**make\_heatmap**

Given a set of aligned read counts, microarray probe intensities, or other relevant value ascribed to a genomic location (*hit file*), a list of genes or other features of interest (*gene list file*), and a set of relative bin start and end locations (*bin file or definition*), produces a matrix of total or average values of hit file elements intersecting with each bin, for each gene list element.

Four distinct command line forms may be used:

make\_heatmap [options] [hit file] [gene list file] [output file] [bin file]

This form is for use with hit file types that include strand information or for running the program without utilizing any strand information.

make\_heatmap [options] -b c ... [output file] [bin start] [bin size] [bin count]

This form is used to automatically generate relative bin start and end locations rather than define them explicitly.

make\_heatmap [options] -b v ... [output file] [bin count]

This form is used to automatically generate a fixed number of variably sized bins per gene list element.

make\_heatmap [options] -p [plus strand hit file] -m [minus strand hit file] [gene list file] ...

This form is for use with hit file types that do not include strand information, especially in cases where such information is required by the selected options.

*Options*:

--help produce a help message describing usage and options

-p, --plus [plus strand hit file] specify file containing plus strand hits

-m, --minus [minus strand hit file] specify file containing minus strand hits

-t, --threads [number of threads] specify number of threads to use (default = 1)

-b, --bintype [method] specify method of defining relative bin locations:  
f file (default)  
c command line, fixed bin size  
v command line, variable bin size

-h, --hittype [type] specify hit file type:  
G standard bedGraph (default,0-based start  
 coordinate)  
g bedgraph (1-based start coordinate)  
b basic bed  
e extended bed  
c cppmatch

-l, --hitloc [location] specify location within hit to match to gene list:  
p physical start (default)  
d physical end  
s genetic start, requires -p, -m, -h c, or -h e  
e genetic end, requires -p, -m, -h c, or -h e  
c center

-a, --anchor [location] specify location within each gene list element to anchor relative bin locations:  
 s genetic start, requires stranded gene list   
 (default)  
 e genetic end, requires stranded gene list  
 p physical start  
 d physical end  
 u user-specified, contained in second column  
 of gene list file

-v, --binvalue [method] specify method of determining value for each bin  
t compute total of hit values (default)  
a compute average of hit values  
d compute density of hit values  
c comput mean per-nt coverage

-d, --binloc [method] specify method of determining location of each bin  
g interpret relative bin locations as genetic distance, requires stranded gene list  
 (default)  
p interpret relative bin locations as physical  
 distance

-s, --strands [method] specify handling of strand identifiers:  
b utilize hits from both strands, also for use  
 with data that lacks strand information  
 (default)  
s utilize only same-strand hits, requires -p, -m,   
 -h c, or -h e  
o utilize only opposite-strand hits, requires -p,  
 -m, -h c, or -h e

--nostrand indicates no strand specific methods are to be used, applies -s b, -l p, -a p, and -d p

--nohead suppresses printing of header to output file

*File Types*:

Bin Definition File: A bin definition file consists of two tab-separated columns, the first containing the bin start, and the second containing the bin end:

-100 -1  
0 99  
100 299  
400 500  
501 900

This file may contain negative values, gaps between the end of a bin and the start of the next, and variable bin sizes, but may not contain any overlapping bins. Additionally, all bin end values must be greater than or equal to their corresponding bin start.

Hit File(s): Five hit file types are supported: standard bedGraph (G), bedgraph with 1-based start coordinate (g), basic bed (b), extended bed (e), and cppmatch format (c). Bedgraph format is described in detail [here](https://cgwb.nci.nih.gov/goldenPath/help/bedgraph.html), and bed format [here](http://genome.ucsc.edu/FAQ/FAQformat.html#format1). For the purposes of this program, basic bed consists of the first three, required, fields (chrom, chromStart, chromEnd). Extended bed must include at least the first five fields (chrom, chromStart, chromEnd, name, score), and may optionally contain additional fields. Of the additional fields, only strand will be included in the analysis, the rest will be ignored. The dataValue field for bedgraph format will be used to calculate bin values. For bed formats, each entry will be assigned a value of 1 for this purpose. ***Both bed and bedGraph format utilize 0-based coordinates for the chromStart field, and 1-based coordinates for chromEnd, which are interpreted as such by make\_heatmap in the cases of bed and standard bedGraph input. In order to maintain compatibility with previously written tools (e.g. bowtie2bedgraph, extract\_fragments), bedgraph input utilizing 1-based start coordinates is also accepted.*** Cppmatch format consists of five required, and one optional, tab separated columns:

[ID] [Value] [Chromosome] [Physical Start] [Physical End] [Strand]

The ID column may contain any relevant information, and is not required to be unique. The value column must contain a number, to be used in the calculation of bin values, and the strand column may contain only the values “plus”, “minus”, “+”, or “-“. The physical start and end, corresponding to the boundaries of the aligned read, probe, or other feature, must be integer values, with physical end being greater than or equal to physical start. The chromosome identifiers used in all hit file types must match those contained in the gene list file exactly.

Gene List File: The gene list file consists of five required, and one optional, tab-separated columns, with each line corresponding to one gene or other feature of interest:

[Gene ID] [Description/User-defined Anchor] [Chromosome] [Physical Start] [Physical End] [Strand]

The gene ID field must contain a unique value for each line, however, physical overlap of features is allowed. If the option for user-specified anchor positions is specified (-a u), the second field must contain an integer value that falls between those in the physical start and physical end columns, otherwise, this field may contain any relevant information to be passed to the output file. There are no restrictions placed on the value contained in the chromosome field, however, the values must match those in the hit file(s) exactly. The physical start and physical end fields may consist of integer values only, corresponding to the boundaries of the gene or other feature, with physical start less than or equal to physical end. The optional strand column may consist of only the values: “plus”, “minus”, “+”, or “-“.

Output File: The output file consists of two parts, the header, which identifies the options and input files utilized in the file’s creation, and the matrix itself. Each row of the matrix consists of the six tab separated columns contained in the input gene list file, and one column corresponding to each bin, each containing the total or average value of hit file elements mapped to the bin.

*Detailed Option Descriptions*:

Bin Definition Method (-b): Relative bin locations may be defined either in a file (f, described above), or on the command line (c, v). Command line definition may consist of either specifying the first bin start location, bin size, and bin count (c), or the bin count alone (v). The first will result in uniformly sized bins with locations relative to the selected anchor point. The second will result in variable sized bins spanning the length of each gene list element. When the length of a gene list element is not evenly divisible by the number of bins, a number of the most downstream bins sufficient to cover the remainder are extended by one nucleotide. If the number of bins is greater than the length of the gene list element, “fractionally” sized bins are created by assigning more than one bin to each position. In this case, an extra bin is included for the most upstream locations, sufficient to cover any remainder when the bin count is not evenly divisible by the interval length. A negative start location may be used with command-line definition, however, two consecutive dashes (--) must be inserted between the options and the first non-option command line argument for the value to be correctly interpreted:

make\_heatmap -b c -- hit\_file.match gene\_list.txt out.matrix -100 1 201

or

make\_heatmap -b c -p hit\_file\_forward.bedgraph -m hit\_file\_reverse.bedgraph -- gene\_list.txt out.matrix -100 1 201

Hit Match Location (-l): Unless mean per-nt coverage per bin is requested (see bin value section below), each entry in the hit file must be resolved to a single location so that it may be mapped to a single bin. This point may be the genetic start (s), genetic end (e), physical start (p), physical end (d), or center (c). All supported hit file types define each hit location by a physical start and physical end. When strand information is available, these may be interpreted as genetic start and end locations, where for the plus strand, genetic start and end are equivalent to physical start and end, and for the minus strand, genetic start is equivalent to physical end, and genetic end is equivalent to physical start. If strand information is not included in the input hit file, and the -p/-m options are not utilized, physical start, physical end, or center match locations must be used.

Gene Anchor Location (-a): For each gene list element, the anchor is the location to which relative bin start and end values are added to produce the absolute bin start and end positions, that are subsequently queried. The anchor may be the genetic start (s), genetic end (e), physical start (p), physical end (d), or it may be specified directly by the user (u). The gene list file defines each element location by a physical start and end, which, as with the hit match location, may be interpreted as genetic start and end when strand information is available. For plus strand elements the genetic start and end are equivalent to the physical start and end, while for minus strand elements, the genetic start is equivalent to the physical end, and the genetic end is equivalent to the physical start. User-specified anchor points must be a desired physical location within the boundaries of each gene list element. These may be specified in the second column of the gene list file. If strand information is not included in the input gene list file, physical start, physical end, or user-specified anchor locations must be used.

Method of Determining Bin Value (-v): By default, the value reported for each bin is the sum of the values associated with all hit locations intersecting the bin (t). Alternatively, the average value (a) may be reported, which is the total divided by the number of intersecting hit locations, the density (d), which is the total divided by the bin size, or mean per-nt coverage(c). In this final case, the value of each line in the hit file is assumed to cover the full interval from the start coordinate to the end. This value is added to a bin’s total for each intersecting nucleotide. The reported value for each bin is this total divided by its length.

Method of Determining Bin Location (-d): Bin locations may be determined using either genetic (g) or physical (p) distance. When genetic distance is utilized, absolute bin start and end locations are determined for plus strand gene list elements by adding the relative bin locations to the anchor point, and for minus strand elements by subtracting relative bin locations from the anchor point. When physical distance is utilized, relative positions are added to the anchor point for both plus strand and minus strand elements. If strand information is not included in the input gene list file, physical distance must be used.

Strand Handling Method (-s): When mapping hit file entries to bins, three methods may be used: same-strand matching (s), opposite-strand matching (o), or non-specific matching (b). For same-strand matching, plus strand hit file entries are mapped only to plus strand bins, and minus strand entries to minus strand bins. For opposite-strand matching, plus strand hit file entries are mapped only to minus strand bins, and minus strand entries to plus strand bins. For non-specific matching, all hit file entries are mapped to both plus and minus strand bins. If either the input hit file or gene list file lack strand information, non-specific matching must be used.

**Compiling make\_heatmap**

This program was originally developed for Red Hat Enterprise Linux Server release 5.11 (Tikanga) using gcc 4.1.2 20080704 and libstdc++ (libc6). To compile on a similar environment, use:

g++ -O3 -o make\_heatmap make\_heatmap.cpp -lpthread

To compile on Mac OS X 10.9 or newer, use:

g++ -O3 -o make\_heatmap make\_heatmap.cpp -stdlib=libstdc++ -lpthread

On systems with no support for pthreads, a single-threaded version may be built using:

g++ -O3 -o make\_heatmap make\_heatmap.cpp -DSINGLE