

EXON_COUNT (required)

This entry indicates the number of exons listed in the 'EXON_STARTS' and 'EXON_ENDS' fields.

EXON_STARTS and **EXON_ENDS (required)**

These two columns contain pipe-delimited coordinates for the coordinates (inclusive) of all exons in the isoform. As with the 'TXN_START' and 'TXN_END' columns, it does not matter which column contains the lower of the two values. For example, two exons with coordinates of 1000→1500 and 2250→3000 would be represented by an entry of '`1000|2250`' (omit quotes) in one column and '`1500|3000`' in the other (and an entry of '`2`' in the 'EXON_COUNT' column). The exon coordinates provided in these columns must all fall within the boundary delineated by the 'TXN_START' and 'TXN_END' columns.

CDS_START and **CDS_END (optional)**

These columns are optional, but if you include one you must include the other. The coordinates provided in these columns delineate the portion of the isoform transcript (if any) that is a protein coding region. These coordinates must fall within the boundary delineated by the 'EXON_STARTS ' and 'EXON_ENDS' columns. MochiView displays non-coding exons with higher transparency than coding exons. (If the boundary of the coding region falls within an exon, MochiView will split the exon.)

ISOFORM_NAME (optional; max 50)

Each isoform can be given a name (which can be seen in the plot tooltip for the isoform).

IS_PRIMARY (optional)

MochiView orders the isoforms in a vertically stacked display (using the criteria described below). An entry of '`Y`' (omit quotes) assigns an isoform as 'primary', ensuring the isoform is displayed first (at the top of the stack). Only one isoform for each gene can have an entry of '`Y`'.

Comments

Including more than ~10 isoforms per gene is not recommended, as the display will become difficult to read. Isoforms are primarily a cosmetic feature in MochiView, providing additional annotation and the ability to highlight individual exons in the plot (and copy the sequence to clipboard). For all MochiView utilities that utilize **Location Set**s, it is the start and end of the gene as a whole that is used to define the **Location** (e.g. when taking the union of two **Location Set**s). The criteria used to sort isoforms for display are, in order of precedence, as follows:

1. Declared primary in 'IS_PRIMARY' column
2. Longest coding sequence
3. Longest transcript

Stored in Database

Everything stored during an *'Import→Location Set→Format: MochiView'* operation, plus the additional gene-specific and isoform-specific annotation.

Import→Location Set (genes)→GFFv3

IMPORTANT NOTE

This utility has been tested for the following genomes (links were last tested Feb 18, 2010):

S. cerevisiae

http://downloads.yeastgenome.org/chromosomal_feature/saccharomyces_cerevisiae.gff

C. albicans

http://www.candidagenome.org/download/gff/candida_21_with_chromosome_sequences.gff.gz

D. melanogaster

ftp://flybase.net/genomes/Drosophila_melanogaster/current/gff/dmel-all-no-analysis-r5.24.gff.gz

Other genomes may not meet the strict file format requirements outlined below…

File Format

This import utility utilizes the '*GFF*' (version 3) file format (with significant restrictions… see below) to create a **Location Set** of **Location Type** '*Gene*'. The file can be `gzip`ped, provided that its name ends in "`.gz`" and that it is not a tarball (i.e. not "tar.gz").

The entries in the first column (*seqid*) must match the name of **Sequence**s in the chosen **Sequence Set**. (The relevant columns for extracting **Location** coordinates are fully described in the entry for the *'Import→Location Set→GFFv3'* utility.)

Required 'Type' hierarchy to reconstruct a gene

Entries in a GFF file are arranged into a hierarchy using '*tag=value*' pairs separated by semicolons in the '*attributes*' column (#9) of the file. Specifically, entries are identified by an '*ID=xxx*' tag, and if the entry has a "Child", the child's entry indicates this relationship using the '*Parent=xxx*' tag (where '*xxx*' is a unique ID). Note that when MochiView searches these tags the lookup is case-insensitive.

The flexibility of the GFF format allows for limitless and extremely complex hierarchical relationships, which do not lend themselves well to storage in MochiView's database in a fashion that allows speedy retrieval for smooth plot navigation. In MochiView, genes can be considered a hierarchy of Gene→Isoforms→Isoform Sub-Features (introns, coding- and non-coding exons).

When trying to extract a gene set, the following '*types*' (column#3) are used (they are case-insensitive):

TIER#1: Gene

Gene, ORF, Transposable_Element_Gene

TIER#2: Isoform (must have Parent that is TIER#1)

mRNA, miRNA, snRNA, snoRNA, tRNA, rRNA, ncRNA, Pseudogene

TIER#3: Isoform Sub-Feature (must have Parent that is TIER#2)

CDS, Exon, Noncoding_Exon, Intron

(This utility is a work in progress… please let me know if I have overlooked any major '*types*')

TIER#1: Requirements

- Must have a unique *ID*
- May not have a *Parent*
- No two entries can have the exact same **Location**

TIER#2: Requirements

- Must have exactly one TIER#1 *Parent*
- Must have a unique *ID*
- Coordinates must fall within TIER#1 *Parent*'s coordinates

TIER#3: Requirements

- Must have at least one TIER#2 *Parent*. If multiple *Parent*s are indicated, all must have the same TIER#1 *Parent*. (i.e. a sub-feature such as an exon can be mapped to multiple isoforms of the same gene)
- Coordinates should fall within TIER#2 *Parent*'s coordinates (exceptions are made, see below)

Extracting the gene properties: 'Feature Name', 'Gene Name', 'Aliases', 'Description'

MochiView requires the each gene have a unique *Feature Name*, and allows for an optional general *Gene Name*, *Alias* set, and *Description*. The maximum lengths for these properties are the same as those described for *'Import→Location Set (genes)→Format: MochiView'* (they are truncated if necessary). The properties are extracted from the TIER#1 entries as follows:

Feature Name

If an '*attribute*' tag of *'Name*' exists, the value is chosen as the *Feature Name*. Otherwise, the *ID* is used. All *Feature Name*s must be unique.

Name

If an '*attribute*' tag of *'Gene*' exists, the value is chosen as the *Gene Name* (this is the format used by Saccharomyces Genome Database). Otherwise, if an '*attribute*' tag of *'fullname*' exists, the value is chosen as the *Gene Name* (this is the format used by FlyBase). Otherwise, no *Gene Name* is entered and MochiView displays the *Feature Name* in its plots instead.

Aliases

If an '*attribute*' tag of *'Alias*' exists, the *Aliases* are taken from the value (multiple *Aliases* should be separated by commas). If the *ID* was not used as the *Feature Name*, it is also included as an *Alias*.

Description

If an '*attribute*' tag of *'Note*' exists, the associated value is used as the *Description*.

Yeast (Candida Genome Database and Saccharomyces Genome Database) workaround

The GFF files from these databases use a format that basically combines TIER#1 and TIER#2 (because there are no isoforms in their files). The import menu gives the option of enabling this workaround, which results in TIER#1 and TIER#2 being combined.

Other File Format limitations

ID entries must be unique

GFFv3 allows for discontinuous features that all have the same '*ID=xxx*' entry. MochiView is unable to handle such relationships, and if they are encountered you will see an error message telling you that non-unique IDs are not allowed.

Isoform sub-features should fall within the isoform boundaries*

In general, sub-features (e.g. *introns*, *exons*, *CDS*) should always fall within the coordinates of the isoform. However, in some cases this does not occur. For example, the FlyBase *Drosophila melanogaster* GFF file has some mitochondrial *mRNA* entries that have a *CDS* child that extends a few base pairs beyond the *mRNA* coordinates because these RNAs are post-processed to add a stop codon. To accommodate such issues, MochiView will truncate sub-features that extend beyond the isoform boundaries so that they fit, provided that they do not extend more than 10bp outside the boundary.

Import→Location Set (alignment)→Format: FASTA

File Format

This utility allows the import of a multiple alignment in FASTA style, provided that one of the aligned sequences (or genomes) can be mapped to an existing **Sequence Set**. The format is rather complex, so an example is provided below (a single aligned block of three "genomes") followed by a detailed explanation of the requirements.

```
#GENOME=My First Genome
#GENOME=2
#GENOME=Reference Genome

> My First Genome:2000-2003 + header comments
ACTT-----
> Reference Genome:10-16 – chr1
AC--GGATC
> 2
-C-TGGTTC
=
```

Headers (blue text)

The headers are lines at the start of the file beginning with '#' (quotes not included). For each aligned genome, there should be a header such as:

```
#GENOME=<genome_name>
```

<genome_name>: All aligned genomes should be named in a header row (the order is irrelevant). The name can be alphanumeric and can include spaces (as in the example), but cannot include a colon. When you select your file in the import utility, you will be asked to select a **Sequence Set** (referred to as your 'reference genome') and identify which of these names corresponds to this **Sequence Set**.

Alignment Blocks (green text)

Each block of aligned sequences should contain two or more sequences in aligned FASTA format followed by a line starting with an equal sign.

The header of each FASTA entry should follow the format:

```
> <genome_name>:<start>-<end> <strand> <Sequence_name>
```

- In the case of the reference genome, the full header is required. For all other aligned genomes the information following the <genome_name> is optional (one must either omit or include it all).
- <genome_name>: The entry should correspond to a genome name from the original headers. Spaces between the '>' and the genome name are ignored.
- <start>-<end>: For the reference genome, the entry must correspond to the **Sequence** coordinates (coordinates are inclusive, start with '1', and the <start> should be ≤ the <end>).
- <strand>: Must be either a '+' or '–'. Note that MochiView will adjust the alignment block such that the reference genome sequence is on the plus strand.
- <Sequence_name>: For the reference genome, this name must correspond to a **Sequence** name of the reference **Sequence Set**. For aligned genomes this entry is ignored.

An '=' sign is used to signal the end of each block. MochiView enforces the following additional formatting rules when importing the alignment file:

- Any block that does not include the reference genome is ignored.
- Each genome name can be used for at most one sequence entry per alignment block.
- The reference genome **Location**s covered by the alignment blocks cannot overlap.
- Each sequence entry in an alignment block must contain the same total number of characters (i.e. letters and hyphens).
- If the sequence in a block contains coordinate information, the number of bases (i.e. non-hyphen letters) in the sequence must agree with the length specified by the coordinates. This is important because MochiView will adjust these coordinates if the block is split into smaller sub-**Location**s (see '*Stored in Database*' section below).
- MochiView verifies that the reference genome sequence in the database at the **Location** coordinates and specified strand agree with the sequence provided in the alignment block entry.

Comments

- Import of alignments for very large genomes (> ~150MB) and/or a large number of genomes (~> 15) is not supported, and will likely result in out-of-memory errors or sluggish display.
- While the terminology of the alignment import is framed in terms of "genomes", the format above can also be utilized to import smaller alignments (e.g. reformatted BLAST results).

Stored in Database

The alignment is stored as a **Location Set** of the special **Location Type** '*Alignment Block*'. Each block is stored as a separate **Location**. In the case of very large blocks (over ~100kb), the block will be broken up into sub-**Location**s to speed database access.

Import→Location Set (alignment)→Format: Mauve

File Format

This utility associates a progressive alignment from the software program Mauve[6] (http://asap.ahabs.wisc.edu/mauve/) with a **Sequence Set** (referred to as the reference genome). The file must be in the non-standard XMFA format (http://asap.ahabs.wisc.edu/mauve-aligner/mauve-user-guide/mauve-output-file-formats.html) that is produced by the Mauve program. The file format and import requirements are fairly similar to those described for the generic *'Import →Location Set (alignment)→Format: FASTA'* utility, with one key distinction: Mauve concatenates chromosomes prior to alignment, and MochiView must be able to map the coordinates from the concatenated reference genome **Sequence Set** back to individual **Sequence**s.

Reference genome requirements

In order to successfully import the Mauve alignment, the reference genome must be concatenated in alphabetical order by **Sequence** name. Please note that MochiView alphabetizes the sequences in standard fashion, except that in cases where two names only differ by a trailing number the lower number is sorted first (i.e. '`chr20`' comes after '`chr3`'). If you are unsure of the alphabetical ordering of your **Sequence**s, just open up the *Location Browser* in a plot and observe the order in the **Sequence**-selection drop-down box.

Stored in Database

As described for '*Import Alignment*'.

Import→Data Set→Format: MochiView (by Location)

Required Headers

SEQ_NAME, START, END (1+ data columns with unique headers)

Optional Headers

STRAND, GROUP

File Format

The entry in the 'SEQ_NAME', 'START', 'END', 'STRAND', and 'GROUP' columns are interpreted as described for *'Import →Location Set→Format: MochiView'*. All entries in the file must exist in the same **Location Set**, but multiple rows referring to the same **Location** are permitted. In this latter case, the median of the provided values is entered into the database.

Every additional column must contain a unique header that will be suggested as the default name for that **Data Set**. In these columns each row should contain either a numerical value or be left blank. Keep in mind the following rules for the three different **Data Type**s:

1. '*Decimal*' **Value Type**s support values from -99,999 to 99,999 and are precise out to three decimal places. Attempts to import values that do not fall within this range will result in an error (try log-transforming such data).
2. '*Integer*' **Value Type**s support values from -2,147,483,648 to 2,147,483,647.
3. Values associated with the '*Boolean*' **Value Type**s should be represented as either 1 (i.e. "true") or 0 (i.e. "false").

Menu Options

The menu options are supported by numerous tooltips that will assist you. When you select your file, the header is scanned and the import menu will provide a list of all potential data columns. You must then use the provided pull-down menus to assign a **Data Type** with each **Data Set** column. This **Data Type** must contain a **Value Type** that corresponds to the format of the values in that column.

Stored in Database

The database stores the annotated **Data Set** (*name,* **Location Set**, **Data Type**) plus the map of **Location**s to values. (Currently, the only way to associate a description with a **Data Set** is to use the **MANAGER** menu.)

Import→Data Set→Format: MochiView (by gene name)

Required Headers

GENE, (1+ data columns with unique headers)

File Format

The file format is as described for *'Import→Data Set→Format: MochiView (by Location)'*, except that the **Location** is determined using the 'GENE' column rather than the 'SEQ_NAME', 'START', and 'END' columns. The lookup methodology is described in the section describing the *Gene Browser*.

GENE (required; max 50)

The entry should correspond to the name of a Gene in the selected gene **Location Set**. If the provided gene name does not provide a unique match, it is ignored. Note that this is different than the behavior of the other methods of **Data Set** import, which consider a **Location** entry that does not match the selected **Location Set** an error and abort the import.

Stored in Database

See description for *'Import→Data Set→Format: MochiView (by Location)'*.

Import→Data Set→Format: Agilent probe file

Note

This utility is hidden by default (this can be changed in *Utilities→Preferences*).

Required Headers

NAME, P-VALUE, P[XBAR], NORMALIZED LOG RATIO, EXCLUDE (all found in the default Agilent probe output file)

File Format

This is a legacy importer designed to read the 'Probe' file output format of the Agilent 'Chip Analytics' software (v1.3.1). The **Location**s are extracted from the 'NAME' column, and must map to an existing **Location Set**. This **Location Set** should correspond to the set of probes used in the ChIP-chip array (which must be imported using the *'Import→Location Set'* menu).

Menu Options

The menu is as described for *'Import→Data Set→Format: MochiView (by Location)'*, with a few important differences in the import process:

- All columns other than the required headers are ignored
- Rows in which the EXCLUDE entry is '1' are ignored (as are rows for control spots)
- The values contained in the columns 'P[XBAR]' and 'P-VALUE' are –$\log_{10}$ transformed upon import
- The import menu enforces the requirement that the associated **Data Type** must be of **Value Type** '*Decimal*'

Import→Location/Data Set→Format: MochiView

Required Headers

SEQ_NAME, START, END, (1+ data columns with unique headers)

Optional Headers

STRAND, ANNO_TAG, ANNO_DESC, GROUP

File Format

The file format is as described for *'Import→Data Set→Format: MochiView (by Location)'*. The difference in this menu is that the **Location Set** is created at the same time as the **Data Set**. This is particularly useful when the **Location Set** is a 'one-off' that is not intended for reuse with other **Data Set**s.

Stored in Database

As described for *'Import→Data Set→Format: MochiView (by Location)'* and *'Import→Location Set→Format: MochiView'*.

Import→Location/Data Set→Format: BED

File Format

This import utility utilizes the 'BED' file format (http://genome.ucsc.edu/FAQ/FAQformat#format1) to create a **Location Set** and **Data Set**. The **Location** and data are extracted from the following columns:

- *Column#1 (chrom):* Equivalent to the standard MochiView 'SEQ_NAME' column.
- *Column#4 (chromStart) and Column#5 (chromEnd):* Similar to the standard MochiView 'START'/'END' columns, except that: (a) the entry in column#4 must be lower than the value in Column#5 and (b) the entry in column#4 begins with a '0' coordinate and the entry in column#5 is (exclusive). In other words, a **Location** represented as 1-100 in MochiView is entered as 0-100 in this format.
- *Column#6 (strand):* Equivalent to the standard MochiView 'STRAND' column
- *Column#5 (score)*: Data values for the **Data Set**

The file is assumed to have a one line header of some sort (the first line is ignored). Import of *Location Annotations* is not supported when using this file format.

Import→Location/Data Set→GFFv3 (by Type)

File Format

This utility is designed for the extraction of various types of data from a General Feature Format (version 3) file. The entries in the first column (*seqid*) must match the name of **Sequence**s in the chosen **Sequence Set**. (The relevant columns for extracting **Location** coordinates are fully described in the entry for the *'Import→Location Set→GFFv3'* utility.)

Usage

This utility extracts a **Location Set**, and **Data Set** (and possibly a **Sequence Set** if the file also contains FASTA sequence) from a GFF file. The utility functions in several stages:

[1] Request and scan GFF File

You will be prompted for the location of a GFF file, which will then be scanned for its contents.

[2] Select Feature Types to include in Location Set

You will be given a list of the *feature types* (column#3) found in the GFF file. This list is limited to those entries that are not children of another entry (i.e. Entries that do not have the *Parent=xxx* attribute in column#9) and is further limited to those *feature types* that have at least one row with a *score* (column#6)

[3] (conditional) Sequence Set query

If your GFF file contains FASTA-formatted sequences, you will be asked whether you would like to create a new **Sequence Set** from these sequences or whether you would like to use an existing **Sequence Set**.

[4] Annotation entry

This menu will contain the following tabs:

Location Annotation

The **Location**s in the **Location Set** can be annotated with *Tags* and *Descriptions* using the settings chosen in this tab (see the description of *Annotation Tags* and *Descriptions* in the description of **Location Set**s in the '*Data Categories*' of this manual). You are given a pull-down menu with options of either setting the *Tag* as the *feature type* (column#3) or the values associated with the available attribute keys found in your file. Similar options are given for the *Annotation Description*. In both cases, if the text is too long it is truncated to the maximum size.

Sequence Set

If you elected to create a **Sequence Set** from the contents of the file in step [3], this tab will request annotation information and resemble the *'Import→Sequence Set→Format: FASTA'* utility. Otherwise, you will be given a pull-down menu from which you can choose the **Sequence Set** that matches the data in your file.

Location Set

This tab requests annotation information for your **Location Set**, and resembles the *'Import→Location Set→Format: MochiView'* utility.

Data Set

The contents of this tab request annotation information, and require that you choose a **Data Type** that is compatible with the data in your file (i.e. do not choose an *Integer* or *Boolean* data type if the file contains non-integer numbers).

Gotchas/Limitations

Support for the format is limited

This utility does not recapitulate the full complexity of the data encoded in the GFF format. In particular, it bears repeating that the utility has no knowledge of the hierarchy of Sequence Ontology terms, and the utility ignores all child terms (those with a *Parent=xxx* attribute entry).

Duplicate Locations

Extracted **Location**s that contain the same *Start* and *End* coordinates (and **Sequence**) will be considered duplicates, and only one will be added to the **Location Set** (at random). If such duplicates are found, it is noted in the black/white console display during import.

Un-Stranded Locations

Finally, please note that MochiView interprets any entry in the *strand* column other than '-' as a plus-strand **Location** (i.e. MochiView does not allow "unknown" or "un-stranded", and interprets these as plus-strand).

Import→Location/Data Set→Format: Agilent segment file

Note

This utility is hidden by default (this can be changed in *Utilities →Preferences*).

Required Headers

CHROMOSOME, START, END, MIN P[XBAR] (all found in the default Agilent segment output file)

File Format

This is a legacy importer designed to read the 'Segment' files output format of the Agilent 'Chip Analytics' software (v1.3.1). Each 'Segment' is a **Location** that corresponds to a predicted binding site, and the associated value is the –log10 transformed p-value. The import process is the same as that used for '*Import Location/Data Set*', with the following modifications:

- All columns other than the required headers are ignored
- The values contained in the column 'MIN P[XBAR]' are $-\log_{10}$ transformed upon import
- Segments are truncated to eliminate extension off the end of the **Sequence**
- The import menu enforces the requirement that the associated **Data Type** must be of **Value Type** '*Decimal*'

Import→Tiled Set→Format: WIG

File Format

This utility is designed for the import of files in the UCSC Genome Browser[7] wiggle format (as described at http://genome.ucsc.edu/goldenPath/help/wiggle.html). To be compatible with MochiView, the '`WIG`' file must observe a few additional constraints:

- The name in the '`chrom=<name>`' tag in the header lines must correspond to the name of a **Sequence**. Names with spaces in them will not be recognized unless they are surrounded by quotes!
- At most one header per **Sequence** is allowed, and all must have the same span.
- All rows under a given header must be in ascending order (by position).
- No two positions can overlap with each other (after taking span into account).
- No more than ~700 **Sequence**s can be included in a single file

The track definition line (the first line in the file, which must start with the word *track*) is required, but the information within is currently not used by MochiView.

Data Values

It is strongly recommended that you do not use values that span zero (i.e. both negative and positive values in the same file). Why? When you zoom out the display will be misleading, because MochiView pre-calculates values for a series of larger span-sizes (either the average or the largest deviation from zero) for the zoomed-out display.

Overview

Tiled Sets are an effective means for displaying large and dense data in MochiView plots while reducing the required hard drive space and maintaining the ability to smoothly scroll/zoom. As the user zooms out, the plot switches from displaying the raw data to displaying pre-computed tiled averages of the data. The tradeoff for enhanced performance and compressed storage is that **Tiled Set**s are only available for plotting, and cannot be used with the *Data Browser* or MochiView's many utilities.

Please note that this utility can take quite some time to run if the WIG file is very large (e.g. human genome 1bp-resolution tiling) and can make your computer quite sluggish during this period.

Menu Options

The first several options are similar to those associated with the many **Data Set** import utilities, and are not described here. The panel titled '*Tiled Set storage settings*' is described in the section in the manual titled *'Common Menu Elements'*.

Import→Tiled Set→Batch Import [Format: WIG and MochiView]

Overview

This utility allows import of multiple **Tiled Set**s at once by creating a *settings file* and placing it in a directory with the 'WIG' files that will be imported. The format for the 'WIG' files is described above (*'Import→Tiled Set→Format: WIG'*). Note that you need to create the **Data Type**(s) ahead of time using *'Utilities→Create Data Type'* (or use a pre-existing **Data Type**).

Format: Settings File

When you launch this import utility you will be prompted to select a setting file. This file contains all of the information that would normally be configured in the **Tiled Set** import utility (*'Import→Tiled Set→Format: WIG'*) using a tab-delimited layout in which the first row is a header row and each successive row is an entry for a **Tiled Set**. The order of the columns is not important, but certain headers are required:

Required Headers

SEQUENCE_SET, FILE_NAME, DATA_TYPE, NAME

SEQUENCE_SET

The name of the **Sequence Set** in your database that corresponds to the **Tiled Set** data (case-sensitive).

FILE_NAME

The name of the 'WIG' file containing the **Tiled Set** data. The file *must* be in the same directory as the settings file, and that the name provided should not include path information (e.g. data.wig, *not* C:\folder\data.wig).

DATA_TYPE

The name of the **Data Type** in your database that will be assigned to the **Tiled Set** (case-sensitive). The **Data Type** cannot be of **Value Type** *Boolean*.

NAME (max 50)

The name that will be assigned to the **Tiled Set** (must not match an existing name).

Optional Headers

DESCRIPTION, PROJECT, COMPRESSION, SIGN, LOG2, MULTIPLIER, ZOOM_VALUES

DESCRIPTION (max 500)

A description of the **Tiled Set**.

PROJECT

Assigns the **Tiled Set** to the named **Project** (case-sensitive). If left blank the **Tiled Set** is assigned to the *Global* **Project**.

COMPRESSION

The level of data compression, indicated by the text 'LOW', 'MEDIUM', or 'HIGH' (no quotes). If the column is not included, the default compression is 'LOW'.

SIGN

The sign-adjustment, indicated by the text '+', '-', or a blank (no adjustment). If the column is not included, the sign is not adjusted.

LOG2

Indicate whether a $\log_2$-transformation should be applied by entering either 'T' (true) or 'F' (false). If the column is not included no transformation is applied.

MULTIPLIER

Indicates whether a multiplier should be applied to the values. Enter a valid number or leave blank for no multiplier. If the column is not included no multiplier is applied.

ZOOM_VALUES

Enter either 'A' (average) or 'Z' (largest deviation from zero) to indicate which approach should be taken to merging values when pre-calculating large span sizes. If the column is not included the default is to use the average.

Import→Tiled Set→Format: Eland Extended

Overview

This utility allows for the conversion of aligned sequence reads into the MochiView **Tiled Set** format, with a tiling span chosen by the user (default is 1bp). A tally of "counts" is kept for each tile and, for each aligned read, the count is incremented for each overlapped tile. Plus-strand and minus-strand **Tiled Set**s can be created simultaneously, as can a **Tiled Set** that includes counts from both strands.

Note

This utility isn't exactly speedy… creating plus- and minus-strand **Tiled Set**s from a large sequencing run can take an hour or more. Also, be advised that the utility needs hard drive space to create temporary files, so make sure your hard drive isn't close to capacity.

File Format: Overview

This utility is designed for the import of ChipSeq data in the Solexa 'Eland Extended' format. The utility can handle very large files (gzipped is OK, provided the file name ends with the suffix .gz). Currently, the most likely scenario under which one could run out of memory is if a *very* large number of reads is concentrated in a small portion of the genome. Please let me know if you have trouble importing a file… there are ways to get around this problem, but I can't devote the necessary time at the moment.

File Format: Sequence names

Each read in the 'Eland Extended' format is assigned a chromosome name, typically the name of the file associated with the chromosome (e.g. *Chr11.fa.gz*). MochiView attempts to match this name (case-insensitive) to the **Sequence** names in the chosen **Sequence Set** as follows:

- Name before the first period (e.g. '*Chr11*' from '*Chr11.fa.gz*')
- Full name
- Name preceding last period (e.g. '*Chr11.fa*' from '*Chr11.fa.gz*')

The upshot is that you should make sure that the files used to produce the 'Eland Extended' file are compatible with your MochiView **Sequence** names.

Menu Options: 'Settings' tab

Options are described in the order in which they appear.

Select a file

Select a file of format 'Eland Extended' (gzipped is OK, provided the file name ends with the suffix .gz).

Select Sequence Set

Select the **Sequence Set** that corresponds to the contents of the file (see note above about **Sequence** name compatibility).

Select Project (optional)

You can assign the **Tiled Set**(s) to a **Project** (for organizational purposes).

Select Data Type

The **Data Type** must be of **Value Type** *Integer* or *Decimal* (the latter if you are using $\log_2$-transformation).

Tiled Set Settings: tiling span

This setting determines the tile size (bp) for the **Tiled Set**(s). The default of 1bp works well for most cases.

Tiled Set Settings: data compression

Tiled Set Settings: apply Log_2 transformation

Tiled Set Settings: when zoomed use deviation from zero

These options are described in the section in the manual titled *'Common Menu Elements'*, in the subsection titled '*Tiled Set storage settings*'. (If in doubt, just leave it on *Low*.)

Eland Import Settings: allow single base pair mismatch

Each line in the file has an entry in the format '`#:#:#`', where the numbers correspond to the number of perfect matches, 1bp mismatches, and 2bp mismatches respectively. If '*allow single mismatch*' is selected, lines with a '`0:1:#`' entry are considered for import (see details below).

Eland Import Settings: location match read length (bp)

This setting controls the size of the location match, starting from the first coordinate of the mapped read and extending in a strand-dependent direction. All tiles overlapping the extended read have their count incremented by one.

Eland Import Settings: distance of read shift (bp)

This option is provided for import of ChIP-Seq data, in which the shearing may necessitate a strand-dependent shift. MochiView does not currently have the means to auto-detect the necessary shift, so you will either need to determine this visually following un-shifted import or determine the shift using another tool.

Menu Options: 'Tiled Set(s)' tab

Options are described in the order in which they appear.

Choose which Tiled Set(s) to extract

You are given the option to create up to three different **Tiled Set**(s), differing in whether they include plus-strand matches, minus-strand matches, or both. In the case of the minus-strand matches, your are given the option to apply a negative-sign to the final values, which helps facilitate plotting plus- and minus-strand data on the same track.

Annotation for Tiled Set(s)

Enter a unique name for those **Tiled Set**(s) that were checked in the menu above.

Sequence match determination

It is important to note that the '`Eland Extended`' format may report a certain number of matches in the '`#:#:#`' column (described in the '*allow single mismatch*' section) and provide a differing account of mismatches in the final column. This is because the '`#:#:#`' column searches a maximum of 32bp of sequence, but the user can elect to get mismatch information about a longer sequence. MochiView uses a two stage selection criteria to identify valid matches. The first stage uses the '`#:#:#`' column. Any line that does not have at least one exact match (or at least one 1bp mismatch if 'allow single mismatch' is selected) is skipped. Next, the match data in the final column are analyzed as follows:

1. Search for exact matches, and declare match if only one is found.
2. Search for 1bp or 2bp mismatches with at least 32bp continuous matching sequence, and declare match if only one is found.
3. Search for single-mismatch hits, and declare match if only one is found.

Import→Tiled Set→Batch Import [Format: Eland Extended]

Overview

This utility allows import of multiple '`Eland Extended`' files at once by creating a *settings file* and placing it in a directory with the files that will be imported. The format for the files is described above (*'Import→Tiled Set→Format: Eland Extended'*). Note that you need to create the **Data Type**(s) ahead of time using *'Utilities→Create Data Type'* (or use a pre-existing **Data Type**).

Format: Settings File

When you launch this import utility you will be prompted to select a setting file. This file contains all of the information that would normally be configured in the import utility (*'Import→Tiled Set→Format: Eland Extended'*) using a tab-delimited layout in which the first row is a header row and each successive row is an entry for a **Tiled Set**. The order of the columns is not important, but certain headers are required:

Required Headers

SEQUENCE_SET, FILE_NAME, DATA_TYPE, TS_PLUS, TS_MINUS, TS_BOTH, READ_LENGTH, ALLOW_MISMATCH

SEQUENCE_SET

The name of the **Sequence Set** in your database that corresponds to the **Tiled Set** data (case-sensitive).

FILE_NAME

The name of the '`WIG`' file containing the **Tiled Set** data. The file *must* be in the same directory as the settings file, and that the name provided should not include path information (e.g. `data.wig`, *not* `C:\folder\data.wig`).

DATA_TYPE

The name of the **Data Type** in your database that will be assigned to the **Tiled Set** (case-sensitive). The **Data Type** cannot be of **Value Type** *Boolean*.

TS_PLUS

Leave blank if you do not want to create a **Tiled Set** from plus-strand reads. Otherwise, supply a unique name (50 characters max), which will be assigned to the **Tiled Set**.

TS_MINUS

As for TS_PLUS, but for creation of a **Tiled Set** using minus-strand reads.

TS_BOTH

As for TS_PLUS, but for creation of a **Tiled Set** using both minus- and plus-strand reads.

READ_LENGTH

Indicate the location match read length using an integer between `1` and `10,000`.

ALLOW_MISMATCH

Indicate whether single-base mismatches are allowed by entering either '`T`' (true) or '`F`' (false).

Optional Headers

TS_PLUS_DESC, TS_MINUS_DESC, TS_BOTH_DESC, PROJECT, COMPRESSION, TS_MINUS_NEGATIVE, LOG2, ZOOM_VALUES, READ_SHIFT, TILING_SPAN

TS_PLUS_DESC (max 500)

A description of the **Tiled Set** made from plus-strand reads (ignored if 'TS_PLUS' is left blank).

TS_MINUS_DESC (max 500)

A description of the **Tiled Set** made from minus-strand reads (ignored if 'TS_MINUS' is left blank).

TS_BOTH_DESC (max 500)

A description of the **Tiled Set** made from both plus- and minus-strand reads (ignored if 'TS_BOTH' is left blank).

PROJECT

Assigns the **Tiled Set** to the named **Project** (case-sensitive). If left blank the **Tiled Set** is assigned to the *Global* **Project**.

COMPRESSION

The level of data compression, indicated by the text 'LOW', 'MEDIUM', or 'HIGH' (no quotes). If the column is not included, the default compression is 'LOW'.

TS_MINUS_NEGATIVE

Indicate whether a negative sign should be added to the counts in the **Tiled Set** made from minus-strand reads by entering either 'T' (true) or 'F' (false). If the column is not included, the sign is not adjusted. (Ignored if 'TS_MINUS' is left blank.)

LOG2

Indicate whether a $\log_2$-transformation should be applied by entering either 'T' (true) or 'F' (false). If the column is not included no transformation is applied.

ZOOM_VALUES

Enter either 'A' (average) or 'Z' (largest deviation from zero) to indicate which approach should be taken to merging values when pre-calculating large span sizes. If the column is not included the default is to use the average.

READ_SHIFT

Indicate the read shift (bp) using an integer from 0 to 10,000. If this column is omitted no shift is applied.

TILING_SPAN

Indicate the tiling span (bp) using an integer from 1 to 10,000. If this column is omitted the default span is 1 bp.

Import→Motif→Format: MochiView

File Format: Overview

This format requires a tab-delimited file containing one or more **Motif**s in 'ACGT frequency' format. The name of the **Motif** must be unique, and is represented in the header row for each **Motif** (see below). Each successive row corresponds to a position in the **Motif** and should contains four tab-delimited numbers representing the values associated with the 'A', 'C', 'G', and 'T' bases in that position (the frequencies must add up to ~1.0). Any line beginning with '#' is ignored.

File Format: Header Row

The header row indicates the start of a new **Motif**, and should begin with a forward slash. MochiView supports both a *Simple* and *Detailed* header format. The *Simple* format assumes that the **Motif** is a PSFM, and takes all text following the '/' to be the name of the **Motif** (maximum of 50 characters). The *Detailed* header is used for PSAM **Motif**s as well as PSFM **Motif**s that have additional information embedded. The format of the *Detailed* header is the forward slash followed by one or more attribute=<value> pairs, each of which is separated by a space. The case of the attributes does not matter, nor does the order, with the exception of the DESCRIPTION attribute, which must come last (if at all). The only mandatory attribute is NAME .

- NAME=<motif_name>: The name of the **Motif** (maximum 50 characters). This is the only mandatory attribute.
- TYPE=<PSFM|PSAM>: Either 'PSFM' for frequency matrices or 'PSAM' for affinity matrices. The default is 'PSFM'.
- CUTOFF=<value>: Recommended score cutoff for the **Motif**. This will be the default cutoff for the **Motif** in the **NEW PLOT** menu as well as many other utilities. This entry (and the BGFREQ entry described below) can be adjusted in the **MANAGER** by double-clicking on a **Motif** and looking in the '*Advanced*' tab of the summary window.
- BGFREQ=<pipe-delimited_frequencies>: This entry is only relevant for PSFM **Motif**s, and is ignored for PSAM **Motif**s. The background frequencies included here override any background frequency assignments made within MochiView (useful if you would consistently like to use the same frequencies for all analyses).
- SPECIES=<species_name>: For annotation purposes, the **Motif** can be associated with a source species. The name can use one of three formats: [1] S. cerevisiae, [2] S.cer, [3] S_cer. If the name does not match a species in MochiView it is ignored (feel free to write the author and request additional species). The import menu offers a pull-down menu that can

globally assign source species to the imported **Motif**s. If the species is specified in the header the entry from the pull-down menu is ignored. Currently, assigning a **Motif** to multiple species is not supported.

- `DESCRIPTION=<text>`: This attribute must come last (if at all), and can contain a description (up to 500 characters) for the motif.

File Format: Position Rows

The rows following the header correspond to successive positions in the **Motif** and provide tab-delimited numbers representing the values associated with the 'A', 'C', 'G', and 'T' bases in that position. The values must be between 0 and 1. In the case of PSFM **Motif**s, these values are frequencies and must add up to ~1.0. In the case of PSAM **Motif**s at least one value in each row must equal 1.0.

Sample Format

Below, a sample six-position PSFM **Motif** is provided (and two additional partial **Motif**s to demonstrate PSAM format, alternative header formats, and the inclusion of multiple **Motif**s in the same file). Note that the comment lines are included for clarity and are not required in your file.

```
# PSFM Motif with simple header
/My Motif Name
0.06   0.84   0.04   0.06
0.06   0.04   0.04   0.86
0.06   0.04   0.84   0.06
0.66   0.04   0.04   0.26
0.86   0.04   0.04   0.06
0.86   0.04   0.04   0.06

# PSFM Motif with detailed header
/NAME=Another Motif TYPE=PSFM CUTOFF=2.0 BGFREQ=0.3|0.2|0.2|0.3
0.06   0.84   0.04   0.06
0.06   0.04   0.04   0.86
… <snip> …

# PSAM Motif with detailed header
/NAME=Affinity Motif TYPE=PSAM CUTOFF=0.05 SPECIES=H_sap DESCRIPTION=blah
0.40   0.06   1.00   1.00
1.00   0.52   0.13   0.91
… <snip> …
```

Menu Options

When importing the **Motif**s, options are provided to:

- Indicate the source species from which the **Motifs** were generated
- Associate the **Motifs** with a particular **Project**

The **Species** and **Project** entries are purely for annotation purposes.

Import→Motif→Format: MEME

File Format

Output file from the **Motif**-finding program MEME[1] in text format. The parsing routine was tested on MEME versions 3.5.0 and 4.1.0. Please let me know if future versions of MEME fail to parse correctly.

Menu Options

This menu option allows the user to selectively import **Motif**s from either a single MEME result file or an entire directory of MEME results files. Once the file/directory is chosen in the menu, the discovered **Motif**s are displayed as sequence logos and checkboxes are provided for the user to indicate whether the **Motif**s should be loaded.

The '*pseudocount*' settings are described with tooltips. In brief, these settings control whether the strength of the **Motif** should be down-weighted if the **Motif** was generated from a small number of sequences.

Import→Motif→Format: Bioprospector

File Format

Output file from the **Motif**-finding program Bioprospector[2] in text format. The parsing routine was tested on a results file obtained by e-mail using the Bioprospector web interface (http://robotics.stanford.edu/~xsliu/cgi-bin/BPsearch.cgi). Please let me know if future versions of Bioprospector fail to parse correctly.

Menu Options

See the description for the *'Import→Motif→Format: MEME'* utility.

Import→Motif→Format: XMS

File Format: Overview

This utility was designed to support the XMS file format, as used by the programs iMotifs and NestedMICA. MochiView expects that the file will only contain **Motif**s with a standard "DNA" alphabet. `<prop>` entries are copied over to the **Motif** description.

Menu Options

See the description for the *'Import→Motif→Format: MEME'* utility.

Import→Annotation→Format: MochiView (by Location)

Required Headers

SEQ_NAME, START, END, STRAND, ANNO_TAG, ANNO_DESC

File Format

Utilizes the same format described for *'Import→Location Set→Format: MochiView'*. However, in this case the **Location Set** must already exist and the **Location**s described by 'SEQ_NAME', 'START', 'END', and 'STRAND' must all be found within the **Location Set**.

Usage

This import utility allows the association of *Location Annotations* with one or more **Location**s in a **Location Set**. *Location Annotation* entries for **Location**s that previously had annotation will be replaced with the new annotation. This also allows the deletion of *Location Annotations* by leaving the 'ANNO_TAG' and 'ANNO_DESC' columns blank.

Import→Annotation→Format: MochiView (by gene)

File Format

Utilizes the same format described for *'Import→Location Set (genes)→Format: MochiView'*.

Usage

This import utility allows the user to update all aspects of one or more genes in a **Location Set** of **Location Type** 'Gene' with the exception of the underlying **Location** (i.e. the original entry from the 'SEQ_NAME', 'START', and 'END' columns. The primary use of this update utility is to keep gene descriptions, nomenclature, and isoforms up to date without having to delete the **Location Set** (and all associated **Data Set**s).

Import→Gene Ontology→Ontology

File Format

Utilizes an ontology file in 'OBO' format (v1.2).

Usage

If you could like to use the Gene Ontology (GO) enrichment analysis utilities, you must first use this import utility to import a file describing the GO Terms. The utility provides the option of either importing an 'OBO' file that you have on your hard drive or attempting to download the file. The latter option attempts to download the file from:

http://www.geneontology.org/ontology/obo_format_1_2/gene_ontology.1_2.obo

If this link is no longer valid you will receive an error message and will need to download/import the file yourself. Once the ontology file is imported, use the *'Import→Gene Ontology→Gene-to-Ontology Assignments'* utility (see below) to associate genes with GO Terms.

Import→Gene Ontology→Gene-To-Ontology Assignments

File Format

This utility utilizes files in the GO Annotation file format described at www.geneontology.org. This website provides such files for a wide variety of species at the following link:

http://www.geneontology.org/GO.current.annotations.shtml

When parsing the file, MochiView ignores any rows that do not have an entry of "gene" in the 'DB_OBJECT_TYPE' column, as well as any rows that have an entry of "NOT' in the 'QUALIFIER' column.

Usage

You must import the GO Term descriptions using the *'Import→Gene Ontology→Ontology'* utility (see above) before using this utility to associate the genes in a **Location Set** of **Location Type** '*Gene*' with GO terms. Just download the appropriate file from the link above, choose the gene set from the pull-down menu, and import the file. If this file is more up-to-date than the 'OBO' file loaded using the *'Import→Gene Ontology→Ontology'* utility, you may encounter an error warning you that a GO Term was not found. In this case you must first import an updated 'OBO' file.

Gene Names

During the import of Gene-To-Ontology assignments MochiView attempts to match the gene names in the file to those in your **Location Set**. MochiView uses the entries in the 'DB_OBJECT_SYMBOL', 'DB_OBJECT_ID', and 'DB_OBJECT_SYNONYMS' columns (in that order) to query the **Location Set** for a unique match to either the gene 'FEATURE_NAME' 'GENE_NAME', or 'ALIASES'. You will be warned if any entries in the file did not match your gene set. Note that it is not unusual for a small number of genes to not match, typically because they refer to mitochondrial or other special genes that are not included in your **Location Set**.

Import→Batch→Format: MochiView

This menu is tailored to allow rapid import of **Data Set**s in multiple different formats. The menu requests a *Settings File* and the location of the directory containing all **Data Set**s described in the *Settings File*.

File Format

The *Settings File* must be a tab-delimited file that contains the following column headers:

FILE NAME

The entry should contain the name of a file to be imported. Note that you should NOT include the full path, as you will select the file directory from the menu.

FILE TYPE

Must include one of the following keywords:

Custom

Use this keyword for files intended for the *'Import→Data Set→Format: MochiView (by Location)'* or *'Import→Data Set→Format: MochiView (by gene)'* menus. The distinction between '(*by Location)*' and '(*by Gene)*' is inferred from the entry in the **Location Set** column. (Note that this means that all **Data Set**s with **Location Set**s of **Location Type** '*Gene*' will be assumed to be in the 'by Gene' format.)

LocationAndData

For files intended for the *'Import→Location/Data Set→Format: MochiView'* menu.

AgilentProbe

For files intended for the *'Import→Data Set→Format: Agilent probe file'* menu

AgilentSegment

For files intended for the *'Import→Location/Data Set→Format: Agilent segment file'* menu

SEQUENCE SET NAME

The name must correspond to the name of a **Sequence Set** in the database.

LOCATION SET NAME

Format varies by FILE TYPE as follows:

'Custom' and 'Agilent Probe'

The entry must correspond to the name of a **Location Set** in the database.

LocationAndData

The entry will be used as the name of the newly created **Location Set** and must NOT correspond to an existing **Location Set** name (and must be 1-50 characters in length).

AgilentSegment

The entry is ignored (in this case the **Location Set** is created *de novo* as the segment file is parsed).

PROJECT NAME

Can only be blank (which is interpreted as '*Global*') for the FILE TYPEs '*Custom*' and '*LocationAndData*'. If the name is not blank and the **Project** is not in the database, it will be created. (Max length = 50 characters.)

PROJECT DESCRIPTION

This column may be left blank. This entry is only used if the 'PROJECT NAME' entry is not blank and is not already in the database. (Max length = 500 characters.)

DATA TYPE NAME

This column may be left blank for the FILE TYPEs '*Agilent Probe*' and '*Agilent Segment*' (a special **Data Type** is created for these formats). Otherwise, the DATA TYPE NAME must correspond to the name of a **Data Type** in the database.

Import→Database→Manage databases

See the entry for *'Database →Manage databases'* (both menu options the same utility).

'Export' Menu

This menu contains utilities that export information from the database. Note that images of plots can be exported using the '*Snapshot*' utility from within a plot (see the section titled *'Plot Window Overview'*).

Export→Sequence Set→Format: FASTA

This utility can be used to export **Sequence**s associated with a **Sequence Set** as a 'FASTA' file. The header of each entry is the name of the **Sequence**.

Export→Location Set→Format: MochiView (with optional Motif scoring)

This utility is identical to the *'Export→Location/Data Set→Format: MochiView (with optional Motif scoring)'* utility (see below) with the exception that export of data from **Data Set**s is not provided as an option. The output format is the same as the format required by the *'Import→Location Set→Format: MochiView'* utility.

Export→Location Set→Format: FASTA

This utility can be used to export sequences associated with a **Location Set** as a 'FASTA' file. Each **Location** in the **Location Set** is added to the file as a separate sequence, with a header that includes the **Sequence** coordinates (and, if available, the names of the two genes with start codons closest to the midpoint of the **Location**, using a maximum search distance defined in *Utilities→Preferences*). Filters and additional setting options allow the user to export subsets and/or sub-regions of **Location**s. (These filters are particularly useful for selecting sequence for import into MEME).

Select Sequence Set

Selecting a **Sequence Set** filters the list of available **Location Set**s in the next menu item.

Select Location Set

Select the **Location Set** that should be exported as FASTA-formatted sequence.

Constrain to a maximum length?

This panel is described in the section in the manual titled *'Common Menu Elements'*.

Sequence Direction

This radio-button menu allows the user to select whether directionality of **Location**s should be preserved when providing sequence or whether all sequence should be provided from the '*Plus*' strand or '*Minus*' strand.

Filters Tab

Filters can be applied to the selected **Location Set** to further customize the output.

Constrain by Data Set filter
Constrain by Location Set overlap filter

These options are described in the section in the manual titled *'Common Menu Elements'*.

Alignment Masking

When exporting sequence for use with **Motif**-finding programs (e.g. MEME[1]) it can be useful to mask bases that are not conserved among closely related genomes. This tab provides the option to use an existing alignment to enforce such masking (see the tooltips for details).

Export→Location Set→Format: Markov model (MEME-compatible)

This utility calculates an exports a N-order Markov model for the selected **Location Set** in a format that can be used by the motif-finding software MEME[8]. For more information on Markov models consult the section titled *'Scoring and Identifying Motifs'*.

Export→Location Set→Gene proximity assignments

Usage

This utility provides a highly flexible means for assigning one or more proximal genes to each **Location** in a **Location Set** (typically used for matching ChIP-chip binding regions in promoters to neighboring genes). The resulting assignments can then be exported to a tab-delimited file (two formats are offered) and/or applied at *Location Annotations*. (Note that *Location Annotations* will overwrite any existing ones!)

The utility is heavily documented with tooltips, and is also described in the manual section titled *'Common Menu Elements'* under the header *'Gene Proximity Assignment' panels*. It bears mentioning that the purpose of the optional filters is to reduce the number of **Location**s being considered, and the purpose of the '*Constrain Locations to a maximum length?*' option is to influence the application of the *Location proximity criterion*.

The third tab ("*Data Sets (optional)*") allows you to include columns in the output file that contain the associate values for selected **Data Set**s. The table in this tab updates to reflect the **Data Set**s associated with the currently selected **Location Set** in the pull-down menu in the first tab.

Export→Location Set→Mapped to different Sequence Set

Usage

This utility can either (1) use a supporting alignment to export the coordinates of a **Location Set** in a **Sequence Set** different than its own, or (2) export the coordinates of all **Location**s in the **Location Set** mapped to a different **Sequence Set** provided that the sequence at the **Location** in the source **Sequence Set** has a single unique match in the target **Sequence Set**. The first approach proceeds basically as described for the *'Utilities→Data Transfer→Transfer data between Sequence Sets'* utility except that the **Location Set** mapping is exported to a tab-delimited file (see below for format). The second approach requires no alignment and is enabled using the "use exact and unique matches only" checkbox. In either case this utility only works for small genomes (no more than 20 million bases) for memory-usage reasons.

Output File Format

The output format is a tab-delimited file in which each row (except the header) corresponds to one of the **Location**s in the **Location Set** being mapped.

The first four columns are always the coordinates for the source **Location** (`SEQ_NAME`, `START`, `END`, `STRAND`). If the exported **Location Set** has any annotations, the next two columns are the `ANNO_TAG` and `ANNO_DESC`. The next four columns are always the coordinates for the mapped **Location** (`SEQ_NAME`, `START`, `END`, `STRAND`). If alignment-based mapping is being used (approache#1, as described above), the next column indicates whether the source and mapped **Location** have a different DNA sequence (the header is `SEQUENCE CHANGED?`). The final header is always COMMENTS, and provides an explanation when a **Location** was not successfully mapped.

Export→Location Set (genes)→Format: MochiView

This utility exports a **Location Set** of **Location Type** '*Gene*' in the same format required by the *'Import→Location Set (genes)→Format: MochiView'* utility. An option to export protein sequences is also provided (only those genes with coding exons with a combined length that is a multiple of three are assigned a sequence).

Export→Location Set (genes)→Promoter sequence (from gene names)

This is a rapid way to retrieve a FASTA file of promoter sequences for use in other **Motif**-finding programs by simply selecting a **Location Set** of **Location Type** '*Gene*', configuring the promoter settings, and pasting in a list of gene names. (Check the readout as export proceeds for any notes on gene names that could not be unambiguously mapped to a gene in your selected **Location Set**.)

Export→Location Set (alignment)→Format: FASTA

This utility can be used to export any **Location Set** of **Location Type** *Alignment Block* in the same format that is read by the *'Import→Location Set (alignment)→Format: MochiView'* utility.

Export→Location/Data Set→Format: MochiView (with optional Motif scoring)

This utility exports a tab-delimited file in which each row corresponds to a **Location** in the chosen **Location Set**. Several optional features can be included in the output:

Include Location Annotations

If the chosen **Location Set** contains *Location Annotations*, they can be included in the output file as columns labeled 'ANNO_TAG' and 'ANNO_DESC'.

Include Data Set(s)

If the chosen **Location Set** has any assigned **Data Set**s, they can be included in the output file as columns labeled with the **Data Set** name.

Include Sequence

The user has the option of including a column that contains the nucleotide sequence for each **Location**.

Motif Scans (optional)

The settings in this tab allow the inclusion of additional columns in the export file that correspond to **Motif** match scores for each **Location** in the **Location Set**. See the tooltips and the section of the manual titled *'Scoring and Identifying Motifs'* for details. (Note that there is a special option for **Location Set**s of **Location Type** *Gene* that scores the promoters instead of the *Gene* **Location**s. This is a very easy way to create a large spreadsheet of **Motif** enrichment at each gene for an entire **Motif** library, which could then be clustered or included with expression data heat maps.)

Export→Tiled Set→Format: WIG

This utility exports a **Tiled Set** to 'wig' format of the type '*variableStep*'. You are given the option to round the exported value to 0-8 digits beyond the decimal. For very large **Tiled Set**s applying rounding can significantly reduce the file size. Moreover, since data compression occurs during **Tiled Set** import, it makes sense to round the values to the number of significant digits that were supplied in the original file.

Export→Motifs→PSSMs [Format: MochiView]

This utility exports all of the **Motif**s currently in the database as a tab-delimited text file in the format described for the *'Import→Motif→Format: MochiView'* utility (using the *Detailed* header format).

Export→Motifs→PSSMs [Format: XMS]

This utility exports all of the **Motif**s currently in the database as a tab-delimited text file in the format described for the *'Import→Motif→Format: XMS'* utility. Note that additional annotation information is not preserved upon export, and thus this is not the recommended export format if you intend to re-import your **Motif**s into MochiView.

Export→Motifs→Logo image [Format: PNG]

This utility exports logos for one or more **Motif**s using user-defined dimensions. The individual options are documented via tooltips.

Export→Motifs→Scan Location Set and export Motif matches

Usage

This utility exports a tab-delimited file in which each row corresponds to the coordinates, **Motif** score, and matching sequence for a motif hit.

Output File Format

The first line is a header with information about the scan configuration:

```
# Scan of motif(s) '<MOTIF_NAMES>' against '<LOCATION_SET or SEQUENCE_SET name> [P-Value|PSFM|PSAM
Cutoff:###] [Flank extension:###bp]
```

The second line contains the column headers and each subsequent row contains a **Motif** hit:

- **MOTIF_NAME:** Name of **Motif** for the hit entry (only included if more than one **Motif** is being scanned).
- **SEQ_NAME/START/END/STRAND:** These columns contain describe the hit **Location** in standard MochiView format.
- **SCORE:** The score for the hit.
- **P-VALUE (-log_{10}):** (Optional) The -log_{10}-transformed p-value for the hit.
- **SEQUENCE:** The sequence of the hit region (upper case) and any included flanking region (lower case) from the strand on which the hit was found.

Walkthrough

The steps required to configure the utility are:

[1] Configure the Sequence/Location Set to scan

Use the pull-down menus to indicate which regions of the **Sequence Set** you would like to scan for **Motif** hits. The choice of a **Location Set** is optional (it defaults to scanning the full **Sequence Set**). Note that the region scanned is the union of all **Location**s in the **Location Set** (this ensures that overlapping **Location**s are not scanned twice).

[2] Configure the Motif and Additional Settings panels

Choose the minimum score for inclusion of a **Motif** hit in the file based on either a maximum score or the -$\log_{10}$ p-value of the maximum score. Note that (1) for any given scan window only the highest scoring strand is considered, and (2) p-value calculations do NOT apply a multiple testing correction.

[3] Select Motifs to scan

Choose one or more **Motif**s to include in the scan.

Export→Database→Manage databases

See the entry for *'Database→Manage databases'* (both menu options the same utility).

'Utilities' Menu

This menu contains additional utilities that do not require external files.

Utilities→Motif→Finder→Source: Locations

Usage

While it is not intended as a substitute for more specific dedicated **Motif**-finding software, MochiView has its own built in **Motif** finder. Although the underlying algorithm is still a bit crude, the finder is surprisingly (at least to its author) quite effective. The recommended usage is to rapidly identify strong **Motif**s in your **Location Set**s, import them into the database, and then further characterize them using the *'Utilities →Motif →Enrichment Table'* and *'Utilities →Motif →Enrichment Plot'* utilities.

Here is a list of the main limitations of the '*Find Motifs*' utility:

- You must specify a specific **Motif** width to search. (However, you can easily run several searches using a variety of widths.)
- The finder uses a "~0.75 **Motif**s per sequence" model, and does not reward multiple occurrences of a **Motif** within the same **Location**.
- The finder does not ignore repeated sequence.
- The finder is quite slow when a large number of **Location**s (200+) is searched (you can cancel the search at any time).
- The larger the number of submitted **Location**s (and the longer the **Location** sequences), the more likely the algorithm is to get stuck in local optima and miss very strong hits for the first few passes. As a rule of thumb, if you are testing over 500 **Location**s, return 10+ **Motif**s and consider setting the search speed to '*Slow*'.
- The finder does not currently calculate a P- or E-value for putative motifs. (Instead, it is recommended that you evaluate the **Motif** using the enrichment utilities.)

Walkthrough

Every option in this menu contains a tooltip that you can consult for specific details. Here, the focus is on a basic overview of how to configure and run the utility.

[1] Configure your search

You will always want to select a **Location Set** from the pulldown menu. If you leave every other default in place, every **Location** within the **Location Set** will be treated as a sequence that is searched for **Motif**s. Alternatively, you have multiple filtering options (see the bottom of tab#1 and tabs #2 and #3) to reduce the number and/or size of the **Location**s being searched. The middle section of tab#1 contains multiple additional options for configuring your search (see their respective tooltips for details).

[2] Run your search

Once your search is configured, press the GO button to initiate the search. You will automatically be taken to the '*Results*' tab. This is where the **Motif**s will appear as they are found (see the bottom information ribbon of the utility for details on the search progress). You can stop a search at any time by pressing the '*Cancel Search*' button. The key features of the '*Results*' table are:

- **Motif**s found during a search will remain in the table, even if you cancel the search while it is in progress and even if you start a new search. (This way you can run searches with different parameters and quickly compare the results.)
- You can remove a **Motif** from the table by pressing the ⊗ button in the second column.
- You can add any **Motif** in the table to the database by clicking the ⊕ button in the first column of the table (you will be prompted to give the **Motif** a name).
- The default names of found **Motif**s in the table use the format '`Motif_#.#`'. The first number refers to the "search run" (each time you press GO you are initiating a new search run). The second number refers to the **Motif** number in the given

search run. If the **Motif** is a "refined" **Motif**, you will see an addition suffix of '`[Cull]`' or '`[Loop]`', for regular and iteratively refined **Motif**s respectively (see tooltips for details on refined **Motif**s).

How does it work?

A detailed explanation of the algorithm is provided in Appendix C. Here is a (basic) overview of how the **Motif** finding algorithm actually works:

1. Get **Location**s from the **Location Set** chosen by the user
2. Apply any filters or size refinement that were chosen by the user
3. Fetch the DNA sequences of the remaining **Location**s
4. For each **Motif** that the user chose to find (default is '5'):
   a. Refine several starting seeds (seed = "motif" built from random windows in each sequence) through a small number of iterations
   b. Choose the highest scoring seed to iteratively refine until a stable local optimum is found
   c. Output the **Motif** to the table
   d. If the user selected the option to create refined **Motif**s, take the found **Motif** and iteratively remove sequences that are reducing the score of the **Motif**.
   e. Mask the sequence windows that define the found **Motif** so that they are not used in the next **Motif** search. (Note that this masking isn't very effective against repeats, because it only masks one instance.)

I asked for five Motifs… why am I getting more (AKA what are refined Motifs)?

The settings tab contains an option called "*Enter Motifs to find (not including any refined Motifs)*". If you choose '`5`', the table will include five **Motif**s plus the refined and iteratively refined **Motif**s (provided you checked the checkboxes to include these options and provided that the refinement actually improved/changed the **Motif**). See the tooltips for these checkboxes for additional details.

I seem to be getting a lot of repeat sequences…

This problem is especially common if your **Location Set** is not particularly enriched for a "real" **Motif**, but it is always a concern. There are several options to minimizing this problem:

1. Try to further refine your **Location Set** (e.g. only search the highest confidence ChIP-chip binding regions).
2. Make sure you are using an appropriate Markov model.
3. Search for more **Motif**s. After each **Motif** is found the region containing the **Motif** is masked for the subsequent searches. In the case of repeats it may take a little while to mask the full repeat, but eventually the repeat-containing region will no longer be visible to the **Motif** search.
4. Use a better **Motif**-finding program ☺.

Utilities→Motif→Finder→Source: promoters

Usage

This utility is identical to the *'Utilities→Motif→Finder→Source: Locations'* utility described above, except the sequences being scanned are obtained in a different fashion. In this utility a tab is provided containing the necessary options to fetch sequences from promoters of a set of genes (as described for the '*Build Promoter Set*' utility).

Utilities→Motif→Finder→Source: sequences

Usage

This utility is identical to the *'Utilities→Motif→Finder→Source: Locations'* utility described above, except the sequences to be scanned are pasted into a text box by the user.

Utilities→Motif→Comparison

***TIP:** A walkthrough of this utility is provided in the tutorial*

*N.B.: If you have a very low number of **Motif**s in the database, don't put much stock in the E-values!*
N.B.: This utility is not recommended for motifs over ~50bp width (it will be very slow)

Usage

This utility allows the comparison of one or more "query" **Motif**s against a library of "target" **Motif**s. The output displays the alignments of query **Motif**s against target **Motif**s and an associated E-value describing the expected number of matches among the target **Motif**s. The results can be saved to a tab-delimited file, and images of aligned **Motif**s can be saved as '`png`' images.

This utility is heavily indebted to an article by Gupta *et al.*[9] that describes a strategy for calculating similarity between **Motif**s and calculating E-values. Please refer to the article for details on how the E-values are calculated (MochiView uses the "Euclidean Distance" similarity metric described in the article).

While the utility is reasonably effective for PSAM **Motif**s as well, the approach taken is rather imprecise. Specifically, the PSAM matrix is converted to an approximate PSFM matrix and then analyzed as if the **Motif** used a PSFM.

Walkthrough

The first tab in the utility contains options for configuring the comparison settings, and the second tab displays the results of your current search. You can stop a search at any time by pressing the '*Cancel Comparison*' button. Most features are well-documented with tooltips… a basic walkthrough is provided below.

[1] Select "query" and "target" Motifs

The top table in the '*Settings*' tab allows you to select one or more query **Motif**s. Each of these **Motif**s will be searched against all of the **Motif**s that you select in the bottom table.

[2] Decide if you want the "slow and exhaustive" approach

The bottom of the '*Settings*' tab includes a checkbox that tells the utility to calculate E-values even for poor matches (E-value > 10.0). This is off by default because these matches tend to be uninteresting and the algorithm can be sped up by an order of magnitude by ignoring these. However, if you want a complete picture of **Motif** similarity, you can enable this option.

[3] Start and view the analysis

Press the GO button to initiate the analysis, and you will immediately be switched to the '*Results Display*' tab. As each query **Motif** is analyzed (see the ribbon at the bottom for progress details) it will appear in the list on the left-hand side of the screen. Next to the **Motif** name is text with three numbers (`##/##/##`). These numbers provide a quick read-out of the number of target **Motif**s that yielded E-values of `0.1`, `1.0`, and `10.0`, respectively. Click the name of a **Motif** and the table on the right side of the screen will display target **Motif**s with an E-value ≤ `10.0` (if you checked the checkbox in the settings table, you will instead see *all* target **Motif**s). Note that the table displays the E-value for each match, and sometimes includes the text '`[REV]`' to mark cases in which the best match was obtained by reversing the target **Motif**.

Why isn't there reciprocity in my "target vs. query" E-values?

Just because query **Motif**#1 gives an E-value of 0.5 against target **Motif**#2, you may not see a similar score for the reciprocal comparison (i.e. when **Motif**#2 is the target and **Motif**#1 is the query). This is because the comparison approach uses the query **Motif**'s specific data to build a scoring matrix (see Gupta *et al.*[9] for details).

Utilities→Motif→Enrichment table

***TIP:** A walkthrough of this utility is provided in the tutorial*

Use this utility to characterize the distribution of **Motif**(s) across one or more **Location Set**(s). For example, you can ascertain whether any **Motif**s in the database are more frequent in your ChIP-chip binding regions than they are in the set of all promoter sequences.

Usage

Select one or more **Motif**s and one of more **Location Set**s and then press the GO button to initiate the analysis. Using the radio buttons in the **Location Set** table, you can (optionally) select a control **Location Set** that will be compared against all other selected **Location Set**s to determine whether the **Motif** distribution is significantly different.

Depending on the number of **Motif**s and the size of the **Location Set**s being scanned, the actual scan may take quite some time. The data are displayed in the '*Results*' tab and updated as the scan proceeds. You can cancel the scan at any time. The completed results can be saved to a tab-delimited file using the button in the '*Results*' tab.

Description of Results

The key columns in the table are the ones titled 10%, 20%, 30%... 100%. The numbers represent the number of *n-mer* windows (where '*n*' is the size of the **Motif**) scanned divided by the number exceeding a **Motif** score of ##% of the maximum possible score (which varies depending on **Motif** and background frequencies for PSFM **Motif**s but is always `1.0` for PSAM **Motif**s). For any given window, only the best score on either the plus or minus strand is considered (i.e. you won't get two hits in a window for a palindromic **Motif**). You can toggle between displaying the total number of '*hits*' and the '*sites per hit*' in the table (the former does not account for the size of the sequence scanned).

P-values

If your scan includes a control **Location Set**, you can also toggle the view to show the -log$_{10}$-transformed p-values once the scan is complete. This p-value reflects a one-tailed Fisher's Exact test comparing the number of scan windows passing and not passing the cutoff for the control and test **Location Set**s. This statistic is only intended as a rough indicator of likely enrichment of the **Motif** in the test set relative to the control set, and does not implement any correction for multiple testing. Also, please note that if over 20 million windows are scanned the p-value is not calculated (for computational reasons) and a value of 'N/A' is given instead ('N/A' is also shown for the rows in the table corresponding to the control dataset),

Usage: Alignment Constraint (optional)

If you have imported a genome alignment, it is possible to constrain the criteria for a **Motif** hit by selecting one or more aligned genomes and choosing the '*N^{th} highest score*' parameter. This is a rather confusing concept that is best explained by example. If four aligned genomes are selected and the '*N^{th} highest*' value is set to '`3`', MochiView does the following at each *n-mer* window scanned:

1. Calculate the score for the reference genome (i.e. your genome of interest that the other genomes are aligned against).
2. Calculate the score for the four selected aligned genomes (if aligned sequence is unavailable, the score is *-infinity*).
3. Sort the four scores of the aligned genomes in descending order and take the 3rd score as the aligned genomes score.

4. Take the lower of the reference genome score and the aligned genomes score (from step#3) as the final score and enter as a hit in the table if the score qualifies (i.e. is >= 10% of max score).

If an alignment constraint is implemented, you can toggle the results table between the alignment-constrained and unconstrained results using the checkbox in the lower panel of the '*Results*' tab.

Utilities→Motif→Enrichment plot

Usage

This utility creates a plot for each selected **Motif** in which each line represents a selected **Location Set** and represents the percentage of **Location**s (y-axis) that pass a range of **Motif** score cutoffs (x-axis). This is useful for calibrating a score cutoff for a **Motif** of interest such that the cutoff includes the majority of expected binding regions.

ROC plot

In addition, the utility can create a plot directly comparing the results from a **Location Set** designated as the "query" set against all other selected **Location Set**s. This plot is displayed next to the primary plot, and displays a line for each combination of query set and other **Location Set**, in which the primary plot y-axis values for each of the two sets are plotted for each primary plot x-axis value, similar to a Receiver Operating Characteristic (ROC) plot. In this plot the percentage of query set **Location**s passing the cutoff is considered the "True positive rate" (x-axis) and the percentage of the other set's **Location**s passing is considered the "False positive rate" (y-axis).

P-values

The utility also gives the option to designate a control **Location Set**. If a control is designated, dashed lines reflecting -log$_{10}$-transformed p-value will appear in the main plot (not the ROC plot) for every **Location Set** other than the control set. The values correspond to the y-axis legend displayed on the right side of the plot, and are calculated using a one-tailed Fisher's Exact test comparing the number of control **Location**s passing and missing the cutoff to those of the test **Location Set**. Note that no correction is made for multiple testing. Also, please note that if the sum of the number of **Location**s in the two **Location Set**s being reported exceed 10,000,000 no p-value is reported (for computational reasons).

Walkthrough

Select one or more **Motif**s for analysis in the first tab and one or more **Location Set**s for analysis in the second tab. If you would like to include the ROC plot, designate one of your **Location Set**s as the "query" set (this only matters if you have multiple **Location Set**s selected). Next, decide which of the offered **Motif** scoring approaches you will use (see top of the first tab, and refer to the section of this manual titled *'Scoring and Identifying Motifs'*).

Next, open a plot tab by clicking the 🟢 button. A new tab will appear, and as your scores are calculated a sub-tab will appear for each **Motif** containing your plot(s). To cancel the process, simply close the tab.

Tips/Tricks/Gotchas

Keep in mind that **Motif** scoring does not account for variation in length among **Location**s within or between **Location Set**s. When possible, it is typically better to compare **Location**s of uniform size. For example, for ChIP analyses you can pair the uniform-length **Location**s obtained from the *'Utilities→Location/Data/Tiled Set→Extract peaks from Data Set(s)'* utility with randomly sampled **Location**s from the '*Sample fixed length Locations from Location Set*' utility.

Utilities→Motif→Distribution→Relative to Locations

Usage

While the *'Motif→Enrichment Table'* utility provides details of the frequency of **Motif** occurrences within **Location Set**s, it provides no information about their distribution. This utility fills this gap, allowing you to test whether **Motif** occurrences within the **Location**s of a **Location Set** are non-uniformly distributed relative to either the end or center of the **Location**s. For example, this utility allows one to ask whether **Motif**s are non-uniformly distributed relative to the start codon or whether a **Motif** is centered at the area of peak enrichment in ChIP-chip/seq binding regions.

Walkthrough

Every option in this menu contains a tooltip that you can consult for specific details. Here, the focus is on a basic overview of how to configure and run the utility.

[1] Configure your search

There are a large number of options (documented by tooltips, and see below), but you will always want to configure the options in the first tab ('*Settings*'):

1. Select the **Location Set** to search.
2. Select one or more **Motif**s to scan against the **Location**s in the **Location Set**.
3. Decide whether you want to analyze positional distribution relative to the starts, ends, or midpoints of the **Location**s. Typically, if you are analyzing promoters you will choose "start" and if you are analyzing binding regions you will choose "midpoint".

The second tab ('*Additional Settings*') provides several additional options for fine-tuning your analysis and adjusting the colors/dimensions of the plots. Some of these are described in further detail below.

[2] Run your search

Once your scan is configured, press the GO button to initiate the scan. You will automatically be taken to the '*Results Display*' tab. Within this tab is a table that will contain a plot for each **Motif**. As the scan progresses the plots are updated (see the bottom information ribbon of the utility for details on the scan progress). Once the plot is completed p-values are calculated and added to the plot (see below). You can stop a search at any time by pressing the '*Cancel Search*' button. Hover your mouse over the tab for an explanation of the features of the plots.

How does it work?

Here is a (basic) overview of how the **Motif** position distribution algorithm actually works:

1. The **Location Set**'s **Location**s are fetched (and refined if the user has chosen to constrain the lengths).
2. The **Location** sequences are fetched relative to their orientation (e.g. such that promoter sequences all end just before the start codon).
3. Each sequence is scanned for **Motif**s. If you have chosen to scan both strands, only the best **Motif** score on either strand is kept for each scan window.
4. For each position along the position distribution (x-axis of the plot) and each possible **Motif** score from 0% (adjustable in settings) to 100% of the maximum score (y-axis of the plot) the percentage of **Location**s that yielded that score or below is expressed on the plot using a color scale (see the legend at the left edge of the plot). If a smoothing span has been chosen (and it is larger than the motif size), the maximum of the **Motif** scores with the span (for each **Location**) is used.
5. A line is drawn on the plot reflecting the percentage of sequences contributing to the calculation at each x-axis position. (In a **Location Set** of varied **Location** size there will be positions where only some **Location**s overlap.)
6. P-values are calculated for the **Motif** distributions at 50%, 60%, 70%, 80%, 90%, and 100% of the maximum **Motif** score. The calculation uses a bootstrapped chi square goodness of fit test, as described by Casimiro *et al.*[10], using a minimum of 5 bins, a maximum of 100 bins, and the lower of either: (1) #hits/5, or (2) the number

of bins that give 100 motif windows per bin. The former method is suggested by Casimiro *et al*. and the latter method is used to reduce over-fitting. By default only positions that have 100% sequence coverage are considered, but an option is given to relax the necessary sequence coverage. In this case a conservative approach is taken in which the missing data are simulated using the expected number of hits taken from the existing data (thereby making the binned distributions more uniform).

Tips/Tricks/Gotchas

My plot is so dark I can barely see anything!

If you expect your **Motif** occurrences to be rare, the plot will essentially be using only the very top of the color scale (for the default scale this is basically dark green to dark black). There is a checkbox in the '*Additional Settings*' tab that changes the color scaling to better emphasize this range. Alternatively, you can increase the smoothing span.

The smoothing span is the key tunable parameter

If you set the smoothing span to a value equal to or below the motif size, no smoothing occurs and you get a per-position readout of **Motif** frequency (however, see the section below about plot width). This is typically the best setting if you are looking for non-uniform distributions at a very fixed distance (e.g. 50bp from the end) or if your **Motif** is very abundant. Otherwise, it is helpful to apply smoothing to spot broader trends and better visualize occurrences of rare **Motif**s. Large smoothing spans result in grey areas on the either end of the plot x-axis because only positions that contain the full span region are plotted. (Note that smoothing has no influence on the p-value calculation.)

Why is my p-value changing slightly even though my settings are the same?

Since the p-value calculation is derived from a simulation, it may vary slightly from test to test. Increasing the number of iterations will diminish this variation (at the expense of a slower calculation).

When the plot width has fewer pixels than the span being measured, compromises are necessary

In such cases smoothing is performed on a per-pixel basis (i.e. smoothing over all distribution values represented by the x-axis pixels). If each pixel represents a small number of distributions, this can create the visual appearance of a periodic non-uniform distribution (e.g. most pixels represent one position but every third pixel represents the smoothed value of two positions). This visual artifact does not influence the p-value calculation.

Interpreting the p-values

There are several things to keep in mind when interpreting p-value significance:

1. No multiple-testing correction is applied.
2. **Motif** distribution can be significantly non-uniform for uninteresting reasons. For example:
   a. Systematic sequence bias in the **Location**s being analyzed, such as sequence composition close to start codons or promoter sequence that extends into neighboring genes.
   b. Repeat sequences leading to **Motif** clusters, such as a **Motif** of predominantly 'A's within a large stretch of 'A's.
3. Only positions with 100% sequence coverage are considered in the calculation unless you relax this constraint in the settings.
4. P-values are displayed as -log$_{10}$-transformed values.

Utilities→Motif→Distribution→Relative to matches to another Motif

Usage

This utility resembles the *'Utilities →Motif →Distribution →Relative to Locations'* utility in most respects, but instead of analyzing position distribution of one or more **Motif**s relative to **Location**s in a **Location Set**, the utility compares their distribution relative to occurrences of a primary **Motif** (i.e. the results are always displayed as a middle-centered distribution, where the middle is the center of each primary **Motif** occurrence).

Walkthrough

Every option in this menu contains a tooltip that you can consult for specific details. Here, the focus is on the basic configuration and the ways in which this utility differs from the *'Relative to Locations'* version.

1. Select a **Location Set** (or full **Sequence Set**). Your primary **Motif** will be scanned against the union of all **Location**s within the set.
2. Select the primary **Motif** and score cutoff for **Motif** occurrences.
3. Select the flank size that will be used to extend the **Motif** occurrences. Each **Motif** occurrence – extended by the flank size – serves as a **Location** that is analyzed in the same manner described for the *'Relative to Locations'* version**.**
4. Select one or more **Motif**s to scan against the primary **Motif**-occurrence **Location**s.

Unlike the *'Relative to Locations'* version, this utility defaults to a smoothing span of 1bp (i.e. no smoothing).

Tips/Tricks/Gotchas

The '*Exclude region of motif overlap from p-value calculation*' checkbox

Because each "**Location**" scanned by the utility is centered at a **Motif** occurrence, the sequence compositions are guaranteed to be non-uniform. You will typically see either a zone of poor/strong hits in the positions that overlap with the primary **Motif** occurrence (i.e. there is an inherent bias towards non-uniformity in this region). By default, this region is ignored (even where the primary and query **Motif** windows only overlap by a single base) and the hit counts are not reported. The region is identified on the plot by a light gray box in the x-axis legend.

Utilities→Motif→Create Location/Data Set from Motif matches

Usage

This utility allows the user to convert a **Motif** to a **Data Set** (and **Location Set**) containing all locations that have a **Motif** score exceeding a user-defined cutoff. The stored values correspond to the **Motif** score for the region. (You will be told the number of locations that pass the selected cutoff and given the option to cancel import before the database entry is created.) Although this approach does not store the PSFM/PSAM information of the **Motif**, one major benefit is that the resulting **Data Set** can be used in the *Data Browser* to quickly browse high-scoring **Motif**s.

Alignment Constraint (optional)

If a genome alignment has been imported into MochiView, the cutoff for inclusion of a **Motif** match in the **Data Set** can be expanded to require that aligned genomes also contain a strong score at the same **Location**. Specifically, the alignment constraint can require that a user-defined number of the selected aligned genomes have **Motif** scores exceeding a user-defined percentage of the maximum possible score for that **Motif**.

Utilities→Motif→Create Data Set from Location Set Motif scores

Usage

This utility creates a new **Data Set** in which each **Location** is assigned a value that corresponds to either the score for the selected **Motif** within the **Location**. Multiple scoring options are provided; see the section of the manual titled *'Scoring and Identifying Motifs'* for details. Note that if you are scanning a PSFM **Motif** with a frequency matrix that has values of zero (e.g. a column with `1.00` for 'A' and `0.00` for all other bases), it is possible that some **Location**s will have no valid LOD score (-infinity). In these cases the **Location**s are given an arbitrary low score (-`10,000`).

Alignment Constraint (optional: max score setting only)

For each scanned sequence window of the reference genome, the selected aligned genomes are also tested for the **Motif** score (if no aligned sequence is available, or if there is an insert/gap, the score is negative infinity). The 'Nth' highest score is chosen as the aligned genome score. The final score for the window is then the lower of the reference genome score and the aligned sequences' score. The highest such score for any window is then assigned to the **Location** as the data value.

The option is also provided to scan on a per-**Location** basis rather than a per-window basis. In this scenario the highest score for each aligned genome within the **Location** of interest (including any inserts) is calculated, and the aligned sequences' score is chosen among these scores (as described for the per-window calculation).

Utilities→Motif→Create Motif from IUPAC

Usage

This utility allows you to create a **Motif** even when you lack the frequency matrix. There are two primary uses for this feature:

1. Create a **Motif** from a consensus sequence (e.g. '`AGSSCT`')
2. Facilitate a degenerate search for a string by making a faux **Motif**

Just type in the *IUPAC* sequence in the section titled '*Enter IUPAC Text Here*' (mouse over the header for a tooltip reminding you of the *IUPAC* code), enter the **Motif** annotation, adjust your certainty level, and press the ✓ button. The certainty level indicates the degree of degeneracy you would like to give your **Motif** (i.e. the lower the certainty, the higher the frequency of the bases that aren't in your submitted sequence). Additional features are provided to give finer control over the uncertainty, and are explained in their respective tooltips. At any time you can check the motif logo tooltip for the current frequency matrix.

Utilities→Location Set→Build promoter set

This utility allows creation of a **Location Set** of the **Location Type** '*Promoter Region*'. The **Location**s in this set are annotated with the name of the gene used to derive the promoter region.

Select Gene Set

The user selects an existing **Location Set** of the **Location Type** '*Gene*'. This set will be used to extract promoters.

Enter Annotation

Provide a name and description for the new **Location Set**.

Promoter Constraints

Enter a '*minimum length*' and '*maximum length*' for the promoter, and declare whether promoters should be truncated at the point where they overlap the nearest gene boundary. These settings are used to determine the length/validity of the promoter for each gene as follows:

- The region upstream of the gene start codon of length equal to the '*maximum length*' is extracted.
- If the gene is on the edge of a **Sequence** and the full length cannot be fetched, the region up to the edge of the **Sequence** is extracted. If the length of this truncated region does not exceed the '*minimum length*', no promoter is extracted for that gene.
- If the '*Stop at gene boundary*' setting is checked and another gene overlaps with the promoter region, the extracted promoter is truncated at the boundary of the neighboring gene. If this truncated region is less than the '*minimum length*', the promoter is extended to the minimum length.

Restrict to Specific Genes (optional)

This tab contains a space for entering a return-delimited list of gene names. The resulting promoter set will be restricted to this set of genes. This option can be used in conjunction with the '*Motif Analysis*' utility to determine whether a set of promoters are enriched for a **Motif**.

Utilities→Location Set→Merge Location Sets→Union

This utility creates a new **Location Set** by merging one or more **Location Set**s. Thus, in the new **Location Set**, overlapping **Location**s will be combined into a single **Location**. The option is provided to merge any perfectly adjacent **Location**s when creating the final **Location Set**.

Single Location Set

If a single **Location Set** is chosen, a '*self-union*' is performed to create the final **Location Set** (i.e. all overlapping **Location**s are combined).

Multiple Location Sets

The union of all **Location Set**s is taken (including '*self-unions*').

Modes of operation

1. Consider Locations on opposite strands separately

This option preserves directionality in the new **Location Set** by separately taking the union of all **Location**s on the plus strands of the selected **Location Set**s and then taking the union of all **Location**s on the minus strands of the selected **Location Set**s. (Note that this means that overlapping **Location**s are not merged if they are on different strands.)

2. Treat all Locations as if they are on the plus strand

Directionality is lost, and overlapping **Location**s on opposite strands ARE merged into a single **Location**.

3. Only keep Locations intersected by all contributing Location Sets ('Union' only)

After the merged **Location Set** is created, only those **Location**s in the merged set that overlap with the intersection of all contributing **Location Set**s are retained. (If only a single **Location Set** is used the intersection is a *self-intersection*).

Utilities→Location Set→Merge Location Sets→Intersection

This utility is similar to *'Utilities→Merge Location Sets→Union'*, except that the resulting **Location Set** only contains **Location**s that are common to all selected **Location Set**s.

Single Location Set

If a single **Location Set** is chosen, a '*self-intersection*' is performed to create the final **Location Set** (i.e. only regions of overlapping **Location**s in the **Location Set** are retained).

Multiple Location Sets

Each **Location Set** undergoes a '*self-union*', and then any region that all **Location Set**s share in common becomes part of the new **Location Set**.

Utilities→Location Set→Merge Location Sets→Subtraction

This utility differs from the other two merge utilities in that a primary **Location Set** is selected, and the new **Location Set** is created by subtracting the **Location**s within one or more additional **Location Set**s. For example, if one of the additional **Location Set**s contained a **Location** that was entirely internal to a **Location** in the primary **Location Set**, subtracting that **Location** would yield two **Location**s (those on either side of the internal **Location**).

Primary Location Set only

If a single **Location Set** is chosen, a '*self-subtraction*' is performed to create the final **Location Set** (i.e. all regions of overlapping **Location**s in the **Location Set** are removed).

Primary Location Set plus additional Location Set(s)

Each **Location Set** undergoes a '*self-union*', and then any region that is only found in the primary **Location Set** becomes part of the new **Location Set**.

Utilities→Location Set→Sample fixed length Locations from Location Set

This utility creates a new **Location Set** from a source **Location Set** by sampling a randomly positioned region of a user-defined width within each **Location** in the source. If the source **Location** is smaller than the user-defined sample width, the source **Location** is ignored, unless the "also sample from Locations smaller than sample width" option is selected. In these cases the sampled **Location** is centered in the middle of the source **Location** and extended beyond the boundaries of the source **Location** to obtain the desired sample width.

Usage

The original purpose of this utility was to create sampled regions of a fixed width from intergenic **Location Set** for use as controls when using *'Utilities→Motif→Enrichment Plot'* to determine motif enrichment of peaks of equivalent length obtained from *'Utilities→Location/Data/Tiled Set→Extract peaks from Data Set(s)'*.

Utilities→Location Set→Create Location Set from sequence matches

This utility creates a new **Location Set** from matches to a sequence search. The region searched can be constrained to lie within a **Location Set**, a **Sequence**, or a full **Sequence Set**.

Usage

This utility is primarily designed for the identification of motifs involving repeats and motifs described by *IUPAC* symbols (e.g. `AT[GC]G[CA]T`). Matches to searches are converted to a **Location Set**, which can then be utilized in conjunction with other utilities to characterize the match distribution.

A large portion of the interface to this utility is described in detail in the section covering the *Sequence Browser*. This utility differs from the browser in that a search can be constrained to lie within a **Location Set**. When this option is chosen, the user has the choice of either scanning the *plus* and *minus* strands of each **Location** or scanning only in the direction of the **Location**.

Utilities→Location Set→Annotate with gene proximity

This utility is identical to the *'Export→Location Set→Gene proximity assignments'* utility, with the exception that *Location Annotations* are added without any option for exporting the assignments to a tab-delimited file.

The options for proximity assignment are described in the manual section titled *'Common Menu Elements'* under the header *'Gene Proximity Assignment' panels*. Note that any existing *Location Annotations* will be overwritten!

Utilities→Location Set→Summary/comparison

This utility provides detailed information about the distribution of **Location**s with **Location Set**s (broken down by strand and **Sequence**). In addition, the utility provides information about the extent of overlap between two **Location Set**s.

Usage

On the main screen you may select either one or two **Location Set**s and then click the GO button to begin the analysis (this will take a little while for large **Location Set**s). At any time you can click the ⊗ button to close the utility. When the analysis is complete, you may click the 💾 button to save the data as a tab-delimited text file, or you may browse the data in the two summary tabs. The analysis is broken down by category, with each category providing its own table of data broken down by strand (plus/minus/both). The first row always contains data describing the full **Sequence Set**, while the remaining rows are divided among the **Sequence**s.

Location Set Summary Tables

The two **Location Set** summary tables include several sub-tables, which vary in the meaning of the '*Value (Both)*' column, as described below. In some cases percentages are also provided, their meaning is also clarified in the summary below.

Coverage (Locations)

This table shows the number of **Location**s in a given region and on a given strand.
'Value (Both)': The sum of the '*Value (Plus)*' and '*Value (Minus)*' columns (i.e. all **Location**s in that region).
PERCENTAGES: The percentage of **Location**s relative to the total number in the **Location Set.**

Coverage (bp)

This table shows the number of base pairs covered by **Location**s in the **Location Set** in a given region and on a given strand. Note that overlapping **Location**s are merged before making this calculation.
'Value (Both)': All **Locations** are treated as if they are on the plus strand.
PERCENTAGES: The percentage of base pairs relative to the '*Value (Both)*' count for the full **Sequence Set**.

Overlap (Locations)

This table shows the number of **Location**s that overlap another **Location** in the **Location Set**.
'Value (Both)': Shows the number of **Location**s that overlap with a **Location** on the opposite strand.

Overlap (bp)

This table shows the number of bases in a region that are covered by two or more **Location**s on the indicated strand.
'Value (Both)': Shows the overlap between the *plus* strand **Location**s and the *minus* strand **Location**s.

Length (Avg., Med., Min., Max.)

These four tables give the average, median, minimum, and maximum **Location** lengths for the specified region and strand.
'Value (Both)': Considers **Location**s on both strands.

Location Set Comparison Table

This table also includes several sub-tables providing descriptions of **Location** overlap between two **Location Set**s. In this table, the final strand column is called '*Value (All as Plus)*' and provides data for the comparison when all **Location**s for both **Location Set**s are treated as if they were on the same strand.

Overlap (bp)

This table indicates the number of bases in a region that are covered by **Location**s from both **Location Set**s (on the indicated strand).

LS1: Overlap (Locations)

This table indicates the number of **Location**s in **Location Set** #1 that overlap with a **Location** in **Location Set** #2 on the indicated strand and region.

LS2: Overlap (Locations)

This table indicates the number of **Location**s in **Location Set** #2 that overlap with a **Location** in **Location Set** #1 on the indicated strand and region.

LS1: Full Overlap (Locations)

This table indicates the number of **Location**s in **Location Set** #1 that are *completely* overlapped by **Location**s in **Location Set** #2.

LS2: Full Overlap (Locations)

This table indicates the number of **Location**s in **Location Set** #2 that are *completely* overlapped by **Location**s in **Location Set** #1.

Utilities→Location/Data/Tiled Set→Map Data Set(s) to Location Set

This utility creates a new **Data Set** assigned to a **Location Set** using data from existing **Data Set**(s).

Usage

First choose the **Location Set** and *source* **Data Set**(s) that will be used to create the *new* **Data Set**. The *source* **Data Set**(s) need not belong to the **Location Set** (but the *new* **Data Set** will). For each **Location** in the chosen **Location Set**, all values from overlapping **Location**s in each **Data Set** will be gathered and combined to determine the value assigned to the **Location Set**'s **Location**. The criteria used to determine whether **Location**s overlap and how the values should be combined are selected in the utility.

Location overlap criterion

When determining the score for each **Location** of the **Location Set** this criterion determines which **Location**s of the *source* **Data Set** will be considered. The available options are '*Any Overlap*', '*Fully Inside*', or '*Midpoint Inside*'. For example, '*Midpoint Inside*' imposes the requirement that, for each **Location** being scored, only **Location**s in the *source* **Data Set** that have a midpoint inside of the **Location** being scored are considered.

Value selection criterion

Once the **Location**s passing the overlap criterion are determined, the score is determined based on the value selection criterion of '*Max*', '*Min*', '*Mean*', '*Median*', or '*Quantile*' (quantile increments of 10 are offered).

Minimum # of Data Sets with overlapping values for inclusion

This setting determines the number of **Data Set**s that must contribute a value in order for the **Location** to be assigned a value.

Apply 'value selection criterion' on a per-Data Set basis

(This setting is irrelevant if only a single **Data Set** is selected or the *value selection criterion* is '*Max*' or '*Min*'.) When checked, the *value selection criterion* is applied on a per-**Data Set** basis before being applied across **Data Set**s. For example, if a **Location** has three values mapped to it from each of two **Data Set**s and the '*Median*' value is requested, the median of the three values for each **Data Set** is calculated, and then the median of these two values is returned. If this option is not selected, the median of the six values is returned.

Example

For example, this utility could be used to create a **Data Set** from a **Location Set** of ChIP-chip binding sites in which each binding site is assigned a value equal to the highest enrichment score from all array probes fully within that region. In this example, the (single in this case) *source* **Data Set** would be the enrichment data (belonging to a **Location Set** of array probes), the location overlap criterion would be '*Fully Inside*', and the value selection criterion would be '*Max*'. For each binding site, MochiView would identify all array probes that fall fully within the site and then determine the maximum value associated with these values. This value would then be assigned to the binding site and become part of the new **Data Set**. If no array probes with valid values meet the overlap criteria for a given binding site, no value is assigned.

Utilities→Location/Data/Tiled Set→Map Tiled Set(s) to Location Set

This utility is essentially the same as the *'Utilities →Location/Data/Tiled Set →Map Data Set(s) to Location Set'* utility, but **Tiled Set**s are mapped instead of **Data Set**s.

Usage

Tiled Set(s) are mapped on to **Location**(s) in the selected **Location Set** on a strand-specific basis. In other words, you choose the **Tiled Set**(s) to be mapped onto plus-strand and minus-strand **Location**s separately. If strand-specificity is not desired, just choose the same **Tiled Set**(s) in the plus- and minus-strand selection tables. Additional options are as described for the *'Utilities →Location/Data/Tiled Set →Map Data Set(s) to Location Set'* utility, except that there is no option for overlap criterion. This is because the **Tiled Set** values are considered on a per-base basis (see below), so overlap is not relevant.

How are the data mapped on to a Location?

- First, if the **Tiled Set**s have span greater than 1bp, they are broken down into 1bp values (i.e. a 3bp tile spanning coordinates `25-27` with a value of `5.3` becomes three 1bp tiles with value `5.3`). Reduction to 1bp resolution is important because it allows only the portion of a tile that overlaps the **Location** to be considered, and if the mean or median is being used the result is that a slight overlap has less weight.
- Next, all **Tiled Set** values that overlap the **Location** (and are assigned to the same strand) are collected. If the option to consider missing values as zeroes was selected, all missing data are filled in with zeroes.
- If the option to apply the data criterion on a per-**Tiled Set** basis was selected, the values from each **Tiled Set** are collapsed to a single value according to the criterion (e.g. median value) before being further collapsed among each other to the final value (using the same criterion). Otherwise, the values are collapsed between **Tiled Set**s on a per-base basis and then the final set of values are collapsed.
- Understanding how missing values are handled is important (unless you have selected the option to consider missing values as zeroes), and varies depending on whether you are first collapsing on a per-**Tiled Set** or a per-base basis. However, in both cases the basic rule is that the missing value isn't even considered. In other words, if only a single base has a value of `10` and all other values are missing, the median, mean, minimum, and maximum values are all `10`. If operating on a per-**Tiled Set** basis, this means that a **Tiled Set** with only a single value (and the rest missing) has equal weight to the others during the final round of collapsing. If operating on a per-base basis, that **Tiled Set** would only contribute to the first round data collapse at that single base.

Example of data collapse

Here, I provided an example of a tiny **Location** of coordinates `1-5` and data from three **Tiled Set**s ('`x`' is missing data). The bottom row shows the median value on a per-base basis, the far-right column shows the median value on a per-**Tiled Set** basis. The bottom right shows how the results from the final median collapse would differ depending on the approach taken.

Coordinate:	**1**	**2**	**3**	**4**	**5**	**Per-Tiled Set collapse**
Tiled Set #1	x	x	x	60	x	60
Tiled Set #2	10	10	10	10	x	10
Tiled Set #3	30	30	40	40	x	35
Per-base collapse	20	20	25	40	x	*FINAL COLLAPSE* Per-base: 22.5 Per-Tiled Set: 35

If missing values were replaced with zeros, the result would change as follows:

Coordinate:	**1**	**2**	**3**	**4**	**5**	**Per-Tiled Set collapse**
Tiled Set #1	0	0	0	60	0	0
Tiled Set #2	10	10	10	10	0	10
Tiled Set #3	30	30	40	40	0	30
Per-base collapse	10	10	10	40	0	*FINAL COLLAPSE* Per-base: 10 Per-Tiled Set: 10

Potential uses

This utility allows the mapping of **Tiled Set** data such as ChIP-Seq or RNA-Seq read counts to genomic features such as genes or intergenic regions so that the regions can be browsed in the *Data Browser*.

Utilities→Location/Data/Tiled Set→Create Data Set by extracting enriched regions from Tiled Set

Acknowledgment

This utility is based on an approach to extracting transcribed regions from RNA-Seq data devised by Brian Tuch and described in this document: http://solidsoftwaretools.com/gf/download/docmanfileversion/138/693/NTR_Finder_Manual_v1.1.pdf.

Overview

This utility will create a **Data Set** from one or more **Tiled Set**(s) by identifying **Location**s in which the **Tiled Set** values exceed a user-defined threshold. The value associated with the **Location** is the average of the **Tiled Set** values within the **Location**. You can extract both plus-strand and/or minus-strand **Location**s, by assigning **Tiled Set**(s) to the two strands in the tables at the bottom of the menu (e.g. plus-strand and minus-strand RNA-Seq **Tiled Set**s).

What is this good for?

The approach was originally devised for extracting transcribed regions from plus- and minus-strand RNA-Seq data (see acknowledgment above). However, the approach is also useful for identifying regions of enrichment in almost any **Tiled Set** for use in the data browser.

How does it work?

Enriched regions are found by scanning a window of user-defined size across each **Sequence**. The **Tiled Set** values at each base within the window are calculated, and the average value is determined (missing data are considered a value of zero). If multiple **Tiled Set**(s) are being used, their values are collapsed on a per-base basis using the user-defined approach (i.e. quantile, median, mean, minimum, or maximum). If the average value in the window exceeds the user-defined threshold, the **Location** defined by the window becomes a candidate **Location**, which is then extended by further scanning the window until the average window value falls below the threshold.

The set of candidate **Location**s identified using this window-scanning approach can be further refined in the following ways (and in the order presented):

1. Trim ends using trim threshold

The **Location** is trimmed on either side on a base-by-base basis until the value for that base exceeds a trim threshold calculated using the user-defined trimming function. The trim threshold is calculated by multiplying the average value for the **Location** by the user-defined trim multiplier (e.g. 0.01).

2. Combine Locations that are less than X bp apart

If the "*maximum gap size for Location merge*" is set to greater than 0bp, all candidate **Location**s that are less than this distance apart are fused into a single **Location**.

3. Apply a minimum Location size filter

If the candidate **Location** is smaller than the user-defined minimum **Location** size, it is removed. This is a useful way to eliminate candidates that arose from a single spike in the data and then were trimmed down to only a few bases.

4. Extend Locations using a scaffold

It may have other uses, but the original intent of this option was to take transcribed regions identified from RNA-Seq data and force those that overlapped with known genes to extend the length of the gene (if multiple overlap the gene, they are fused). Note that this is applied on a strand-specific basis. An option is also provided to split extracted **Location**s that span multiple scaffold **Location**s, as might occur if two genes are very close together such that there is no clear break in the RNA-Seq data.

Usage

The utility is heavily supported by tooltips, and hopefully between the description above and the tooltips everything will be clear. If not, send me an e-mail ☺.

Utilities→Location/Data/Tiled Set→Refine Location/Data Set

This utility allows the user to create a new **Location Set** (and, optionally, a new **Data Set** as well) with a reduced number **Location**s. The **Location**s are culled based on the filters chosen in the filter tab. If a **Data Set** is chosen, the **Location**s are further culled to include only those **Location**s that have a data value in the **Data Set**. An additional option is provided to reduce the size of any **Location**s that pass these criteria.

Usage

The usage (in terms of filtering and refining) is similar to that described for the *'Export→Location Set→Format: FASTA'* utility. However, in this case rather than exporting to a FASTA file a new **Location Set** is made with the **Location**s. If you choose an optional **Data Set** in the menu, a new **Data Set** is made as well, carrying over the values from the old **Location Set** (even if the new **Location**s have been truncated using the '*Enable Location Size Refinement*' option).

Utilities→Location/Data/Tiled Set→Combine Tiled Sets

Usage

This utility creates a new 1bp span **Tiled Set** from two or more **Tiled Set**(s) by scanning values on a base-by-base basis and taking either the quantile, median, mean, minimum, or maximum value. If a **Tiled Set** has a tiling span greater than 1bp, it is decomposed to single-base values by assigning the value of the span to every base within the span. Missing values are treated as zero by default, but the option is also provided to omit them. If missing values are omitted, the option is given to set a minimum number of **Tiled Set**s that must have a value at a given base for a value to be assigned to that base.

The panel titled '*Tiled Set storage settings*' is described in the section in the manual titled *'Common Menu Elements'*.

Utilities→Location/Data/Tiled Set→ Create Tiled Set groups (addition/subtraction)

Usage

This utility allows the selection of two groups of **Tiled Set**(s), which are first combined to a single value per group (at each base), and then further merged through either addition (*group#1* value + *group#2* value) or subtraction (*group#1* value - *group#2* value) to make a new **Tiled Set**. The first step (intra-group merger) is performed as described for the *'Utilities →Location/Data/Tiled Set →Combine Tiled Sets'* utility. The second step is performed only if both groups yielded a valid number (i.e. bases with a missing number for either group are skipped).

The panel titled '*Tiled Set storage settings*' is described in the section in the manual titled *'Common Menu Elements'*.

Utilities→Location/Data/Tiled Set→Extract peaks from Data Set(s)

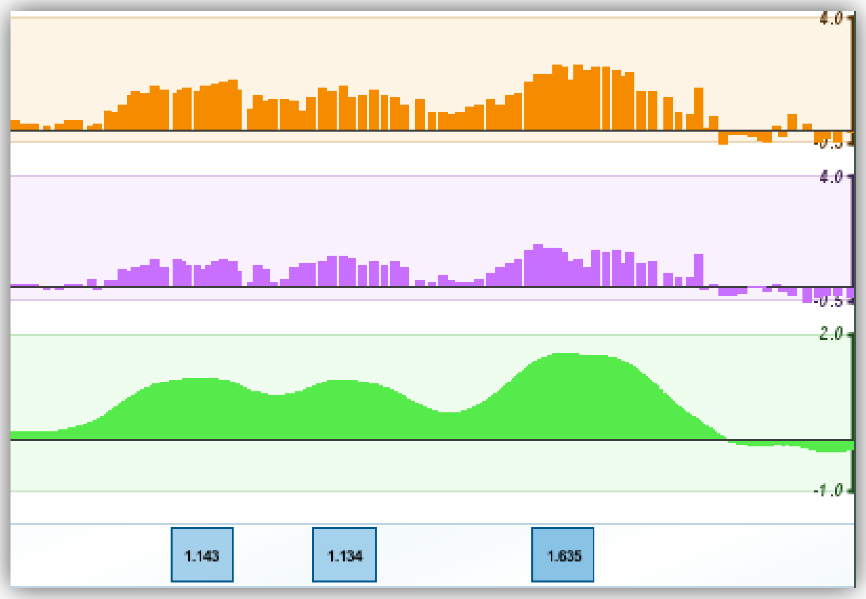
This utility is only appropriate for **Data Set**s with fairly high data density (it was designed for use with $\log_2$-transformed ChIP-Chip tiling array data). The utility smoothes the data from the **Data Set** (using the same approach as the *'Utilities →Location/Data/Tiled Set →Create smoothed Tiled Set from Data Set'* utility) and then applies a multi-pass filter to identify regions of a user-defined width and user-defined minimum peak value. This is especially useful to refine peak calls prior to searching for **Motif**s. While in principle any type of data can be used, the utility was designed with $\log_2$-transformed tiling enrichment data in mind.

Background

There are many ChIP-Chip peak-finding algorithms available, including approaches that deconvolve individual binding events using peak profiles predicted from the ChIP shearing distribution (e.g. MeDiChI[11] and Joint Binding Deconvolution[12]). These approaches are much more sophisticated than the peak-finding algorithm employed by MochiView, but they are also based upon underlying assumptions such as uniformity of shearing that can prove problematic. Moreover, these approaches can be easily confounded by noise in the data (e.g. individual probes with very high expression values being called as peaks).

MochiView's peak extraction algorithm is provided as a convenient and tunable complement to deconvolution-based approaches, and provides the means to rapidly narrow the search space when attempting to identify novel **Motif**s. The algorithm can analyze multiple replicates at once (including control samples), and a randomized sampling technique is used to test the significance of each peak region.

Understanding Smoothing



TRACKs #1 and #2: ChIP-Chip **Data Set**s (each bar is a $\log_2$ expression value for an individual probe)
TRACK#3: Smoothed **Tiled Set** created from the **Data Set**s (containing a value for each individual base)
TRACK#4: Three 500bp peak regions extracted from the **Tiled Set**

The peak extraction algorithm searches smoothed **Data Set**s for peaks. When deciding on your smoothing settings for the peak extraction utility, it is recommended that you first visualize the smoothed data using the *'Utilities →Location/Data/Tiled Set →Create smoothed Tiled Set from Data Set(s)'* utility (see the manual entry of that utility for a description of the smoothing algorithm). The smoothing options for this utility are identical to those for the *'Utilities →Location/Data/Tiled Set →Extract peaks from Data Set(s)'* utility.

Try to find a balance between not over-smoothing (which can merge closely adjacent peaks) and under-smoothing (which can lead to a jagged **Tiled Set** and confound the peak extraction algorithm). For our laboratory's ChIP-Chip tiling arrays, which typically have ~50-100bp between probe midpoints, a 500bp smoothing flank size works well.

Stage#1: Extraction of Candidate Peaks

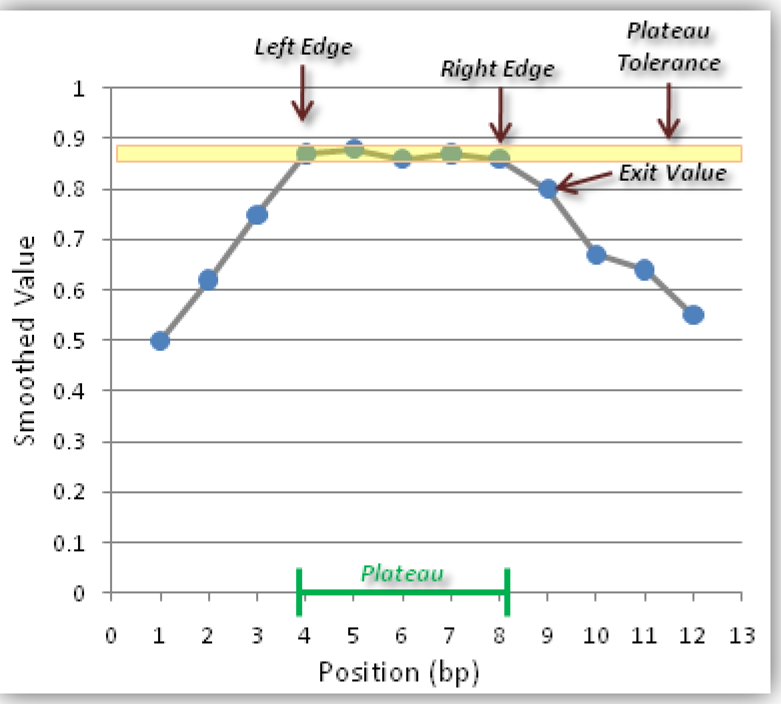
Basic *candidate peak* identification

After the smoothed **Tiled Set** has been created from the **Data Set(s)**, the peak extraction algorithm is used to generate a set of *candidate peak* windows. For each **Sequence** in the **Sequence Set**, the sequence is traversed from the first base to the last searching for bases that:

A. Have a smoothed value that exceeds a user-defined minimum cutoff
B. Have a smoothed value that exceeds the smoothed values of the flanking bases

All bases that meet these criteria are considered the midpoints of *candidate peaks*, and are further analyzed in the stage#2. To be more precise, peak-calling criterion 'B' also requires that difference between the base value and the right flanking value must exceed a "Plateau Tolerance" value. This concept is further explained in the next section.

Handling of plateau-shaped peaks



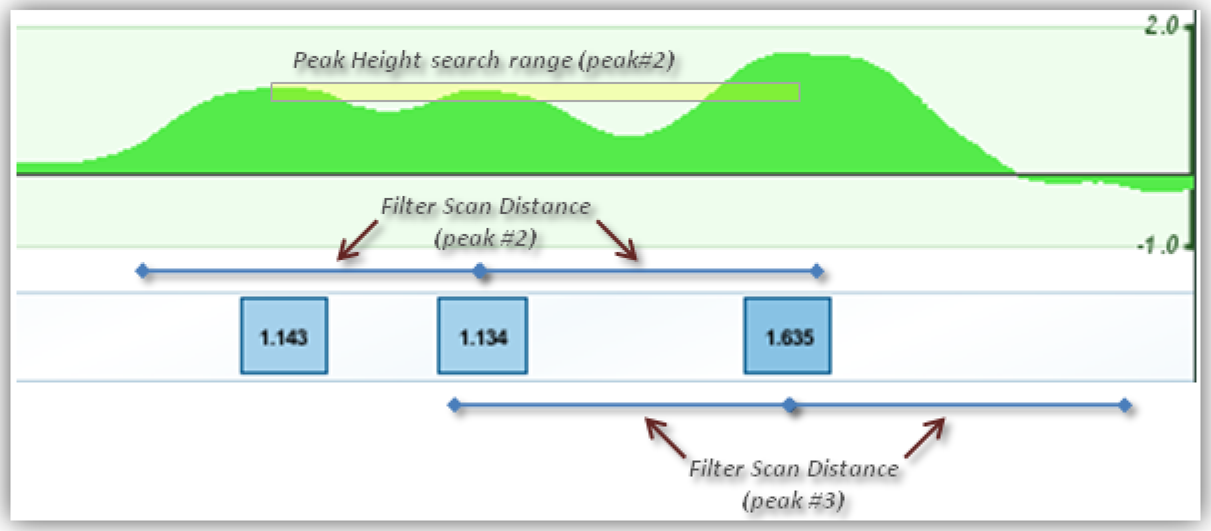
The Stage#1 algorithm also identifies peaks shaped like a plateau. A simple sample case is diagrammed in the image above. The algorithm recognizes entry into a potential plateau (i.e. the *left edge*) as any case in which the smoothed value is greater than that of the left flanking base value but within the range of a "*plateau tolerance*" from the right flanking base value. When such an entry point is identified, additional bases are scanned (left-to-right) until the value deviates from the *left edge* value by more than the *plateau tolerance*. If this *exit value* exceeds the *left edge* value, the plateau is ignored (because it is stepping upward). Otherwise, the midpoint of the plateau region is considered the midpoint of a *candidate peak*.

Handling of missing smoothed values

Depending on your smoothing parameters and the **Location** distribution of your **Data Set**(s), you may have some regions of missing values in the smoothed **Tiled Set**. These regions are not scanned, and if they interrupt a plateau the plateau is not considered a *candidate peak*.

Stage#2: Candidate Peak Refinement

Peak height filtering: overview


Peak Height search range (peak#2)
2.0
Filter Scan Distance (peak #2)
-1.0
1.143
1.134
1.635
Filter Scan Distance (peak #3)

Each peak in the list of *candidate peaks* must meet ***one*** of following two criteria:

A. The peak passes the *peak height* filter within the *peak height search range*
B. The peak has the highest smoothed value of all peaks within a user-defined *filter scan distance* (in each direction from the midpoint)

Peak height filtering: criterion 'A'

In order to be considered a "true" peak, the smoothed values must dip a certain amount on either side of the peak apex before they rise again. Otherwise the peak is deemed too shallow or a shoulder on the ascent to a taller and more robust peak. Criterion 'A' tests whether a *candidate peak* meets these requirements.

The search range for assessing peak height for each peak is bounded on each side by either the first *candidate peak* midpoint with larger value or the *filter scan distance*. In the diagram above the search range is shown for peak#2 (yellow bar), and on both flanks it is limited by a neighboring *candidate peak*. In order to pass the *peak height* filter, the smoothed value must decline on either side of the flank (scanning outward from the *candidate peak* midpoint) by the user-defined *minimum peak height change* before it either ascends by the same amount or reaches the end of the search space. In the image above, one can see that the smoothed values dip below this range (i.e. below the yellow bar) on each flank within the search space before they dip above. Thus, the peak would pass the filter.

Peak height filtering: criterion 'B'

The goal of criterion 'B' is to ensure that the tallest peak in the user-defined scan distance is retained regardless of whether it is too broad to meet the *minimum peak height change* requirement of criterion 'A' within the *filter scan distance*. Put more simply, the requirement of a conventional peak "shape" is relaxed for a *candidate peak* if it is the tallest peak in the neighborhood.

In the image above the *filter scan distance* of peak#3 is indicated. One other peak (#2) is within the filter scan distance of peak#3, but this peak's value (1.134) is less than the value of peak#3 (1.635). Thus peak#3 passes criterion 'B'. Such a peak is designated a "primary peak" and is retained (regardless of whether it satisfies criterion 'A').

Filtering overlapping peaks

After the above filters are applied, the new list of *candidate peaks* is searched for overlapping peaks. Overlap is defined as two *candidate peak* midpoints that do not exceed the user-defined "minimum distance between peak midpoints". Overlapping peaks are handled as follows:

A. Remove all overlapping peaks except the peak(s) with the highest value.
B. If any of these remaining peaks still overlap, combine them into a single peak using the average of the peaks' midpoints.

Stage#3: Overlap Filtering (optional)

'Location Set that must overlap peak' filter

Unlike the filtering of overlapping peaks described above, this stage offers the option to remove *candidate peak*s that do not overlap with a user-defined **Location Set**. This **Location Set** will typically be a set of intergenic regions (you can make these with the *'Utilities →Location Set →Merge Location Sets →Subtraction'* utility by subtracting genes from your **Sequence Set**). Three overlap options are supplied:

A. *Midpoint Inside*: The candidate peak midpoint must fall within a **Location** in the **Location Set**.
B. *Any Overlap*: Some portion of the peak **Location** (of user-defined width centered around the midpoint) must overlap with a **Location** in the **Location Set**.
C. *Fully Inside*: The entire peak **Location** (of user-defined width centered around the midpoint) must overlap with **Location**(s) in the **Location Set**.

Please note that confining peaks to intergenic space is a tradeoff between eliminating false positives (ChIP can yield enrichment over heavily transcribed ORFs, presumably due to non-specific interactions) and creating false negatives (the assumption is being made that binding events of interest are confined to intergenic space).

'Location fully contained by peak' filter

This optional filter eliminates any candidate peak that completely overlaps a **Location** in a **Location Set**. This filter was specifically designed to eliminate artifacts in the Johnson Lab ChIP-Chip experiments involving non-specific peaks over tRNAs. When using the 'Location Set that must overlap peak' filter with the "Midpoint Inside" option, we sometimes found that the midpoint fell just outside of the tRNA the peak was not filtered. This 'Location fully contained by peak' filter removed these peaks because the tRNA were completely within the peak **Location**s.

Stage#4: Significance Sampling (optional)

Significance relative to scrambled experimental data

This option asks the following question of each *candidate peak*: How often does a smoothed value derived from randomly sampled data equal or exceed the actual value? The user defines the number of samples tested (the default of `10,000` is fast, but `100,000` or even `1,000,000` is recommended for greater confidence) and the maximum number of random samples that may match or exceed the actual peak value before the candidate peak is discarded.

Note that the smoothed values are derived in exactly the same fashion from the sampled data as they were from the actual peak, thus accounting for **Location** (typically ChIP-Chip probe) density and spacing around the peak. Therefore, a peak with a modest value is more likely to pass if it has high **Location** density, and similarly a peak that is primarily reliant on only a single **Location** is more likely to fail.

Significance relative to control Data Set(s)

Optionally, the user can also choose to select one or more control **Data Set**s (from the same **Location Set** as the experimental **Data Set**s) and require that the *candidate peak* value for these sets does not show significant enrichment in the control sets. The procedure is identical to that described above for scrambling experimental data, except that the sampled data comes from the control **Data Set**s, and the filtering criteria are reversed (i.e. *candidate peak* is retained unless *less than* 'X' random samples exceed the actual control value at the *candidate peak*).

Stage#5: Save Peaks as Location and Data Set

Each peak that passes all of the criteria outlined above is included in the final **Location Set**. The individual peak **Location**s are of a user-defined width centered at the *candidate peak* midpoint (if the **Location** extends off the end of a **Sequence** it is truncated). The **Data Set** contains the value at the smoothed *candidate peak* midpoint. If significance sampling was conducted, the **Location**s are assigned annotation ('ANNO_TAG' and 'ANNO_DESC') providing details on the p-value and the number of samples passing.

Menu Options

The specific menu options in this utility are described in MochiView with tooltips (select user-defined attributes are also described above).

Utilities→Location/Data/Tiled Set→Create smoothed Tiled Set from Data Set(s)

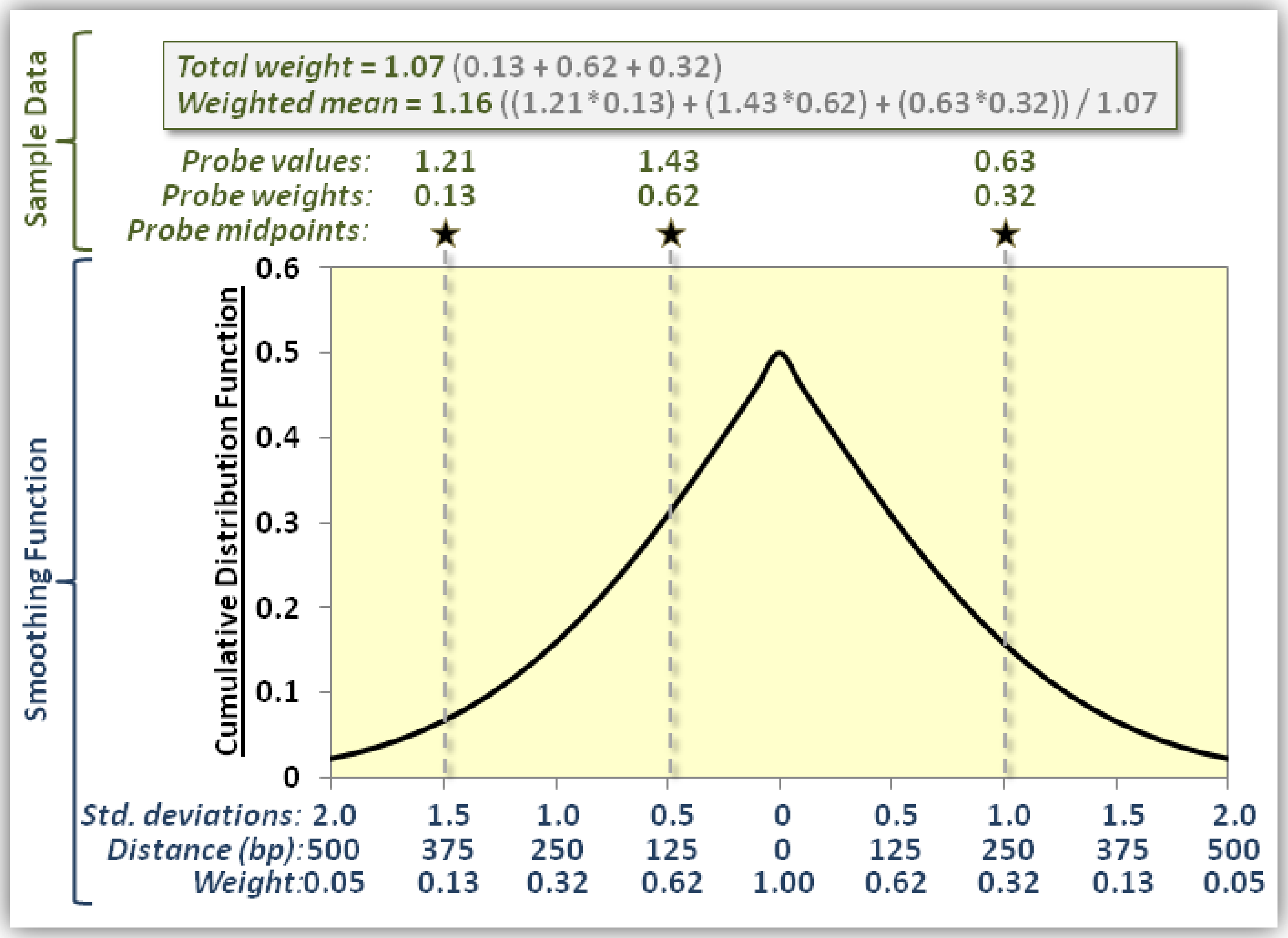
Overview

This utility is only appropriate for **Data Set**s with fairly high data density (it was designed for use with $\log_2$-transformed ChIP-Chip tiling array data). The data are smoothed using a user-defined smoothing window and a weighted average (the weight decreases with distance) and then saved as a **Tiled Set**.

Please note that **Tiled Set**s cannot currently include more than ~700 **Sequence**s. If this limit is problematic for you, contact me and I'll finally get around to removing it ☺.

The panel titled '*Tiled Set storage settings*' is described in the section in the manual titled *'Common Menu Elements'*.

Usage



Smoothing is applied by calculating a smoothed value for every position in the **Sequence Set**, using a smoothing function constructed from the cumulative distribution function of a normal distribution in which the standard deviation is half of the user-defined "smoothing flank size". Follow along on the figure above as I describe the calculation of the smoothed value for a given position in the **Sequence Set**:

1. Construct a smoothing function based on user-defined smoothing flank size. In the example above the entered size is `500`bp (see the row at the bottom labeled "*Distance (bp)*").
2. Identify all **Location**s in the **Data Set** with a midpoint falling within two standard deviations (i.e. the smoothing flank size) of the position being smoothed. In the sample data above three ChIP-Chip probes fall within this region (see the row near the top labeled "row midpoints"). For convenience, these probes are exactly `0.5`, `1.0`, and `1.5` standard deviations from the smoothing position.
3. Calculate the combined weight of the **Location**s using the smoothing function (the weight for a given **Location** is actually twice the value on the y-axis of the graph to account for the flank symmetry). In our example the combined weight (see the gray box near the top) is 1.07.
4. Provided that the combined weight exceeds the user-defined "minimum weighted **Location** count for inclusion", the smoothed value is calculated as the weighted average of the three **Location** values (again, see the gray box). If the weight is not exceeded, no value is assigned and there will be no smoothed value at this position (see interpolation below).

Data Interpolation

As described in step four above, it is possible that some positions in the **Sequence Set** will lack a smoothed value. How these missing values are handled is decided by the user-defined "maximum gap for data interpolation". For each position with a missing value, the value is calculated by linear interpolation of the closest flanking smoothed values, provided that the distance between these closest values does not exceed the user-defined maximum gap. (Setting the maximum gap to zero eliminates all data interpolation.)

Reduce Spikes in Data

Sometimes ChIP-Chip data can contain noisy probes that result in large spikes in the data (even after normalization). These spikes can interfere with smoothing, and thus a crude-but-rather-effective option is provided to dampen these spikes prior to smoothing. If this option is enabled, each **Location** value in the **Data Set** is adjusted as follows:

1. Identify the closest **Location**s upstream and downstream of the **Location**
2. If no **Location** is found on either side, or if the distance between the midpoint of either of these flanking **Location**s and the central **Location** exceeds the user-defined smoothing flank size, do nothing.
3. Otherwise, if the value associated with the central **Location** exceeds the higher of the two flanking values (F_{max}) by greater than the user-defined spike reduction value (S), change the central value to $F_{max} + S$. (The same principle is applied if the value is lower than the two flank values).

Smoothing With Multiple Data Sets

The option is provided to select more than one **Data Set** for smoothing (provided they all belong to the same **Location Set**). In this case smoothing is conducted separately for each selected **Data Set** (as described above), and the smoothed data are then combined by taking the combined value (default approach is to use the median) among all smoothed **Data Set**s at each position.

Utilities→Location/Data/Tiled Set→Create smoothed Tiled Set from Tiled Set(s)

Overview

This utility is as described above for the *'Create smoothed Tiled Set from Data Set(s)'* utility except that (you guessed it) **Tiled Set**(s) are used instead. Make sure that you consider your settings carefully, as many of the defaults are data-density dependent (e.g. '*Minimum weighted count for inclusion*' should probably be raised if the tiling span is 1bp). Also, keep in mind that the **Tiled Set**(s) are broken down to 1bp **Location**s for the purpose of smoothing (thus, a value of 3.4 in a 10bp tile becomes ten 1bp **Location**s, each with a value of 3.4).

The panel titled '*Tiled Set storage settings*' is described in the section in the manual titled *'Common Menu Elements'*.

Utilities→Data Transfer→Transfer data between Sequence Sets

This utility transfers data between two closely related **Sequence Set**s (e.g. an updated genome assembly that contains indels and/or sequence changes) using a multiple alignment to provide mapping information.

Requirements/Restrictions

- The target **Sequence Set** cannot be larger than 20 million base pairs (this utility uses a sequence mapping technique that requires a considerable amount of memory).
- **Location Set**s of **Location Type** '*Gene*' and '*Alignment Block*' cannot be transferred. However, in the case of gene sets, keep in mind that you can transfer **Data Set**s to a new gene set using *'Utilities→Data Transfer→Transfer gene Data Sets between gene Location Sets'*.
- An alignment (i.e. **Location Set** of **Location Type** '*Alignment Block*') containing both the source **Sequence Set** and the target **Sequence Set** must be provided. (Such an alignment can be created using the software MAUVE and imported into MochiView using the *'Import→Location Set (alignment)→Format: Mauve'* utility.)
- Source **Location**s must be at least 10bp in width or they will not be considered.

How does it work?

First, in order to speed sequence lookups in the target **Sequence Set**, a lookup table is created using the technique described by Reneker and Shyu [13]. Then, for each **Location** of at least 10bp width in the selected source **Location Set**:

1. The alignment is used to find the target DNA sequence that is aligned with the DNA sequence of the source **Location**. This includes any gaps/inserts.
2. If the target sequence is less than 10 bases, the **Location** is not mapped over and an entry is made in a log-file that can be saved and viewed by the user upon completion of the transfer.
3. The source **Sequence Set** is searched for the target sequence, and provided that a single unique match is found, the **Location** is mapped to that matching sequence. Note that this means that the genomic context in the alignment outside of the target **Location** is not considered (i.e. this information is not used to disambiguate non-unique matches).

Provided that at least one **Location** in the target **Sequence Set** was successfully mapped to the target **Sequence Set**, a new **Location Set** is created for the target **Sequence Set** containing the mapped **Location**s. Then, for any **Data Set** belonging to the source **Location Set** new **Data Set**s are created for the target **Sequence Set** containing the data mapped from the source **Location**s to the target **Location**s.

Utilities→Data Transfer→Transfer gene Data Sets between gene Location Sets

As gene annotations change over time, one may want to keep the gene set (i.e. **Location Set** of **Location Type** '*Gene*') in MochiView current without having to re-upload any associated **Data Set**s. If the gene boundaries have not changed (and no genes were added and removed), *'Import→Annotation→Format: MochiView (by gene)'* serves this purpose. However, if changes have occurred to the overall gene boundaries (or if you desire to add/remove genes), it is necessary to import a new gene **Location Set**. One can then transfer **Data Set**s associated with the old gene set to the newly updated gene set using this utility. Note that this can also be used to transfer gene data between **Sequence Set**s.

Usage

Once you have selected the source and target gene sets and **Sequence Set**s as well as the **Data Set**s to be transferred, the utility attempts to make a unique match between the source and target genes. The criteria for a unique match involve a series of comparisons, surveyed in the order listed below until one or more matches are found. If multiple matches are found the gene name is considered non-unique, and if a single match is found the gene is considered a unique match.

1. Source gene '*Feature Name*' vs. target gene set '*Feature Names*'
2. Source gene '*Feature Name*' vs. target gene set '*Gene Names*'
3. Source gene '*Feature Name*' vs. target gene set '*Aliases*'
4. Source gene '*Gene Name*' vs. target gene set '*Feature Names*'
5. Source gene '*Gene Name*' vs. target gene set '*Gene Names*'
6. Source gene '*Gene Name*' vs. target gene set '*Aliases*'

The data associated with successfully matched genes is then transferred from the source **Data Set**(s) to new **Data Set**(s) associated with the target gene set.

Utilities→GO Enrichment→Source: gene names

It is often useful to know whether a set of genes share common traits. MochiView can identify these commonalities once it is provided with assignments of genes to Gene Ontology (GO) terms. In order to use this utility you must first import this information using *'Import→Gene Ontology→Ontology'* and *'Import→Gene Ontology→Gene-to-ontology assignments'*. This utility was inspired by Elizabeth Boyle's GO TermFinder utility[14], and uses a similar methodology for calculation of GO Enrichment using the hypergeometric distribution. The reader is referred to the GO TermFinder documentation and www.geneontology.org for background.

Usage

The second tab provides a space in which a space- or return-delimited gene list can be typed or pasted. This list will then be searched for GO term enrichment using the settings defined in the first tab. The options provided in the first tab are described below, in top-to-bottom order.

Select Gene Set

This pull-down menu allows you to choose the **Location Set** of **Location Type** *Gene* that will be used for matching query genes to gene ontology assignments. Only those gene sets that have imported ontology assignments (using *'Import→Gene Ontology→Gene-to-ontology assignments'*) are listed.

P-value cutoff settings

This panel allows choice of the p-value cutoff for inclusion in the results table. (Note that the p-value entered here is non-transformed, whereas the p-value displayed in the results tab is -$\log_{10}$ transformed.) The cutoff can either be applied before or after a multiple testing correction. The correction is a simple Bonferroni correction in which the p-value is multiplied by the number of GO terms tested (note that this is a rather conservative correction, especially given the redundancy/overlap among GO terms).

Gene cutoffs for inclusion of ontology category

GO terms with a very large or very small number of assigned genes are often uninteresting. These settings allow control of the minimum and maximum number of assigned genes for inclusion of a GO term in the analysis. An additional checkbox provides the option to ignore all genes in the query list that have no assigned GO terms. This box is checked by default so that a list of query genes is not unfairly penalized for containing genes that are completely uncharacterized. Any such omissions will be listed in the results tab.

Use GO Slim

Gene ontology definitions (imported using *'Import→Gene Ontology→Ontology'*) often contain 'subsets', often called GO Slim sets, which define a more restricted range of GO terms that are either more relevant to a particular organism or less redundant. Any such subsets that were listed in the imported `OBO` file will be listed in the pull-down box in this panel.

Ontology aspects to include

GO terms are divided into three major aspects: component, function, and process. This panel provides the option for selecting which of the aspects to include (see the tooltips for descriptions of the aspects).

Ontology relationships to include

Three major relationships exist between GO terms: `IS_A`, `PART_OF`, and `REGULATES`. The precise definitions of these terms are beyond the scope of this manual (see www.geneontology.org). This panel provides the option to include one or more of these relationships when assigning genes to GO terms. If none of the options are selected a given gene is only assigned to the specific GO term(s) to which it was assigned. Otherwise, the selected relationships dictate to which additional GO terms the gene is "connected", and the gene is then also assigned to all other related (typically lower-specificity) GO terms.

Evidence codes to include

The assignment of a gene to a GO term is accompanied by an "Evidence Code", indicating the type of evidence supporting the assignment. This panel allows selection of which evidence codes are included in the search (a common use of this panel is to exclude all assignments that are not based on experimental evidence). Tooltips provide the full names of each evidence code.

Tips/Tricks/Gotchas

Why do my corrected p-values change after I adjust some settings?

The multiple testing correction multiplies the p-value by the number of GO terms considered. If you alter the settings such that more/less GO terms are included (e.g. including only one of the three GO term aspects), the corrected p-value will change.

Why am I not getting any significant results?

The multiple testing correction is quite conservative, so only very strong matches will be significant. To see more matches, try adjusting the '*P-value cutoff settings*' so that the cutoff is applied before multiple testing correction. (To view all GO terms that have at least one match against your submitted gene list uncheck the box in this panel and set the p-value cutoff to `1.0`).

Utilities→GO Enrichment→Source: Location Set

The general GO Enrichment function of this utility is the same as described for *'Utilities→GO Enrichment→Source: gene names'*. This utility differs in that the gene list tested for enrichment is derived from gene proximity assignments for a **Location Set** of your choice (gene proximity assignments and **Location Set** filtering are described in the manual section titled *'Common Menu Elements'*.

Utilities→GO Enrichment→Source: Motif scores

The general GO Enrichment function of this utility is the same as described for *'Utilities→GO Enrichment→Source: gene names'*. This utility differs in that two options are provided for identifying significant GO terms:

Mann-Whitney U (default)

In the default case the **Motif** score is calculated for all promoters and then tested for each GO term using a one-tailed Mann-Whitney U test (AKA Wilcoxon Rank Sum) to determine whether the **Motif** scores in the promoters of the gene mapped to the GO term tend to have higher scores than the remaining promoters (FYI, this is an oversimplification of how the Mann-Whitney p-value is interpreted). Please note that this statistical test is quite conservative.

When this test is used the result table looks a bit different than for other GO tests (hover the mouse over the tab to see an explanation of what each column contains). The most important difference is that two different types of p-value are reported. These values differ in how they handle the ranking of promoters with identical **Motif** scores. The Mann-Whitney test does not play nicely with such rankings, and MochiView reports the p-values from both a highly conservative and a highly 'optimistic' approach. The conservative approach always ranks the GO term promoters below the other promoters, while the 'optimistic' approach does the opposite. (Thus, these p-values are the upper and lower bounds on the "fuzzy" range of possible p-values from different ranking permutations.) The conservative P-value is the one used for determination of inclusion in the table (using the user-supplied p-value cutoff from tab#1).

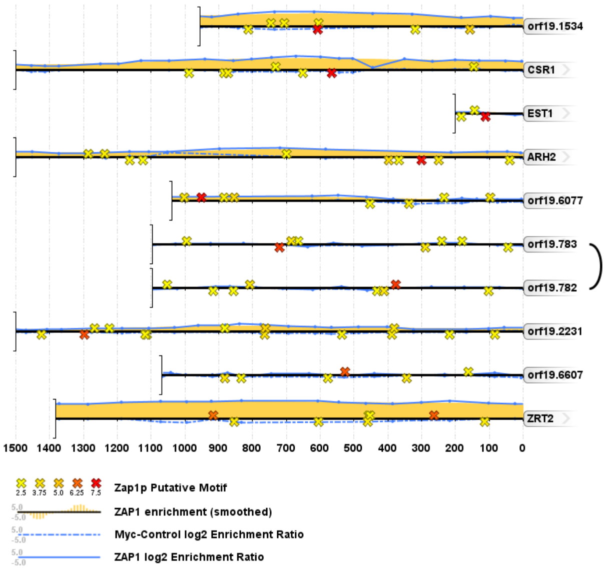
Motif score cutoff

This approach more closely resembles the approach of the other GO utilities (hypergeometric distribution), and assembles the query gene list by identifying all genes with promoters that have a **Motif** score exceeding a user-defined cutoff.

Utilities→Compact motif/data/location plot

Usage overview

This utility allows you to create a compact depiction of **Motif** matches and/or data from **Data Set**s and **Tiled Set**s on a highly customizable plot that can be saved as a `pdf` or `png` file. The utility was designed to produce publication-quality summaries of your data (you can further edit your `pdf` in Illustrator), and to provide an alternate and more compact view of your data. When **Motif** matches are plotted on promoters, this view also serves to make any non-random positioning evident (e.g. matches clustering 300-400bp from the 5'UTR).



Step#1: Select Sequence Set

When you launch the utility you are immediately prompted to select a **Sequence Set**. Your selection will influence the available range of **Location**s and data offered in the subsequent menu.

Step#2: Configure settings ('Plot Settings' tab)

This tab contains numerous options for controlling the layout and color scheme of the plot, all of which are supported by tool tips that appear when you hover your mouse over them. Why so many options? The defaults will suit you fine for many purposes, but the intent of the utility is to give you sufficient control to make publication-quality images.

Step#3: Configure the Location source ('Location Settings' tab)

Each line in the sample plot above represents a "**Location** entry". This tab contains a 'Maximum number of Locations' setting as well as three different options (in a pull-down box) for determining the **Location**s used in the plot.

Maximum number of Locations

Due to memory constraints only a limited number (up to 100) of **Location**s can be displayed. If more are provided, the 'top' **Location**s are retained by first sorting using the following criteria:

1. If the **Location**s were promoters fetched using a user-provided list of gene names, the order of the entered list is retained.
2. If any **Motif**s were selected, **Location**s are sorted in descending order of the maximum **Motif** match score.
3. **Location**s are sorted in ascending order based on the **Location** name.

Location source: Gene promoters

Promoters are extracted using user-defined promoter extraction criteria, as described for the *Utilities→Location Set→Build promoter set* utility. An option is provided to restrict the promoter set using a list of gene names. Note that if you use this option the **Location** name is written exactly as provided (as long as it uniquely matches a gene in the database); otherwise, the **Location** name is the *Gene Name*, or, if the gene has no *Gene Name*, the *Feature Name*.

Location source: Location Set(s)

You may choose one or more **Location Set**s as the source of **Location**s. If *Annotation Tag*s are assigned to the **Location**s, they are used as the **Location** name. Otherwise, the **Location** name is either the **Location** coordinates (if the 'Use coordinates as **Location** name' checkbox is checked) or the name is left blank. Note that if you use the coordinates as names you may want to widen the available space for writing the **Location** names in the 'Plot Settings' tab.

Location source: Sequence coordinates (with optional names)

You may enter a return-delimited list of sequence coordinates with an optional, bracketed, **Location** name at the end of each line. Here are some examples of acceptable formats:

```
sequencename:1000-5000 + [my gene name]
sequencename 1,000 5,000 -
sequencename:1000-5000 + [my gene name]
sequencename 1000 5000 -
sequencename 5000 1000 -
```

```
sequencename:1000-5000 [my gene name]
```

- The **Sequence** name and first coordinate can be separated by either a colon or whitespace (whitespace includes spaces and tabs).
- The **Sequence** name is not case-sensitive, provided that this does not result in ambiguity.
- The two coordinates can be separated by either a hyphen or whitespace and may contain commas.
- Optionally, following the coordinates you can have whitespace followed by a plus or minus to indicate strand. If this is not include, strand direction is inferred by the order of the coordinates (if the first coordinate is <= the second, the strand is interpreted as plus strand.
- Optionally, you may finish with whitespace followed by bracketed text that will be used as the **Location** name.

Step#4: Choose Motifs to scan ('Motifs (optional)' tab)

This tab allows you to choose **Motif**s that you would like to scan against your **Location**s. The **Motif** matches are reported as either -$\log_{10}$ p-values or as LOD scores for PSFM **Motif**s (there is a checkbox to control which) or as Affinity scores for PSAM **Motif**s. The tab also includes an option to designate which **Location Set** is used to generate the Markov model for scoring.

Motif selection table

The lower portion of the tab contains a table of **Motif**s that can be selected for inclusion using the checkboxes in the first column. The second column contains a button that displays a representation of the shapes and color spectrum used to display **Motif** matches. Clicking the button opens a menu that can be used to adjust the shape types and size, the color scheme, and the score range corresponding to the color scheme.

Step#5: Choose Data Sets to plot ('Data Sets (optional)' tab)

This tab allows you to select **Data Set**s that can be plotted on the **Location**s in either *Line Style* or *Bar Style*. Click the checkboxes in the first column to include the corresponding **Data Set**. The second column contains a button that displays a representation of the display style. Clicking on the button opens a configuration menu that resembles a limited version of the one found in the 'NEW PLOT' menu. The only option specific to this menu is:

Select data range

This option controls the minimum/maximum value of the y-axis. For example, if you entered '`3`', the y-axis would range from `-3` to `3`, with `0` being the **Location** line. If a value falls outside of this range, it is adjusted to fit (e.g. `-4.7` would become `-3`).

Step#6: Choose Tiled Sets to plot ('Tiled Sets (optional)' tab)

This tab is identical to the '*Data Sets (optional)*' tab, with the only exception being that the table displays **Tiled Set**s.

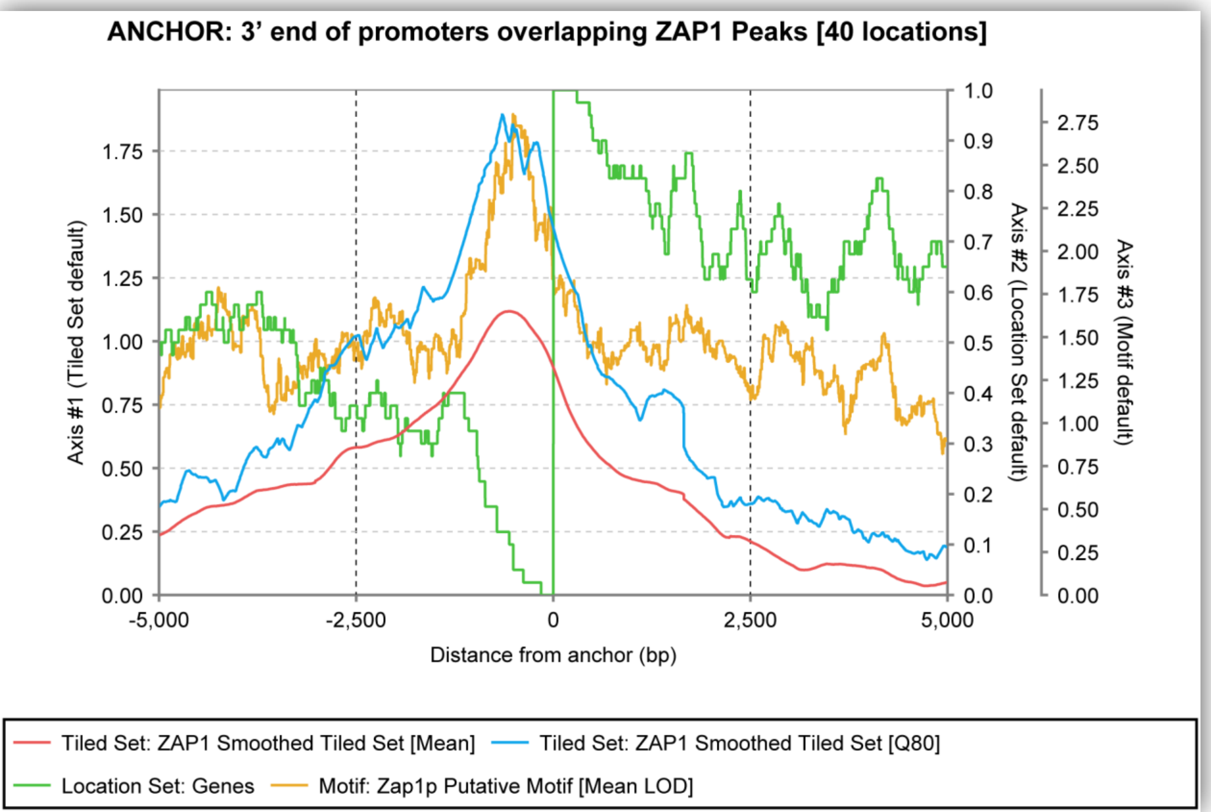
Step#7: Create plot and save image if desired

`Click` the GO button and your plot will be generated in a new tab. (You can create multiple plot tabs, though you might run out of memory if you make multiple tabs with especially large images.). The plot tab contains buttons providing the option to save the image as a `pdf` or `png` file.

Utilities→Anchor plot

Usage overview

This utility creates an XY-plot in which the X-axis is position relative to a set of "anchor" **Location**s (upstream/downstream or left/right) and the Y-axis represents the data value at that position across all anchor **Location**s (e.g. mean/median/maximum) for a given source of data. Anchor **Location**s can be derived from either a **Location Set** or from matches to a **Motif** that exceed a given cutoff. The sources of data can be **Motif** scores, **Tiled Set**s, or **Location Set**s (in this case the value indicates presence/absence).



This utility takes some work to learn, but it is one of the most powerful and flexible utilities in MochiView and is capable of making publication-quality images of your plots. Each tab has its own tooltip, as do most of the items within the tabs. There are details in those tooltips that are not all repeated here!

Note about quantiles

For computational reasons, using quantiles (e.g. median) rather than the default (mean) to calculate y-axis values will result in much slower calculations, especially if there are a lot of anchor **Location**s or if the scan distance is large.

Tab: Anchor Settings

This tab allows you to pick the set of anchor **Location**s and specify both the anchor point (e.g. midpoint or 5' coordinate) and the distance to either side of the anchor point to scan. Understanding the "plot x-axis is relative to anchor strand checkbox" is very important. If selected, the negative values on the x-axis refer to position ***upstream*** of the anchor point (i.e. it is strand-specific), otherwise the negative value refers to the position ***to the left*** of the anchor point (i.e. as if all anchors were on the plus strand).

Tab: Axes Settings

The plot can have up to six y-axes. (You assign one to each line you add to the plot.) In this tab you can configure the location (left/right) of the axes as well as whether they auto-range or have locked upper/lower bounds.

Tab: Plot Settings (AT/GC)

This tab provides settings for the plot layout such as font sizes and line stroke thickness. In addition, the option is provided to create a line that plots AT or GC%. For example, if GC% is chosen an [x,y] value of [-100,0.35] would mean that 35% of the bases 100bp upstream (or to the left… see anchor settings section above) contained either G's or C's.

Tab: Tiled Sets

This tab allows you to select one or more **Tiled Set**s to include in the plot. By clicking on the button with the representation of the line you can open a configuration menu adjust both the appearance of the line and the manner in which the y-values are calculated (e.g. mean, median, etc.). What if you want to display more than one line for a single **Tiled Set** (e.g. one showing median and another showing mean)? Click the "Add Line" button near the bottom of the configuration menu to create an additional line (it appears as a tab with the heading "Extra"). You can also click the "Full Spread" button to create tabs covering every possible quantile as well as the mean.

Tab: Location Sets

This tab works in a similar fashion to the **Tiled Set** tab, but instead of dealing with data points it deals with presence/absence of a **Location**. For example, if the [x,y] value is [0, 0.4], it means that 40% of the **Location**s in the chosen **Location Set** overlap the anchor point.

Tab: Motifs

This tab also works in a similar fashion to the **Tiled Set** tab, but it uses **Motif** scores as the data-points (either LOD/affinity scores or –log$_{10}$ p-values). Unless you are dealing with a scenario in which you expect an absolute spacing between anchor point and **Motif** (e.g. anchors made from matches to a **Motif** that always appears 10bp from the **Motif** being used to scan), you will want to smooth the data using the "*Scan flank smooth size (bp)*" option. For example, if displaying mean LOD values and the [x,y] value is [0, 1.5], and a smooth flank size of 0bp is selected, this means that the average LOD score at the anchor points is 1.5. However, if a smooth flank of 10bp was selected, it would mean that the average max LOD score within 10bp of the anchor point is 1.5.

Utilities→Create new Project

These utility allow the creation of a new **Project**. (Most import menus provide a ⊕ button to conveniently create these entries as needed.)

Utilities→Create new Data Type

These utility allow the creation of a new **Data Type**. (Most import menus provide a ⊕ button to conveniently create these entries as needed.)

Utilities→Preferences

'Preferred Location Sets' Tab

This tab contains options for controlling:

Active Gene Set

Although multiple **Location Set**s of the **Location Type** '*Gene*' can be imported for each **Sequence Set**, a primary **Gene Set** must be assigned for use with several aspects of MochiView function, including the *Gene Browser* and *Data Browser* (see below). This setting is persistent across MochiView sessions.

Location Set for Background Frequencies

This menu allows selection of which **Location Set** background frequencies should be used for calculation of PSFM **Motif** LOD scores. At the bottom of the tab is an alternative, allowing you to provide a universal set of background frequencies for all LOD calculations.

URL Prefixes

When browsing a plot, `R-click`ing on a gene while holding '`Alt`' will launch your default system browser with the *feature name* of the gene appended to the URL prefix provided in this column. You can enter a different prefix for each **Sequence Set**. (*Mac users*: hold both '`Option`' and '`Command`' and `L-click`.)

'Plot Settings', 'Plot Axis Settings, 'Additional Settings', and 'Legacy Settings' Tabs

Numerous additional options are available in these tabs, and are described with tooltips. Note that, in most cases, when these settings are changed they do not affect plots that are currently open. The primary exceptions are the options in the 'Plot Axis Settings' tab, most of which update open plots immediately.

'Database' Menu

This menu facilitates the import, export, and management of all MochiView databases.

Database→Manage Databases

At any given time, MochiView utilizes a single, "*active*", database. However, MochiView can contain multiple databases stored as compressed archives (compression reduces the database size to ~20% of the original size) and switch between them using this utility. In addition, this utility allows the export of archived databases that can be shared between MochiView users (or serve as backups).

The utility displays all available databases (*active* and *archived*) in a table. The various database management options are provided via buttons on the left-hand side of the utility, and are discussed in detail below.

Create Database

Click the '*Create*' button to create a new (i.e. empty) database. The database is initially created as an *archived* (i.e. not the *active*) database. To switch to the new database use the '*Activate*' button.

Import Database

Click the '*Import*' button to import a previously exported database (i.e. a backup or a database that has been shared with you by another user). The database will be imported as an *archived* (i.e. not the *active*) database. To switch to the new database use the '*Activate*' button.

Export Database

It is strongly recommended that you use this function to make periodic backups of your database!

Click the '*Export*' button to export an existing database (select a row from the database table first). All of your current settings files and preferences will be saved, as will the current state of all opened tabs. If you currently have tabs open, you will be given the option to lock the exported database in a '*View-only Mode*'.

View-Only Mode

When sharing data with a colleague or presenting data as the supplement to a publication, it may be desirable to freeze the current state of MochiView. View-only mode can serve this purpose, and does the following:

1. Removes the menu bar (i.e. **NEW PLOT**, **MANAGER**, and *Import/Export/Utilities/Skin* menus)
2. Disables tab-closing
3. Disables updating of the current plot configurations sessions (i.e. the plots open to the same place/state every time)

Please note that this mode is not intended to provide any form of data security… in fact, here are the instructions for unlocking an exported archive if you have a change of heart:

1. Open the archive with a utility that can read '`zip`' files (e.g. WinZip or 7zip).
2. Navigate into the 'SETTINGS_FILES' folder and open the file called '*GlobalSettings.properties*'.
3. Delete the line that says 'LOCKED=true'.
4. Save the file.

When sharing a database you may wish to only share a portion of the contents of your database. The best way to do this is export the database, re-import it (i.e. make a copy), activate the copy, and delete the database items and settings files that you do not wish to share.

Activate Database

Click the '*Activate*' button after selecting an *archived* database from the table to *activate* the database. The currently *active* database will be compressed and stored as an *archived* database with all preferences and settings intact.

Delete Database

Click the '*Delete*' button after selecting an *archived* database from the table to completely delete the selected database. Note that you cannot delete the currently *active* database.

Rename Database

For identification purposes, each MochiView database must be given a unique name. Click the '*Rename*' button to change the name of a database selected in the table.

Database→Defragment database

Over time your database can become heavily fragmented (especially when deleting items), reducing performance and increasing its size. This utility defragments and compresses your current *active* database. This process can take a considerable amount of time (very roughly 5min/GB).

Database→Reclaim memory (purge cache)

MochiView uses a lot of memory, but often it is just holding data for quick retrieval and can release quite a bit of memory to your system. If you like to leave MochiView open on your computer for long periods of time, it might be worth clicking this every once in a while.

'Skin' Menu

This menu contains the option of switching between several skins, courtesy of the Substance look and feel (https://substance.dev.java.net/). The chosen skin will be remembered across sessions, and has a purely cosmetic effect. If the skin looks bad, it is probably my fault and not that of Substance ☺.

Scoring and Identifying Motifs

One of the central features of MochiView is the variety of options for exploring **Motif** enrichment in DNA sequence. These options are described in detail throughout the manual; here, the focus is on a more general overview of the implementation and available approaches to **Motif**-finding. (For a walkthrough of some of the basic **Motif** analysis functionality, you may also want to try the tutorial available on the website.)

Overview of Motif Scoring

Position Specific Frequency Matrix (PSFM) Motifs

These **Motif**s are represented by a matrix representing the frequency (`0.0` to `1.0`) of each of the four bases at each position of the **Motif**. For a tutorial on the calculation of PSFM **Motif** LOD scores, I recommend two primers written by Patrik D'haeseleer:

- *What are DNA sequence motifs?*[15]
- *How does DNA sequence motif discovery work?*[16]

In brief, a window the size of the **Motif** is scanned along the target sequence, and at each position a log-likelihood score is calculated. The sum of the scores for each position yields the LOD score, which reflects the likelihood that the sequence in the window was generated by the **Motif** model.

Position Specific Affinity Matrix (PSAM) Motifs

These **Motif**s are represented by a matrix representing a scaled measure of the free energy of binding (`0.0` to `1.0`) for each of the four bases at each position of the **Motif**. Details on the calculation of PSAMs using the program MatrixREDUCE can be found in the following article by Barrett Foat of the Bussemaker laboratory:

- Statistical mechanical modeling of genome-wide transcription factor occupancy data by MatrixREDUCE[4]

MochiView displays PSAM logos in the fashion described in this article, and calculates **Motif** scores using the algorithm presented in Figure 1 of this article, with the exception that the final score is not log-transformed. Thus, for each sequence window the score will range from `0.0` to `1.0`, with `1.0` representing the optimal match.

Please note that the information content displayed in bits for PSAM **Motif**s is only a rough approximation. (For each position, the bit value is calculated as $\log_2(1.0/Savg)$, where `Savg` is the average score for the four bases at that position, and the final information content is the sum of these values across all positions.)

Overview of Motif Scoring Schemes

There are many options for scoring **Motif**s provided in MochiView, differing in whether they are appropriate for PSFMs, PSAMs, and whether they are a score for individual windows (i.e. a stretch of sequence the length of the **Motif**) or an entire **Location**. These scoring options are summarized in the table below and then discussed in greater detail afterward.

SCORE TYPE	REGION	PSFM	PSAM	OVERVIEW
LOD	Window	YES	NO	Likelihood the window sequence was generated by the Motif model rather than the background model.
Affinity	Window	NO	YES	Affinity of the window sequence for the Motif model (does not use background model).
Maximum	Location	YES	YES	Maximum score (LOD or affinity) found in all windows of the Location.

SCORE TYPE	REGION	PSFM	PSAM	OVERVIEW
Cumulative	Location	KLUDGE	YES	Combined score from all windows in the Location (for PSFM only LOD score > 0 are considered).
W-Score	Location	YES	KLUDGE	HMM-based cumulative score calculation with inherent weighting for weak/strong sites.
P-Value	Location	YES	KLUDGE	Frequency with which samples from the background model produce an equivalent or higher LOD score.

P-values for Motif Matches

The PSFM motif LOD score gives an idea of the likelihood that a sequence match was generated by the **Motif** rather than the background model, but does not tell you how often sequences sampled from the background model will have a similar or better LOD score. P-values can be used to address this question, and the MochiView p-value implementation has the added benefit of allowing incorporation of a higher-order Markov model. (Note that p-values are not intended for use with PSAMs… MochiView uses LOD scores from a crude conversion of the PSAM to a PSFM to approximate p-values for PSAMs.)

Many utilities offer the option to use a -$\log_{10}$-transformed p-value in place of the **Motif** score. When this option is chosen the LOD score is assigned a p-value using the approach described by Barash *et al.* in their paper titled "CSI: Compound importance sampling method for protein-DNA binding site p-value estimation"[17]. The relevant parameters used by MochiView are:

- A mixture of 10 distributions is used
- 81,739 samples are taken (ranging from 2,000 samples for the motif to 20,000 samples for the background model)
- The above process is repeated using the reverse motif (to simulate the minus strand)

Note that this approach yields an estimate of the p-value, and the actual values will change slightly between runs (the algorithm is tuned to reduce variation in the more interesting low p-value range).

P-values are currently offered as offered as an option in the following utilities (this list may be a bit out of date):

1. *Utilities →Motif →Enrichment Plot*
2. *Utilities →Motif → Create Location/Data Set from Motif matches*
3. *Utilities →Motif →Create Data Set from Location Set Motif scores*
4. *Export →Motifs →Scan Location Set and export Motif matches*
5. *Export →Location Set →Format: MochiView (with optional Motif scoring)*
6. *Export →Location/Data Set →Format: MochiView (with optional Motif scoring)*

In those cases when the p-value is applied to a **Location**, the p-value reflects the p-value of the match with the highest LOD score within the **Location**. An option is provided to correct for multiple hypothesis testing, in which case the p-value is multiplied by twice the length of the **Location** (to account for searching both strands).

Cumulative Scoring and Sinha's w-score

Certain utilities in MochiView provide the option of scoring a **Motif** against **Location**s in a **Location Set**. By default, these scores are calculated as the maximum score (on either strand) within the **Location**. You may also be given the option to use one of two additional scoring schemes, *w-scores* and *cumulative* scoring, both of which provide an estimate the number of matches to the **Motif** in the **Location**, weighted by the strength of the match. These schemes are offered as options in the following utilities (list may be out of date):

1. *Utilities →Motif →Enrichment Plot*
2. *Utilities →Motif →Create Data Set from Location Set Motif scores*
3. *Export →Location Set →Format: MochiView (with optional Motif scoring)*
4. *Export →Location/Data Set →Format: MochiView (with optional Motif scoring)*

Sinha's w-score

This scoring approach is based on a Hidden Markov Model and is described by Saurabh Sinha in a 2006 Bioinformatics paper[18]. This is a much more sophisticated approach than the *cumulative* score described below, but is also more time-consuming to calculate. The details are provided in the referenced paper, but note that:

- The transition probability from the background frequencies and the query **Motif** is calculated as ($0.5/S_L$), where S_L is the length of the **Location** (`0.5` is used because both strands are tested).
- The background frequencies used are those chosen as the default for **Motif** scoring in the *Preferences* menu.

Cumulative Scoring

This is a simple scoring approach that takes the sum of each scoring window in the **Location** as the final score. Keep in mind the following quirks to this scoring scheme:

- For each scoring window, only the best score from the plus/minus strands is considered.
- In the case of PSFM **Motif**s, only LOD scores above zero are included.
- No correction for **Location** length is added, so be careful when interpreting **Location** scores if the length of your **Location**s varies.

Background Frequencies

Whenever a **Location Set** is imported into MochiView, the frequencies of the bases '`A`', '`C`', '`G`', and '`T`' at the imported **Location**s are calculated and stored. For very large genomes this calculation can be time-consuming, so the option is given to simply assign the **Location Set** the same base-frequencies as the full **Sequence Set**. *Utilities→Preferences* provides an interface for assigning the default **Location Set** to utilize as a background model when calculating PSFM **Motif** LOD scores in a plot.

The base frequencies serve to inform the log-likelihood calculation for a PSFM **Motif**. Put simply, if a **Location Set** contains a very high bias towards '`A`'s, a **Motif** model in which each position contains a high frequency of '`A`'s becomes uninformative. Conversely, if '`A`'s are rare in the **Location Set** sequence, the presence of an '`A`' in the sequence and the **Motif** can contribute strongly to the overall LOD score.

Implementation Details (not too important)

In practice, ChipView averages the frequencies of the complementary bases before making these calculations. Otherwise, for example, if a sequence was 90% '`G`'s on the *plus* strand, a PSFM **Motif** of '`CCCCCC`' would have a much better LOD score for a match on the *plus* strand than the *minus* strand. While there might be scenarios in which **Motif** strand-preference is known and this behavior would be desirable, this is typically not the case.

Pseudocounts and the Zero-frequency Gotcha

When you import a PSFM **Motif** from a MEME[1] or Bioprospector[2] output file, information about the number of sequences used to generate the **Motif** is extracted and the option to add a pseudocount when importing the **Motif**s is provided. Pseudocounts are a means of down-weighting the strength of a **Motif** that is generated from a small number of sequences, and has little effect if a large number of sequences were used to generate the **Motif** (the specific pseudocount options are described in the tooltips).

Note that if you are scanning a **Motif** with a matrix that has values of zero (e.g. a column with `1.00` for 'A' and `0.00` for all other bases), those columns become an absolute requirement for having a score greater than "`-infinity`" (or zero in the case of a PSAM **Motif**). Sometimes this may be the desired effect, but if you wish to have some leniency in your **Motif** consider adding pseudocounts. Alternately, you can use the option that allows you to set the minimum frequency of each value in the matrix to eliminate all zero frequencies.

Markov Models

Overview

Many motif finders (including the one in MochiView) can use an N-order Markov model to refine their search. A Markov model is basically a description of the frequency of all possible combinations of bases in a sequence of length 'N+1' and lower. The simplest Markov model, zero-order, is the same as the background frequencies described above (i.e. frequency of the bases 'A', 'C', 'G', and 'T' in the submitted sequences). A first-order model includes these frequencies as well as the frequencies of all possible combinations of two bases ('AA', 'AC', 'AG', 'AT', 'TA', etc.), and so forth.

Usage

Why is a Markov model useful for motif finding? It gives the finder information about the composition of the background sequences, thus down-weighting motifs composed of commonly occurring (or co-occurring if the model is more than zero-order) bases. For example, if the intergenic regions of your organism are enriched for stretches of 'A's or a simple repeat sequence (e.g. 'CAACAA'), a Markov model constructed from these sequences will identify those patterns as less interesting. MochiView's motif finder does this using a scoring method suggested by Thijs *et al.*[19]. MochiView also provides a utility for exporting Markov models constructed from **Location Set**s (*'Export→Location Set→Format: Markov model'*) in a format that can be read by the motif finding software MEME[8].

Choosing a Markov Model Location Set and Order

It is important to carefully choose the **Location Set** used to create your Markov model. First, one must including enough sequence to generate adequate coverage for the order of Markov model chosen. A rough rule of thumb is that the **Location Set** should have at least $10x4^{N+1}$ bp of sequence, where 'N' is the order of the Markov model. Second, one must choose **Location**s that reflect the background sequence composition of the type of feature that you are interested in. Typically, this means one should construct the Markov model from the intergenic regions or promoters in the **Sequence Set**. While one could simply use the same sequences that are being searched to construct the model, this is not ideal as the enriched sub-sequences of interest would have higher frequencies in the model and be down-weighted (whereas their impact on the overall frequencies is greatly diminished if the set of all intergenic regions is used).

Viewing Imported Motifs

There are four primary modes for viewing/browsing **Motif** hits in a plot, each having its own pros and cons.

1. Motif Score Track

When one or more **Motif**s are selected in the *Motifs* tab of the **NEW PLOT** menu, MochiView scans these **Motif**s against the sequence as you move around the plot.

Pros

- MochiView has a specially designed *Track* for displaying hits for **Motif**s on multiple plots without overlapping.
- You can double-L-click on a **Motif** hit and view the **Motif** logo juxtaposed against the sequence of the **Motif** hit.
- You aren't committing any additional data to the database.

Cons

- You cannot search the **Motif** hits (or browse through them in the *Data Browser*).
- The **Motif** hits are not available for use with any of the general utilities for **Location Set** manipulation.

- Scanning **Motif**s on the fly can slow down MochiView plot browsing.

2. Motif Hits as Data Set

The *'Utilities →Motif →Create Location/Data Set from Motif matches'* utility can be used to convert **Motif** hits above a user-defined score cutoff to a **Data Set** (with the **Motif** scores as the values assigned to the **Motif Location**s).

Pros

- The converted **Data Set** can be traversed with the *Data Browser* (i.e. sort by highest score and jump from hit to hit).
- The **Data Set** (and **Location Set**) can be used with many of MochiView's other features (e.g. filtering, export, **Location Set** manipulation utilities…).
- You can constrain the **Motif** search to specific **Location**s (e.g. promoters)
- You can apply an alignment constraint to the **Motif** hits.

Cons

- The **Data Set** is no longer associated with the **Motif**, and thus you cannot view the **Motif** logo via the **Data Set**.

3. Motif Hits Mapped onto a Location Set

The *'Utilities →Motif →Location Set →Create Data Set from Location Set Motif scores'* utility allows you to score **Location**s in an existing **Location Set** (e.g. a promoter or intergenic region set) for their highest (or cumulative) score. This approach shares some of the pros/cons of the *'Utilities →Motif →Create Location/Data Set from Motif matches'* utility, plus:

Pros

- By mapping to an existing **Location Set**, you may reduce the number of hits to a more manageable size (e.g. 1 value per promoter) that is more easily browsed in the *Data Browser*.
- If you choose the cumulative score setting you may gain information about regions with clusters of weaker sites.

Cons

- Knowledge of multiple hits within a **Location** (and their precise coordinates) is lost.

4. Search Using the *Sequence Browser*

Rather than importing a frequency matrix, it is also possible to search for a degenerate sequence using IUPAC symbols and the *Sequence Browser*.

Pros

- Does not require a frequency matrix.

Cons

- No permanent record of the **Motif** hits.
- Less precise than the other methods.

Searching for Motifs (tips and tricks)

MochiView leaves the heavy lifting of *de novo* **Motif** identification to the many pre-existing **Motif**-finding algorithms (for example, see the comparative analysis of many of these algorithms conducted by Tompa *et al.*[20]). However, the built in *'Utilities →Motif →Finder'* utilities are quite effective at finding strongly enriched **Motif**s (see the section in the manual for these utilities for details).

MochiView focuses on facilitating analyses of existing **Motif** matrices and the refinement and export of input sequences for use by external search algorithms. Below, some tips are provided on how best to utilize these tools to identify **Motif**s in your **Location Set**s.

Refining a Location Set for Motif Finding

In a typical scenario, a MochiView user may import raw ChIP-chip data (i.e. array probe plus enrichment and/or p-value) in addition to a **Location Set** that consists of binding regions called by an external algorithm. The underlying assumption is that the peak probe-enrichment in these binding regions is likely to be centered at or near a binding site. Thus, it is often advantageous to refine the binding region prior to searching for **Motif**s.

Eliminating incorrect/non-specific binding locations

No algorithm for binding-region determination is perfect, and through manual curation it is often possible to identify faulty calls. These **Location**s can be eliminated from a **Location Set** using hand-curation in *Edit Mode* followed by filtering of the **Location Set** using the *'Utilities→Location/Data/Tiled Set→Refine Location/Data Set'* utility. Similarly, regions that are also "enriched" in a control experiment (e.g. a pull-down in a strain with no epitope tag) can be eliminated using the *overlap filter* in the *'Utilities→Location/Data/Tiled Set→Refine Location/Data Set'* utility.

Constraining binding regions to intergenic space

It is often desirable to remove any overlap between a predicted binding region and open reading frames, as ORFs are considered less likely to contain a valid binding site. A **Location Set** containing all intergenic regions can be created by using *'Utilities→Location Set→Merge Location Sets→Subtraction'* utility to subtract a **Location Set** of **Location Type** '*Gene*' from a **Sequence Set**. Next, the *'Utilities→Location Set→Merge Location Sets→Intersection'* utility can be used to create a new **Location Set** consisting of the intersection of the intergenic **Location**s and the **Location Set** of binding regions. If you would rather restrict the binding regions to promoter regions that are within 'X' bases of the start codon (i.e. refine the size of huge intergenic **Locations**), a **Location Set** created with the *'Utilities→Location Set→Build promoter set'* utility could be used in place of the intergenic **Location Set**.

Constraining Locations to a maximum length

Some utilities provide the option of restricting the size of each exported **Location** using a designated flank length that is centered at the highest value in an associated **Data Set**. The typical application of this option is to limit the sequence of a binding region to a few hundred base pairs centered at the ChIP-chip probe **Location** with the highest associated enrichment value (or p-values).

Extracting peaks of fixed length

A more sophisticated version of the option above is proved by the *'Utilities→Location/Data/Tiled Set→Extract peaks from Data Set(s)'* utility, which identifies peaks of a defined length in ChIP-chip data based on a set of user-defined criteria.

Masking Non-conserved Sequences

With the expectation that binding site sequence is more highly conserved than the surrounding promoter sequence, it can be helpful to mask poorly conserved sequence prior to searching for **Motif**s.

If you have imported a genome alignment, the *'Export→Location Set→ Format: FASTA'* utility provides the option of masking your exported binding site **Location**s in this manner. This approach is only useful for very closely related species (or intra-species alignments) in which the promoters have not diverged to the extent that they cannot be aligned.

Searching subsets of binding regions

In cases where a **Location Set** of binding regions might contain multiple different **Motif**s (e.g. a transcription factor that obtains some/all of its specificity via cofactors), it can be helpful to search a subset of binding regions for a **Motif**. For example, once the full set is searched and reveals a **Motif** that is associated with half of the binding sites, one could search the remaining half independently in an attempt to identify a second binding **Motif**. In such a scenario, one could import the first **Motif**, utilize the *'Utilities→Motif→Create*

Location/Data Set from Motif matches' utility to score the **Motif** against the full set of binding regions, and then export the **Location Set** using the *'Export→Location Set→ Format: FASTA'* utility while using a *data filter* to eliminate **Location**s with a high associated **Motif** score.

Analyzing Existing Motifs

The '*Import Motifs*' utility can be utilized to populate the MochiView database with a large library of **Motif**s. For well characterized organisms such as *Saccharomyces cerevisiae*, numerous meta-analyses of transcription factor binding sites have conducted, and these data can be reformatted for import into MochiView. **Motif**s represented by simpler *IUPAC* descriptions (e.g. '`A[CT]GG[CG]`') can either be reformatted into faux-frequency-matrices (*'Utilities→Motif→Create Motif from IUPAC'*), turned into a **Location Set** (*'Utilities→Location Set→Create Location Set from sequence matches'*), or identified using the *Sequence Browser*.

Motif quality control

Be aware of the information content of your **Motif**s. A low information **Motif** will occur by chance all over the genome. If you have reason to believe that the **Motif** meaningfully reflects a binding specificity, try to uncover what other factors lend specificity and incorporate them into your analysis (e.g. association with other **Motif**s, proximity to transcriptional start, etc…).

Motif frequency analysis

Once the database has been populated with a set of known **Motif**s, any existing **Location Set** can be tested for enrichment of these **Motif**s using *'Utilities→Motif→Enrichment Table'*. Many of the methods outlined in the previous section can be utilized to refine the **Location Set**s prior to testing and generate the appropriate control sets.

Motif predictive value

Ultimately, it is of interest to know how well a **Motif** accounts discriminates between the known binding regions (from ChIP) of the transcription factor thought to recognize the **Motif** and other comparable (e.g. promoter) regions. Does a LOD score cutoff exist such that all of the "true" binding regions contain a score above that cutoff but no "control" binding regions pass? The *'Utilities→Motif→Enrichment Plot'* utility addresses this question. If using this utility with extracted peaks of fixed length (from *'Utilities→Location/Data/Tiled Set→Extract peaks from Data Set(s)'*), try using control **Location Set**(s) of matched length **Location**s using the *'Utilities→Location Set→Sample fixed length Locations from Location Set'* utility on a **Location Set** of intergenic regions.

Motif distribution analysis

Does a **Motif** of interest show enrichment at a certain distance from start codons? Does the **Motif** co-occur with positional constraint with another **Motif** in the library? These questions can be answered by *'Utilities→Motif→Distribution→Relative to Locations'* and *'Utilities→Motif→Distribution→Relative to matches to another Motif'*, respectively.

Motif similarity analysis

Does a **Motif** of interest resemble any known **Motif**s? You can use the *'Utilities→Motif→Comparison'* utility to test whether newly discovered **Motif**s resemble any **Motif**s in your library (or to identify redundant **Motif**s in your library). A variety of **Motif** libraries are available on the MochiView website.

Plotting Motifs on a Track

Once likely candidate **Motif**s have been identified, they can be added to a *Track* on your MochiView plots, providing a visual representation of the **Motif** site locations relative to other genomic features (e.g. genes, ChIP-chip enrichment peaks) and each other.

Searching for Inverted Repeats

Most motif-finding algorithms are poor at identifying direct or inverted repeats that are separated by a variable length gap. MochiView provides robust searching for such repeats (see the *Sequence Browser*). This feature can be used to identify repeats on the plot. If any repeat coincides with the binding peaks it merits further investigation and refinement.

Keyboard Commands

GENERAL HOTKEYS:

Alt+P	Open **NEW PLOT** view
Alt+M	Open **MANAGER** view
Ctrl+F	*Opens search menu in **MANAGER** tabs and some **NEW PLOT** tabs
Esc	Closes most menus (equivalent to pressing the ⊗ button)

PLOT NAVIGATION HOTKEYS:

Space	Re-center plot around currently highlighted area
Shift+Space	As for Space, but also zooms to fit the width of the highlighted area
←/→	Scroll left/right along the X-axis
↑/↓	*Expand/contract X-axis display width
Shift+←/→	Page left/right along the X-axis
Shift+</>	Jump to the previous/next **Sequence**
1,2...0	Number key changes plot width to multiple of 10kb (Shift =100kb; Ctrl=1MB)
Home	Go to start of **Sequence**
End	Go to end of **Sequence**
Page Down	Contract plot to minimum allowed width
Page Up	Expand plot to maximum allowed width
Insert or F1	Resize plot to preferred width

PLOT BROWSER HOTKEYS:

Alt+L	Open the ***Location Browser***
Alt+G	Open the ***Gene Browser***
Alt+S	Open the ***Sequence Browser***
Alt+D	Toggle between ***Data Browser*** modes (*hidden*, *ribbon*, and *detailed*)
Alt+→/Alt+←	Select Next/Previous item in the ***Data Browser***

PLOT AXIS HOTKEYS:

Alt+Z	Rescale y-axis to fit currently visible plotted values
Alt+X	Restore y-axis to default min/max values (dictated by Data Types)

PLOT AESTHETICS HOTKEYS:

Ctrl+H	Adjust track heights and titles
Alt+V	Toggle line display (lines only, lines and makers, markers only)
Alt+C	Toggle line smoothing (natural cubic spline) on/off
Alt+B	Toggle gridlines on/off
Alt+H	Toggle legend size (the legend normally shrinks to avoid overlapping tracks… this option will enlarge the legend if it is not already full size)
Alt+N	Toggle legend on/off
Alt+K	Toggle sequence display on/off
Alt+J	Toggle sequence between single/double strand
Alt+T	Toggle track titles on/off
Ctrl+B	Toggle track background shading on/off

PLOT MISCELLANEOUS HOTKEYS:

Alt+Q	Open the **NEW PLOT** view initialized with the settings of the currently visible plot
Alt+W	As with Alt+Q, except current plot is closed
Ctrl+S	Open the *'Snapshot'* menu for making 'pdf' and 'png' files
Shift+I	Save an image of the current plot as a 'png' file
Ctrl+I	Save an image of the current plot as a 'pdf' file
Ctrl+Q	Copies an image of the currently visible plot area to the clipboard
Shift+Q	As with 'Ctrl+Q', except that the mini-panels in the bottom corners are omitted
Ctrl+C	Copies the sequence in the currently visible plot area to the clipboard
Shift+S	Save the current plot configuration settings

CHART LOCATION L-CLICK MODIFIERS:

Ctrl	*Copy reverse complement to clipboard
Alt	*Add FASTA-style header
Shift	*Copy ###bp centered on the midpoint of the clicked location (# chosen in *Utilities→Preferences*)

CHART LOCATION R-CLICK MODIFIERS:

Alt**	*If the mouse is over a gene name, web browser is launched.
Alt**	*Dragging the mouse with the button down will highlight a region for zooming upon release.

`Shift`	*Zooms to sequence-level view centered at x-axis value of clicked region

*Available only by keyboard (i.e. no equivalent menu command)

***Mac users*: hold both '`Option`' and '`Command`' and `L-click`.

Acknowledgments

This project would not have been possible without the support of Dr. Alexander Johnson, the members of the Johnson Laboratory, and many other people. I would like to single out two people for an extra big thank you… this project would not have been possible without them.

Dr. Brian Tuch

Dr. Tuch, a former member of the Johnson Lab, conceived of the approach to displaying *Motif Track*s and ChIP-chip data that is implemented in MochiView. A portion of the motif scoring code used by MochiView was written by Dr. Tuch.

David Gilbert (author of the JFreeChart library for Java)

The JFreeChart library (in a heavily modified form) is the core of MochiView's plots. The library is free and offered through the GNU Lesser General Public License (http://www.jfree.org/jfreechart/).

Software Testers

Several researchers at UCSF were kind enough to use MochiView and/or provide feedback while it was still a work in progress.

Johnson Laboratory

Lauren Booth
Christopher Cain
Sarah French
Dr. Aaron Hernday
Dr. Quinn Mitrovich
Dr. Clarissa Nobile

Sil Laboratory

Dr. Mark Voorhies

Yamamoto Laboratory

Samantha Cooper

DeRisi Laboratory

Dr. Polly Fordyce

Cheryl Chun Madhani Laboratory

Cheryl Chun

Design Inspiration

Apart from the contribution of Dr. Tuch (noted above), several additional sources strongly influenced the overall MochiView design.

UCSC Genome Browser[7]

The visual presentation and file formatting of the UCSC browser helped inspire the MochiView track and gene file format designs. In addition, the strategy employed for storing/retrieving location coordinates used by MochiView was adapted from the binning approach described in their paper.

WebLogo[5] Motif Display

The visual presentation of PSFM **Motifs** in MochiView is adapted directly from the WebLogo motif design concept.

TreeView[21]

The name MochiView is an homage to the java-based phylogenetic viewing software TreeView.

Java Libraries

MochiView utilizes several excellent Java libraries.

JFreeChart

As mentioned above, this library forms the core of MochiView's plotting capabilities.

Website

http://www.jfree.org/jfreechart/

License

GNU Lesser General Public License (LGPL), version 2.1

JCommon

This library is required for JFreeChart functionality.

Website

http://www.jfree.org/jcommon/

License

GNU Lesser General Public License (LGPL), version 2.1 or later

Substance

This library provides the "Look and Feel" of MochiView (including the '*Skins*').

Website

https://substance.dev.java.net/

License

Berkeley Software Distribution License (BSD)

Commons-Math

This library is used for statistical analyses.

Website

http://commons.apache.org/math/

License

Apace License v2.0

Derby

This library is used for JavaDB database access.

Website

http://db.apache.org/derby/

License

Apace License v2.0

iText

This library is used to create PDFs in snapshot mode.

Website

http://www.lowagie.com/iText/

License

Mozilla Public License v1.1

Code/Concept Contributions

A few sources on the web provided code that was utilized (or served as the inspiration for code) in MochiView.

TOMTOM (Motif Comparison utility)

The *'Utilities →Motif→ Comparison'* utility was originally coded and designed without knowledge of the existence of a similar utility called TOMTOM (Shobhit Gupta, Charles E. Grant, and William Noble; http://meme.nbcr.net/meme4/cgi-bin/tomtom.cgi). The MochiView comparison utility was significantly improved by incorporating the methodology for scoring similarity and calculating E-values described in the associated article by Gupta *et al.*[9].

GO TermFinder (GO enrichment utilities)

The general approach and design to the GO term analysis utilities was strongly influenced by Elizabeth Boyle's GO TermFinder utility[14] (though none of the actual code was used).

www.geneontology.org[22] (GO enrichment utilities)

The GO enrichment analyses would not be possible without the GO definitions and curated files provided by the GO consortium website[22].

MatrixREDUCE →PSAM Motifs and logos

The concept, logo design, and scoring algorithm for PSAM **Motif**s originated from the MatrixREDUCE program of Barrett Foat in the Bussemaker laboratory[3,4] (though none of the actual code was used).

BioJava[23] cookbook (Gibbs sampler)

Although BioJava is not actually used in MochiView, the code posted online for the BioJava cookbook Gibbs distribution sampler served as the basis (with very heavy modification and embellishment) for the *'Utilities →Motif →Finder'* motif-finding algorithm. Below I provide a citation for BioJava as a whole and also a link to the web page containing the code in question:

> BioJava: an Open-Source Framework for Bioinformatics
> *R.C.G. Holland; T. Down; M. Pocock; A. Prlić; D. Huen; K. James; S. Foisy; A. Dräger; A. Yates; M. Heuer; M.J. Schreiber*
> Bioinformatics 2008; doi: 10.1093/bioinformatics/btn397
> ***Cookbook:*** GNU Free Documentation License v1.3
> ***Biojava License***: LGPL v2.1
> ***Website***: http://www.biojava.org/wiki/BioJava:CookBook:Distribution:Gibbs

Motif finder Markov model incorporation

The algorithm used by MochiView's **Motif** finder for including Markov information while scoring sequences was adapted from the equation presented by Thijs *et al.*[19].

Transfer data between Sequence Sets

This utility required the ability to rapidly search a **Sequence Set** for sequence matches. The utility utilizes a hashmap created using an algorithm presented by Reneker and Shyu [13].

W-Scores for motifs

Saurabh Sinha for developed the w-score describing the number of matches to a **Motif** in a sequence[18]. Thank you to Mark Voorhies for his assistance in helping me to understand the algorithm implementation.

P-Values for motifs

The sampling approach for **Motif** p-value calculation employs approach described by Barash *et al*. in their paper titled "CSI: Compound importance sampling method for protein-DNA binding site p-value estimation"[17].

Timothy Wall

MochiView utilizes a few elegant classes of code written by Timothy Wall (under LGPL v2.1) to provide animated loading icons (http://rabbit-hole.blogspot.com/2006/09/animated-icon-redux.html).

Kumar T Santhosh

Heavily modified versions of a few of Santhosh's progress dialog classes (from the LGPL MySwing library) are used to display feedback while utilities run. I also used code from Kumar's excellent blog (http://www.jroller.com/santhosh/) to handle the display of modal windows.

Ed Buckler

MochiView's Fisher's Exact Test statistics are calculated using the FisherExact class written by Ed Buckler (in the TASSLE bioinformatics package: http://sourceforge.net/projects/tassel/)

Jacob Dreyer

Jacob's 'ColorUtil' class (under LGPL v2.1) was used to manipulate color shades.

"weihang" and "pstng"

These users of the JFreeChart forums provided code that was utilized when implementing line smoothing (http://www.jfree.org/phpBB2/viewtopic.php?f=3&t=20671&p=59377).

"Marlin314"

This member of the Java Forums provided the foundation of an algorithm for rapidly computing the union, intersection, and difference of two **Location Set**s.

"Torgil"

This member of the Java Forums provided some code utilized in handling rollover highlights in tables.

Additional Coding Tools

A few additional tools were used in this project.

Launch4J

This tool was used to make the Windows executable file.

Website

http://launch4j.sourceforge.net/

License

BSD License and the MIT License

Eclipse IDE

This IDE was used for writing all of the MochiView code.

Website

http://www.eclipse.org/

Known Bugs, FAQ, and Tips

Known Bugs

Open plots do not always update to reflect new data

Generally speaking, the best policy is to close all open tabs before doing anything that will update the database (e.g. importing data, modifying existing annotations, etc.). Here are a few specific limitations to how MochiView updates open plots:

- The *Data Browser* in plot tabs that are already open will not update to reflect newly imported data.
- Changes to a **Data Set** made using *Edit Mode* are reflected in the *Data Browser* of the edited plot, but not the others.
- Edits to an *Annotation Tag* will not necessarily be reflected in other open plots (or in other *Track*s within a plot).
- Changes to database annotations made with the **MANAGER** menu (e.g. changing the name of a **Location Set**) will not be reflected in currently open plots.

Rare cases of printing causing a freeze

In rare instances (possibly specific to certain printer drivers), if one attempts to print a plot and the printer is not available, the software may freeze.

Sluggish File Chooser in Windows

The file chooser window can be rather slow to respond when MochiView is used with the Windows OS. This is a known Java bug, and not something I can easily fix. It tends to be the most problematic if you browse into a folder containing a very large archive file (e.g. zip) or if you browse to somewhere on your network.

Sluggish Filters

Using filters on large **Location Set**s (~500,000+ **Location**s) is not advised, as plotting and *Data Browser* performance will become quite sluggish.

Performance Bottlenecks (not a bug, but worth knowing)

There are three primary performance bottlenecks in MochiView:

Database I/O

Retrieval of data from the MochiView database requires substantial hard drive activity. MochiView uses table indexing and memory caching to attempt to minimize hard drive access slowdowns, but there is only so much that can be done to speed a local database. As a general rule of thumb, MochiView is configured to speed database access at the expense of slower database import.

Background Frequency Calculation during Import (large Sequence Sets)

This bottleneck arises when background frequencies are being calculated for **Location Set**s that span very large distances (e.g. human genome genes), and can result in very slow import times. To speed import, it is helpful to use the alternatives provided in the import menu (either using the **Sequence Set** frequencies or manually entering the background frequencies).

Motif Score Calculation

MochiView can slow substantially when many **Motif**s are being scanned across large distances.

FAQ (work in progress…)

The NEW PLOT tables aren't displaying the full contents of my database… what is going on?

There are two likely possibilities. The first is that you have inadvertently pressed the '*Hide Unselected*' button in the menu (see the row of buttons at the bottom). The second possibility is that you have multiple **Sequence Set**s in your database, in which case MochiView (by design) only shows the database items associated with the currently selected **Sequence Set** in the '*Settings*' tab.

MochiView is giving me an error when I try to import my data… what am I doing wrong?

If the error message is not sufficiently informative to point you in the right direction, please try re-reading the file format description in this manual and making sure that your file is in the correct format. If you still have problems, please contact me and I will be happy to help (you may have encountered a bug or I may have not explained the file format clearly enough in the manual).

Why does the startup screen sometimes linger a while on "Initializing database…"?

This can occur the first time you start the application after making significant changes in the database. Don't worry, nothing is wrong with the database, it is just annoying but necessary database maintenance.

How can I display Location Sets with overlapping Locations?

Stack Mode is designed for this purpose. If you'd also like to separate the plus strand and minus strand **Location**s, use *Stack Mode* in conjunction with the *Directional: Axis Shape Style*.

How do I make a Location Set of intergenic regions?

Use *'Utilities →Location Set →Merge Location Sets →Subtraction'* to subtract your gene **Location Set** from the full **Sequence Set**.

How do I decide the best Motif score cutoff for displaying Motifs on a *Motif Track*?

There is no perfect answer, because the tuning of the cutoff will vary from **Motif** to **Motif**, and will also vary depending on your intent and your expectations of the genome-wide abundance of the **Motif**. The *'Utilities →Motif →Enrichment Table'* and *'Utilities →Motif →Enrichment Plot'* utilities can provide you with an overview of the **Motif** distribution at different cutoffs. If you have confirmed some **Motif**s as "real" (i.e. legitimate binding targets) you can use the score of these **Motif**s to calibrate your cutoff score. Many researchers begin with a (rather arbitrary) PSFM **Motif** cutoff of 75% of the maximum LOD score (this is what MochiView provides as the default). Alternatively, you can use p-value scoring and pick a p-value cutoff.

Can I merge two Data Sets that utilize the same Location Set?

This cannot be done directly in MochiView, but can be easily accomplished with the following steps:

1. Export the **Data Set**s of interest using *'Export→Location/Data Set→Format: MochiView'*.
2. Open the resulting file in Excel and make a new column containing the merged data (e.g. take the median or the mean).
3. Save the file as a text file and re-import into MochiView using *'Import→Data Set→Format: MochiView (by Location)'*.

Tips

- Back-up your database frequently using *'Database→Database manager'*!
- P-values are often easier to display when transformed to -$\log_{10}$.
- Expression ratios are easier to interpret when transformed to $\log_2$.
- Take a moment to configure the preferences menu to your liking…
- Hotkeys are your friends… especially '`Alt+W`' to revise the settings of a currently open plot.
- Use the *'Database→Defragment database'* utility from time to time to help keep things speedy and shrink your database.
- Every time you launch MochiView you'll see a tip in the bottom left corner of the window (unless you have plots open from the previous use).

Appendix A: Tips for displaying RNA-Seq data

Overview

This appendix provides a very brief overview of how MochiView can be used to display and browse RNA-Seq data.

Step#1: Format data for import

The key is to create a wiggle format file of your sequence read counts (typically at 1bp resolution using a sliding window) and import the file as a **Tiled Set** using the *'Import→Tiled Set→Format: WIG'* utility. Create a *separate* **Tiled Set** for the plus- and minus-strand counts (**Tiled Set**s have no notion of strand).

Note that this utility has several convenience options that allow you to apply log transformation or give all values a + or − sign. Utilize the latter to make all plus-strand sequence counts positive, and all minus-strand sequence counts negative.

Step#2: Display the data

The most convenient way to display a **Tiled Set** is to use a track and a bar display, and if you have separate plus- and minus-strand **Tiled Set** tracks formatted as described above, you can have these share the same track (in the '*Track Axis Order*' tab of the NEW PLOT menu).

Step#3 (Optional): Make a Data Set for browsing

Since **Tiled Set**s are too data-rich for use with the *Data Browser*, you will likely want to create a supplemental **Data Set** that you can use to browse areas of interest in your data. One option is to map the **Tiled Set** to genomic features of interest (e.g. genes or intergenic regions) using the *'Utilities→Location/Data/Tiled Set→Map Data Set(s) to Location Set'* utility. Alternatively, you can create a file outside of MochiView and then import the file using one of the *'Import→…'* utilities.

Appendix B: Tips for analyzing/displaying CGH data

Overview

This guide gives instructions on how to analyze your CGH data in R using the DNAcopy[24] library and then import the results into MochiView. It is strongly recommended that you download and read the DNAcopy manual (http://www.bioconductor.org/packages/2.2/bioc/html/DNAcopy.html) before following these instructions.

One time only: Install R and the necessary libraries

Install R

If it is not already on your computer, download and install R (http://www.r-project.org/).

Install DNAcopy

If you haven't already installed the *DNAcopy* libraries, do so by typing the following in R:

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("DNAcopy")
```

Formatting CGH data for input

Create a tab-delimited file (e.g. use Excel and 'Save as' *Text (Tab delimited)*) with one row for each array probe (no header row!) and the following information in each column:

Column#1: Chromosome names

For compatibility with MochiView, the names should match the name of the **Sequence**s in the MochiView database.

Column#2: Probe midpoints

This column should contain the midpoint of the probe. This value does not have to be an integer (e.g. 71.5), but in such cases you will have to round the segment boundaries if they are not an integer before you can import into MochiView. So, it is probably easiest just to use integers in the first place. (In Excel, you could use the floor() and average() functions to do this.)

Column#3: log$_2$-enrichment

This column should contain the log2-transformed enrichment ratio from your CGH array. Missing values are OK.

Analyzing the data using DNAcopy

1. Load the DNAcopy library

```
> library(DNAcopy)
```

2. Load your data into an R object (called 'cgh')

```
> cgh <- read.table("C:\\Users\\myFile.txt", sep="\t", col.names=c("Chromosome","Probe","Value"))
```

You will obviously need to substitute your own file path. The example above is for Windows file systems. (Note that the backslash is doubled... this is necessary to "escape" the backslash, which otherwise has special meaning in R.)

3. Load the 'cgh' object into a Copy Number Array object (called 'CNA.object')

```
> CNA.object <- CNA(cbind(cgh$Value), cgh$Chromosome, cgh$Probe, data.type = "logratio", sampleid = "myCGH")
```

4. Create a smoothed version of the 'CNA.object' (called 'smoothed.CNA.object)

```
> smoothed.CNA.object <- smooth.CNA(CNA.object)
```

5. Find segments and load into an object called 'segment.smoothed.CNA.object'

```
> segment.smoothed.CNA.object <- segment(smoothed.CNA.object, verbose = 1)
```

6. Visualize segments

DNAcopy offers several options for visualizing the segmentation results.

Try viewing the data on a per-chromosome basis:

```
> plot(segment.smoothed.CNA.object, plot.type = "s")
```

Try viewing the data sorted by ascending segment mean:

```
> plot(segment.smoothed.CNA.object, plot.type = "p")
```

7. Clean up the segments

The second plot in section 6 may show segments that are due to over-sensitivity of the algorithm (see the *DNAcopy* manual for details).

The segment calls can be cleaned up as follows:

```
> sdundo.CNA.object <- segment(smoothed.CNA.object, undo.splits = "sdundo", undo.SD = 3, verbose = 1)
```

Visualize whether this changed things:

```
> plot(sdundo.CNA.object, plot.type = "p")
```

If the change seems either too harsh or two weak, try re-running the cleanup with a different value for 'undo.SD' and re-visualizing.

8. Write the segment boundary data to a MochiView-compatible file

The set of segments identified by *DNAcopy* will become a MochiView **Location Set**, and the mean values within these segments will become part of a MochiView **Data Set**. A file that can be imported into MochiView using *'Import→Location/Data set→Format: MochiView'* can be created as follows:

This part writes the file header:

```
> write(c("SEQ_NAME","START","END","Segment avg"),"C:\\Users\\mochiFile.txt ", sep="\t", ncolumns=4)
```

This part writes the segment data (*it is all one line*):

```
> write(c(rbind(as.vector(sdundo.CNA.object$output$chrom), sdundo.CNA.object$output$loc.start,
sdundo.CNA.object$output$loc.end, sdundo.CNA.object$output$seg.mean)), "C:\\Users\\mochiFile.txt", sep="\t",
ncolumns=4, append=T)
```

Two more things…

[a] You will have to replace the file path your own desired file destination (the file should not exist, it will be created by R).
[b] If you did not do step#7 (cleaning up the segments) or if you would simply rather use the segments created before this step, just substitute *segment.smoothed.CNA.object* in place of *sdundo.CNA.object* wherever it is found in the two lines above.

Appendix C: Motif Finder Algorithm

Background

This **Motif** finder was developed using the basic Gibbs sampling code provided in the BioJava cookbook as a starting point and enhanced using the Markov model-based scoring method described by Thijs *et al.*[19]. The only code of note that I added was a few methods for producing optimized variants of a **Motif** (the "cull" and "loop" **Motif**s) after the initial **Motif** is identified.

Please note that I describe the algorithm to the extent that it is (hopefully) clear how the **Motif**s are identified, but I do not discuss most of the specific implementation details (e.g. speed optimizations). The source code is available if you would like this level of detail.

Generate Markov model

The user decides the order of the model and which **Location Set** sequences are used to derive it. The following rules are used when building the model:

1. A single "pseudocount" is added to every possible arrangement of bases in the model (so no sequence ever has a probability of zero).
2. Only the plus strand of DNA is used to create the Markov model (it is also used when fetching sequences from **Location**s or promoters, regardless of orientation).
3. Overlapping **Location**s are not counted twice (all such **Location**s are merged prior to generating the model).

Gibbs sampling overview

The basic premise of the Gibbs sampler is to start with an alignment built from a random position (and strand) on each sequence and then optimize this "motif" by iteratively taking a random sequence and choosing the position with the highest score against the current version of the "motif". (I'll refer to the motif-in-progress using quotes and the final product without.). The "motif" is then recalculated and the sampling process is repeated, until the stop condition is met. The sequence being sampled is chosen randomly for the first iteration, and is then incremented through the set of sequences in a loop, with the loop being disrupted 10% of the time by a new randomly chosen sequence.

Calculating the current "motif"

At any given time, the current "motif" is represented as a *base x position* frequency matrix constructed from the top 75% sequence windows (by score and not including the sequence being sampled; see below) contributing to the "motif" (on any given iteration there is one window position per sequence, hence the finder does not account for multiple motif instances on a sequence). A "pseudocount" is added when calculating the base frequencies. The pseudocount is equal to the frequency of the bases in the zero-order Markov model, with the additional stipulation that the frequencies of matched bases (`A` and `T`; `C` and `G`) are averaged. This averaging is necessary to consistently score the frequency matrix against either strand.

Why use only 75% of the sequences? This is a somewhat arbitrary cutoff that lets the finder converge on strong motifs that occur in most of the sequences more frequently than weaker motifs that are found in all of the sequences. (The final calculation once the iterations are complete uses the best scoring sequence window in *all* of the sequences.)

Scoring a sequence window against the current "motif"

Here I'll use the example of the sequence "`ACGGT`" being scored against a 5bp "motif" (*base x position* frequency matrix). First, the probability of the sequence being generated by the "motif" is calculated, by multiplying the frequency of each base in the sequence at the appropriate motif position (e.g. `0.23` x `0.40` x `0.05` x `0.91` x `0.85`). Next, this probability is divided by the probability that the sequence was created by the Markov model. This serves to penalize motif matches that resemble sequences of bases (e.g. repeats or runs of As or Ts) that are enriched in the background model. Finally, the score is normalized for sequence length by dividing it by the number of bases in the sequence (i.e. longer sequences do not score as well and are less likely to contribute to the "motif"). So, the full calculation is:

$$(P_{\text{"motif"}} / P_{\text{markov}}) / \text{seqLength}.$$

How is the minus strand scored? By comparing the plus-strand sequence to the reversed *base x position* frequency matrix.

Unfortunately, higher-order Markov models don't work well for sequence windows that contain non-ACGT bases (collectively considered as 'N'). In such cases, the background probability is calculated from the zero-order Markov model, and for any position containing 'N' the probability for that position is calculated as follows:

- **Probability generated by "motif":** max(lowest frequency base at the position, 0.1). This represents a compromise between not rewarding a sequence for containing an 'N' but also not punishing it too severely.
- **Probability generated by background:** max(frequency of 'N's in the sequence, 0.25).

If the sequence contains at least 25% Ns the final score is considered zero.

Avoiding local optima: Phase shift strategy

For every iteration of the sampler, there is a 10% chance that the current "motif" will be compared to the alternative "motif"s created when the current sequence positions are shifted one base pair to the right or left (if such a shift goes off the end, it loops to the other side of the sequence). This serves to shift a motif to its strongest core (if the search width is less than the motif width), and also provides a rapid means to alter the current windows early in the search when no strong "motif" has been found. The left- and right-shifted "motif"s are compared to the current "motif" using the following strategy:

1. Calculating the scores for the pre- and post-shift "motif"s and sequence positions. (Note that in this case all sequences are used except the current sequence being sampled.)
2. Summing the ranks of these scores when sorted in ascending order. For example, say the left-shift has scores of 4.2, 1.8, and 1.3 and the pre-shift has 3.5, 0.4, and 0.1. The ranks of the left-shift are 3,4,6 (sum 13) and the pre-shift are 1,2,5 (sum 8). The "motif" with the greatest sum (either left-shift, right-shift, or no-shift) is retained. Note that for the purposes of ranking all scores below 1 are considered as zero.

Avoiding local optima: Seed strategy

Rather than start the motif search from a single set of random sequence positions, the finder tests the waters with multiple such sets (the "seeds") and runs them for a limited number of iterations (500) before choosing the "best" seed to run to completion. The seeds are compared using the rank summing of scores described above for the shift strategy. The number of seeds varies between 25 and 100 depending on the search speed chosen by the user.

Avoiding finding the same motif over and over: Masking strategy

After finding the first motif, the finder masks all sequence windows that contributed to the motif with a score greater than 100 (a somewhat arbitrary cutoff). The masked region includes any sequence window that overlaps the masked window by at least 2bp. These masked regions are excluded from future sampling (and from phase shifts), but are not excluded from the final motif refinement (see below).

Deciding when to stop iterating

The heuristic used to stop the sampling iterations uses either a hard cutoff of "X" iterations or a conditional cutoff of "X" iterations without improvement (whichever comes first). These values are controlled by the user-selected search speed, and vary from 25,000 to 200,000 for the hard cutoff and from 2,500 to

25,000 for the conditional cutoff. Improvement is judged by a comparison of "motif" scores, as described in the phase shift section.

Finalizing the motif

Once the iteration stop condition has been met the motif is "finalized". This process involves the following:

1. The current frequency matrix (built from the top 75% scoring sequences) is used to find the best scoring window (and strand) for each sequence. Masks are ignored.
2. A new frequency matrix is built using the new windows, this time including all sequences. Pseudocounts are added to the matrix if the user has selected this option.
3. The windows and frequency matrix are adjusted by phase shift up to 10 times (if this improves the score; masks are ignored).

Refining the motif ([LOOP])

The refinement options in MochiView are in place to compensate somewhat for the lack of a true "zero or one motif per sequence" algorithm. The *LOOP* refinement reduces the number of sequences contributing to the motif as long as the information content increases (pseudocounts are used, so there is a very slight penalty for having fewer sequences) and the number of sequences does not fall below a user-defined limit. The *LOOP* proceeds as follows until those conditions are met:

1. Determine max number of sequences to remove in a given pass (10% of total number, minimum of one).
2. For each sequence, check whether removing the sequence and re-deriving the frequency matrix results in an increase in information content. If so, add the sequence to a list of potential sequences to drop.
3. Once all sequences have been checked, remove the sequences that created the largest gains in information (remove no more than the maximum determined in step#1, and no more than would reduce the total number of sequences contributing to the motif below the user-defined limit). If no sequences are removed, the *LOOP* is terminated.
4. Re-derive the frequency matrix from the remaining sequences and repeat the process (using those sequences only).
5. Reposition each sequence window to the best-scoring window against the new frequency matrix.

If a *LOOP* motif is generated (i.e. at least one sequence is dropped), it is reported in the finder.

Refining the motif ([CULL])

The *CULL* refinement essentially adds another layer of iteration to the *LOOP* refinement, with the potential for the refinement to drift to a very different motif. The *CULL* refinement is performed a maximum of 3 to 50 times (depending on speed settings), or until the *LOOP* refinement yields no gain in information. The *CULL* refinement is simply multiple iterations of the *LOOP* procedure described above in which iteration reconsiders all sequences (as opposed to *LOOP*, which starts with all but slowly whittles them away).

Software Design Comments

Much of the underlying design of MochiView was shaped by two decisions:

1. Exclusive use of Java as the programming language
2. Utilization of a local database, Java DB, which can be bundled inside the application and requires no installation

This approach maintains platform independence, and also allows users to easily archive and share databases.

Performance Considerations

The design of MochiView requires rapid data retrieval to ensure smooth panning and zooming within plots. Each set of chromosomal locations (e.g., array probes) is stored in a separate database table using the binning strategy developed for the UCSC genome browser [7]. Sets of data (e.g., raw data from a ChIP-Chip experiment) are stored in separate tables, rather than with the locations themselves. This approach prevents redundant entry and retrieval of heavily used location sets, such as array probes.

Further gains in retrieval speed are achieved using a memory caching scheme to limit database access. Requests for locations or DNA sequence are first checked against a cache of Java soft references. If the requested information is available it is provided without accessing the database, otherwise it is retrieved from the database and cached.

The database storage and caching approach described above is insufficient for the smooth display of very high-density data (e.g., single-base resolution ChIP-Seq read counts). These data can be stored and displayed as “Tiled Sets”, an approach modeled after the UCSC browser “Wiggle” format. In this format, the data are stored in a compressed binary file and the database is only used to store a table providing access points into the file at intervals across each chromosome. The compression format used in the binary file is similar to that described for the UCSC browser “.wib” file (http://genomewiki.ucsc.edu/index.php/Wiggle) with three enhancements. (1) Because data compression can result in a loss of precision, three different compression levels are offered, corresponding to the storage of each value in one, two, or three bytes. (2) The “.wib” format uses a byte to represent each position lacking data, which can be inefficient when large intervals without data are included. MochiView addresses this issue by using a single byte to indicate a “skip” command followed by a second byte indicating the number of positions to be skipped. (3) Multiple compressed files are generated, each corresponding to different position intervals (e.g., 1bp span, 100bp span, etc.). As a user zooms out in a plot, the plot dynamically switches between these files to prevent the inefficient processing of small-interval data when they exceed the visible resolution (bases per pixel).

Plot Design

The genome browser display utilizes a heavily modified version of the JFreeChart library (http://www.jfree.org/jfreechart/). This library was designed for static plot display; thus, numerous modifications were made to enable smooth panning, zooming, and rendering of large data sets. Additional changes were made to accommodate custom displays, such as the motif and alignment tracks.

Contact Information and License

Contact Information

I welcome requests for new features, bug reports, general questions, and any feedback on issues of clarity/usability. Please contact me (Oliver Homann) by e-mail:

oliver.homann@ucsf.edu

License

MochiView is available in source and executable forms, without fee, for academic, non-profit and commercial users. In order to prevent the sale of MochiView by third parties, the license (below) imposes restrictions on the redistribution of the software.

References

1 Bailey, T. L. & Elkan, C. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc Int Conf Intell Syst Mol Biol* **2**, 28-36 (1994).

2 Liu, X., Brutlag, D. L. & Liu, J. S. BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pac Symp Biocomput*, 127-138 (2001).

3 Foat, B. C., Houshmandi, S. S., Olivas, W. M. & Bussemaker, H. J. Profiling condition-specific, genome-wide regulation of mRNA stability in yeast. *Proc Natl Acad Sci U S A* **102**, 17675-17680 (2005).

4 Foat, B. C., Morozov, A. V. & Bussemaker, H. J. Statistical mechanical modeling of genome-wide transcription factor occupancy data by MatrixREDUCE. *Bioinformatics* **22**, e141-149 (2006).

5 Crooks, G. E., Hon, G., Chandonia, J. M. & Brenner, S. E. WebLogo: a sequence logo generator. *Genome Res* **14**, 1188-1190 (2004).

6 Darling, A. C., Mau, B., Blattner, F. R. & Perna, N. T. Mauve: multiple alignment of conserved genomic sequence with rearrangements. *Genome Res* **14**, 1394-1403 (2004).

7 Kent, W. J. *et al.* The human genome browser at UCSC. *Genome Res* **12**, 996-1006 (2002).

8 Bailey, T. L., Williams, N., Misleh, C. & Li, W. W. MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Res* **34**, W369-373 (2006).

9 Gupta, S., Stamatoyannopoulos, J. A., Bailey, T. L. & Noble, W. S. Quantifying similarity between motifs. *Genome Biol* **8**, R24 (2007).

10 Casimiro, A. C., Vinga, S., Freitas, A. T. & Oliveira, A. L. An analysis of the positional distribution of DNA motifs in promoter regions and its biological relevance. *BMC Bioinformatics* **9**, 89 (2008).

11 Reiss, D. J., Facciotti, M. T. & Baliga, N. S. Model-based deconvolution of genome-wide DNA binding. *Bioinformatics* **24**, 396-403 (2008).

12 Qi, Y. *et al.* High-resolution computational models of genome binding events. *Nat Biotechnol* **24**, 963-970 (2006).

13 Reneker, J. & Shyu, C. R. Refined repetitive sequence searches utilizing a fast hash function and cross species information retrievals. *BMC Bioinformatics* **6**, 111 (2005).

14 Boyle, E. I. *et al.* GO::TermFinder--open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics* **20**, 3710-3715 (2004).

15 D'Haeseleer, P. What are DNA sequence motifs? *Nat Biotechnol* **24**, 423-425 (2006).

16 D'Haeseleer, P. How does DNA sequence motif discovery work? *Nat Biotechnol* **24**, 959-961 (2006).

17 Barash, Y., Elidan, G., Kaplan, T. & Friedman, N. CIS: compound importance sampling method for protein-DNA binding site p-value estimation. *Bioinformatics* **21**, 596-600 (2005).

18 Sinha, S. On counting position weight matrix matches in a sequence, with application to discriminative motif finding. *Bioinformatics* **22**, e454-463 (2006).

19 Thijs, G. *et al.* A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics* **17**, 1113-1122 (2001).

20 Tompa, M. *et al.* Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotechnol* **23**, 137-144 (2005).

21 Page, R. D. TreeView: an application to display phylogenetic trees on personal computers. *Comput Appl Biosci* **12**, 357-358 (1996).

22 Ashburner, M. *et al.* Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet* **25**, 25-29 (2000).

23 Holland, R. C. *et al.* BioJava: an open-source framework for bioinformatics. *Bioinformatics* **24**, 2096-2097 (2008).

24 Venkatraman, E. S. & Olshen, A. B. A faster circular binary segmentation algorithm for the analysis of array CGH data. *Bioinformatics* **23**, 657-663, doi:btl646 [pii]
10.1093/bioinformatics/btl646 (2007).